



Process Expert

User Guide

EIO0000001114.18
08/2023



Legal Information

The information provided in this document contains general descriptions, technical characteristics and/or recommendations related to products/solutions.

This document is not intended as a substitute for a detailed study or operational and site-specific development or schematic plan. It is not to be used for determining suitability or reliability of the products/solutions for specific user applications. It is the duty of any such user to perform or have any professional expert of its choice (integrator, specifier or the like) perform the appropriate and comprehensive risk analysis, evaluation and testing of the products/solutions with respect to the relevant specific application or use thereof.

The Schneider Electric brand and any trademarks of Schneider Electric SE and its subsidiaries referred to in this document are the property of Schneider Electric SE or its subsidiaries. All other brands may be trademarks of their respective owner.

This document and its content are protected under applicable copyright laws and provided for informative use only. No part of this document may be reproduced or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), for any purpose, without the prior written permission of Schneider Electric.

Schneider Electric does not grant any right or license for commercial use of the document or its content, except for a non-exclusive and personal license to consult it on an "as is" basis.

Schneider Electric reserves the right to make changes or updates with respect to or in the content of this document or the format thereof, at any time without notice.

To the extent permitted by applicable law, no responsibility or liability is assumed by Schneider Electric and its subsidiaries for any errors or omissions in the informational content of this document, as well as any non-intended use or misuse of the content thereof.

As part of a group of responsible, inclusive companies, we are updating our communications that contain non-inclusive terminology. Until we complete this process, however, our content may still contain standardized industry terms that may be deemed inappropriate by our customers.

© 2023 – Schneider Electric. All rights reserved.

Table of Contents

Safety Information	14
About the Book.....	15
Understanding EcoStruxure Process Expert.....	19
Presentation.....	20
Introducing EcoStruxure Process Expert.....	20
Operating Modes	22
EcoStruxure Process Expert Platform and Explorers	23
Capabilities	24
Hardware and Software Capabilities.....	24
Supported Data Types.....	28
Naming Rules.....	31
General Concepts.....	35
Object Model Concept.....	36
Object-Based Engineering	36
Object Template Description	38
Application Templates.....	42
Topological Templates	43
Interface Models	44
Instances of Templates.....	45
System Engineering Life Cycle Concept	46
Overview	46
Global Templates Definition Stage.....	48
Instantiation Stage	49
Project Definition Stage	53
Assignment Stage	56
Generation Stage.....	58
Refinement Stage	59
Configuration Stage.....	61
Mapping Stage.....	63
Build Stage	65
Deployment Stage.....	67
Execution Stage.....	68
Simplified System Engineering Life Cycle.....	69
Working with EcoStruxure Process Expert	70
Starting and Quitting System Server and Clients	71
Logging into and out of the System Server and Clients.....	72
Locking and Unlocking EcoStruxure Process Expert.....	74
Starting the System Server Again.....	75
Client/Server Connection.....	76
Quitting EcoStruxure Process Expert.....	80
Multiuser Functionality	82
Multiuser Functionality.....	82
Action Traceability	84
Notification Panel	85
Audit Trail.....	88
Activity Log Files.....	92
Using the Help.....	94
Software Participant Help	95

Using Help for Different Versions of Schneider Electric	
Templates	96
Help on User-Created Templates	98
User Interface	102
Engineering Client User Interface Description	103
About Dialog Box	106
Engineering Client and Explorer Work Areas.....	107
Organizing Work Areas	110
Working with Panes	113
Using Menus	117
Software Participant Windows	118
Filtering and Sorting Data	120
Keyboard Shortcuts	123
Engineering a System	124
System Creation Workflow and Tasks	125
System Creation Workflow.....	126
System Creation Tasks.....	130
Managing Systems	134
Organizing Systems	135
Creating Systems	137
Managing the System Access Password	141
Creating System Backups	145
Scheduling System Backups.....	150
Restoring Systems.....	152
Instantiation Stage.....	157
Instantiating Templates.....	158
Organizing the System Application	159
Application Explorer	160
Browsing Application Templates	163
Using the Application Browser Pane	166
Creating Application Folders	170
Assigning Icons to Application Folders	176
Creating Instances	177
Counting Instances	180
Accessing Instance Properties	182
Configuring Instances.....	188
Using the Asset Workspace Pane.....	193
Instance Indicators.....	195
Instance Progress Indicators.....	196
Instance Validity Indicator	198
Instance Write-Lock Indicator	199
Linking Instances by Using the Asset Workspace Editor.....	200
Asset Workspace Editor	200
Using the Simplified View.....	203
Instance Link Status	204
Linking Instances	206
Using the RTNS Interface	208
Managing Application Folders and Instances	210
Viewing Instance Configuration	211
Changing the Application Hierarchy	213
Copying and Pasting Application Folders and Instances.....	214

Moving Application Folders and Instances.....	216
Updating and Replacing Templates of Instances.....	220
Exporting and Importing Application Objects to/from CSV and XML	
Formats	227
Exporting Folders and Instances to CSV and XML Formats	228
Exporting Asset Workspaces to CSV Format.....	230
CSV Application Export File	231
Editing the CSV Application Export File.....	235
XML Application Export File	244
Editing the XML Application Export File to Manage Folders and	
Instances.....	246
Editing the XML File to Manage Interface Links	251
Importing a CSV or XML File Into the Application	254
Project Definition Stage.....	262
Project Explorer and Project Browser	263
Project Explorer	264
Project Browser	266
Control Project Definition Stage	268
Creating Control Projects	269
Creating Control Project Sections	273
Managing Control Projects	274
Control Project Section Order.....	275
Exporting Control Projects	276
Importing Control Projects.....	280
Importing a Control Expert Project	294
Managing Data Backup Files.....	296
Supervision Project Definition Stage.....	298
Internal Supervision Architecture	299
Creating Supervision Projects	300
Creating Supervision Project Components.....	304
Creating Clusters	305
Creating Supervision Servers and Clients	306
Creating I/O Devices	309
Creating Communication Ports.....	312
Implementing Redundancy	314
Managing Supervision Projects.....	317
Exporting a Supervision Project.....	318
Importing a Supervision Project.....	323
Participant Project Indicators	333
Participant Project Progress Indicators	334
Participant Project Validity Indicators	336
Participant Project Write-Lock Indicator.....	338
Viewing the Last Action Summary	339
Assignment Stage	340
Creating and Managing Sections, Instances, and Containers	341
Assignment Editor	342
Creating Control Project Sections.....	348
Creating Instances In-Place	351
Creating Supervision Tag Containers	353
Creating Supervision Pages.....	358
Selecting Supervision Page Templates	359

Assigning Facets of Instances.....	362
Defining the Control Facet Assignment Mechanism	363
Previewing Facets Assignments	372
Assigning Control Facets of Instances	375
Assigning Supervision Facets of Instances.....	377
Assigned Facet Description.....	379
Automatic Assignment of Remaining Facets.....	381
Automatic Assignment of Supervision Facets	383
Viewing the Application Structure in Control Projects	384
Managing Assigned Facets.....	389
Facet Actions After Assignment.....	390
Viewing Facet Assignments	392
Order of Control Facets	394
Navigating from Facets to the Instance	395
Generation Stage	396
General Description	397
Generation Process Description	398
Control Project Specifics	401
Automatic Generation of Sections.....	407
Selecting Versions of Types in Control Facets During Generation	409
Statuses of Facets	424
Generating a Project for the First Time.....	426
Generating Entire Projects	427
Generating Sections, Containers, and Nodes	428
Generating Projects After Changes	429
Generating Participant Projects After Changes	430
Regenerating a Control Project	434
Generating Equipment in Supervision Projects.....	437
Generating Equipment Based on the Application	437
Generation Diagnostic Messages.....	440
Message Description	440
Managing Generated Facets.....	442
Working With Generated Facets.....	442
Refinement Stage.....	445
Control Project Refinement Stage	446
Refining the Logical Control Project	446
Identifying Code Generated by EcoStruxure Process Expert	452
Creating and Updating Device DDT Variables in Logical Control Projects.....	455
Managing the Consistency of Control Logic.....	460
About Variables for Peer to Peer Communication.....	463
Managing Network Variables.....	471
Managing Peer to Peer Communication.....	475
Overview	475
Peer to Peer Communication Configuration – Remote Project Selection 1/2.....	476
Peer to Peer Communication Configuration – Variable Mapping 2/ 2	478

Sections and Logic Created for Peer to Peer	
Communication	481
Creating Peer to Peer Communication Mapping	484
Modifying and Deleting Peer to Peer Communication Mapping	487
Managing Peer to Peer Communication Example	492
Supervision Project Refinement Stage.....	508
Refining the Supervision Project.....	508
Configuration Stage	512
Topology Explorer Description	513
Topology Explorer	513
Catalog of the Topology	516
Scope of the Topology Explorer	519
Control Participant Parameter Correspondence.....	522
User Preferences	524
SYSTEM PROJECT Pane.....	525
SYSTEM PROJECT Pane Description	525
SYSTEM PROJECT Pane - Topology Node.....	527
SYSTEM PROJECT Pane - Logical Networks Node	532
SYSTEM PROJECT Pane - Physical Views Node.....	534
Topology Explorer Views.....	536
Physical Views	536
LOGICAL NETWORK VIEW	541
Communication Mapping Views	544
I/O Mapping Editor	547
DHCP VIEW	550
Fieldbus Editor	553
ACCESS CONTROL VIEW	556
Basic Diagnostic	558
Rack Viewer	559
Creating the Topology of the System	562
Organizing the Topology in the SYSTEM PROJECT Pane	563
Adding Topological Entities to a System	565
Adding Entities to a Physical View from the SYSTEM PROJECT Pane	568
Topological Entity Context Menu Commands.....	570
Creating and Configuring Logical Networks	576
Representation of Entities and Ports.....	578
Representation of Entities in Physical Views.....	578
Representation of Ports of Entities in Physical Views	582
Configuring Topological Entities	584
Selecting Topological Templates.....	584
Using Configuration Change On The Fly (CCOTF).....	585
Setting the Webuser Password for M580 Controllers	588
Synchronizing Changes.....	590
Configuring Controllers.....	591
Configuring Workstations.....	597
Configuring STB Islands	599
Configuring PROFIBUS Remote Masters and Decentralized Peripherals	601
Configuring Modbus TCP and EtherNet/IP Devices.....	603

Configuring HART Modules.....	606
Connecting Ports of Entities by Physical Links.....	608
Connecting Quantum and M340 Controllers and PRMs to Logical Networks	615
Configuring Fieldbuses.....	620
Managing Field Devices on Fieldbuses	620
Managing the Topology	622
Updating Templates of Topological Entities.....	622
Assigning Devices and Modules to DHCP Servers.....	624
Managing Logical Networks and IP Addresses	625
Highlighting Logical Networks in Physical Views	625
Changing IP Address Assignments to Logical Networks at Topology Level.....	627
Using Device Type Managers	629
Using DTMs.....	629
Viewing DTMs Used by Devices of the Topology.....	633
Updating DTMs Used by Devices	636
Implementing Communication With Non-EPE Managed Devices and Controllers	638
Configuring I/O Profiles of Devices	638
Configuring I/O Scanning of Devices and Local Slaves.....	641
Creating Items in Input/Output Registers.....	642
Data Exchange with Local Slaves - Example	645
Exporting and Importing Topological Entities (SBK)	647
Exporting Topological Entities.....	647
Importing Topological Entities.....	651
Exporting and Importing I/O Devices to/from CSV Format	660
Exporting I/O Devices to CSV Format	661
Editing the CSV Device Export File	663
Importing a CSV File Into the Topology	667
Properties of Topological Entities	673
PROPERTIES and Properties Inspector Panes	673
PROPERTIES Pane - DESCRIPTION Tab.....	675
General Category - DESCRIPTION Tab	675
PROPERTIES Pane - CONFIGURATION Tab.....	676
General Category - CONFIGURATION Tab	676
Interfaces Category - CONFIGURATION Tab	679
Physical Ports Category - CONFIGURATION Tab.....	682
Server Memory and Client Memory Categories - CONFIGURATION Tab.....	685
Communication Card Category - CONFIGURATION Tab	686
Fieldbus Category - CONFIGURATION Tab	687
DTM Category - CONFIGURATION Tab	688
Hot Standby Category - CONFIGURATION Tab	689
PROPERTIES Pane - SECURITY Tab.....	690
Global Policy Category - SECURITY Tab.....	690
Services Category - SECURITY Tab	691
Password Protection Category - SECURITY Tab.....	692
Control Expert Category - SECURITY Tab.....	693
PROPERTIES Pane - SERVICES Tab.....	694

DHCP Server and DHCP Client Categories - SERVICES	
Tab.....	694
FDR Server Category - SERVICES Tab	696
RSTP Category - SERVICES Tab	697
NTP Category - SERVICES Tab	698
Syslog Server Category - SERVICES Tab	700
SNMP Category - SERVICES Tab	701
QoS Category - SERVICES Tab	703
Advanced Category - SERVICES Tab	705
Email Category - SERVICES Tab	706
OPC UA Category - SERVICES Tab	707
Workstation Services Category - SERVICES Tab	708
PROPERTIES Pane - I/O PROFILE Tab	709
Ethernet Modbus TCP and Ethernet EtherNet/IP Categories - I/	
O PROFILE Tab	709
IOS Type Category - I/O PROFILE Tab	716
PROPERTIES Pane - DIAGNOSTIC Tab	717
System Diagnostics Category - DIAGNOSTIC Tab	717
Mapping Stage.....	719
Control Project Mapping Stage.....	720
Creating Control Project Executables.....	721
Mapping Services.....	723
Mapping Communication.....	725
Creating, Modifying, and Deleting Communication	
Channels.....	729
Variable Mapping Editor	738
Mapping Network Variables.....	741
Mapping Hardware.....	747
Exporting and Importing Hardware Mappings.....	754
Exporting Hardware Mapping Data.....	755
Editing the Hardware Mapping Export File.....	757
Importing Hardware Mappings.....	761
Supervision Project Mapping Stage.....	763
Creating Supervision Project Executables.....	764
Mapping Services.....	767
Mapping Communication.....	769
Build Stage.....	773
Control Project Build Stage.....	774
Building the Logical Control Participant Project.....	775
Changing the Built Control Participant Project.....	781
Managing Hardware Abstraction Layer (HAL) Variables and	
DFBs.....	784
Supervision Participant Project Build Stage.....	787
Building the Supervision Participant Project.....	788
Changing the Built Supervision Participant Project.....	792
Deployment Stage.....	793
Control Project Deployment Stage.....	794
Control Participant Project Deployment Process.....	795
Deploying to the Controller Simulator.....	799
Communication Between the Computer and Engines.....	803

Managing Passwords of Deployed Control Participant Projects	804
Deploying to Redundant Controllers	812
Deploying the Control Participant Project	814
Re-Deploying the Control Participant Project	819
Deploying Data to Controllers and Controller Simulators	824
Deploying Control Project Changes	830
Changes That Require Stopping the Controller	840
Supervision Project Deployment Stage	841
Preparing for Deployment	842
Deploying the Built Supervision Participant Project From the Executable	843
Deploying the Built Supervision Participant Project from the Workstation	846
Deploying Supervision Participant Project Changes	851
Execution Stage	854
Control Project Execution Stage	855
Control Participant Project Execution Process	855
Execution With Redundant Controllers	859
Starting and Stopping Controllers	861
Synchronizing Redundant Controllers	866
About Changes Made Online	870
Making Changes Online	874
Backing Up Controller Data	880
Importing Sections from the Logical Control Participant Project	885
Redundant Controller Deployment and Executions Scenarios	889
Redundant Controller Reservation and Connection Scenarios	892
Updating the Logical Control Participant Project and Controller Configuration	897
Supervision Project Execution Stage	907
Executing the Deployed Supervision Participant Project	907
Engineering M580 Safety Systems	908
Using Modicon M580 Safety Controllers	909
M580 Safety Systems in the Instantiation Stage	911
M580 Safety Systems in the Project Definition Stage	912
Creating Control Projects of M580 Safety Controllers	912
Exporting and Importing the Control Project of an M580 Safety Controller	914
Importing a Safety-Related Control Project Associated to an Existing M580 Safety Controller	917
M580 Safety Systems in the Assignment Stage	919
M580 Safety Systems in the Generation Stage	920
M580 Safety Systems in the Refinement Stage	921
M580 Safety Systems in the Configuration Stage	923
Creating and Configuring M580 Safety Controllers	923
Managing the Functional Safety Password Protection Password	925
Exporting and Importing M580 Safety Controllers	927

M580 Safety Systems in the Mapping Stage	929
M580 Safety Systems in the Build Stage.....	930
M580 Safety Systems in the Deployment Stage	931
M580 Systems in the Execution Stage.....	932
Using Supervision with M580 Safety Systems	934
Managing Communication.....	935
Communication Through Modbus TCP Ethernet Implicit Messaging (I/O Scanning)	936
Communicating with Devices Through Modbus TCP Implicit Messaging (I/O Scanning).....	937
Communicating with STB Islands Through Modbus TCP Implicit Messaging.....	940
Communicating with PROFIBUS Remote Masters Through Modbus TCP Implicit Messaging	941
Peer to Peer Communication Through Modbus TCP Ethernet Implicit Messaging.....	943
Communication by Using Modbus TCP Ethernet Explicit Messaging	944
Communication with Devices by Using Modbus TCP Ethernet Explicit Messaging	944
Communication by Using EtherNet/IP Explicit Messaging	949
Communication with Devices by Using EtherNet/IP Explicit Messaging.....	949
Communication Through Modbus Explicit Messaging	952
Communication with Devices Through Modbus Explicit Messaging.....	953
Example Overview	958
Example A.....	962
Example B.....	966
Example C.....	972
Example D.....	979
Monitoring Modbus Data From Supervision.....	984
Monitoring Modbus Device Data From Supervision	984
Communication by Using a PROFIBUS Primary Module	988
Communication With Devices by Using a PROFIBUS Primary Module	988
Managing, Modifying, and Creating Templates	990
Global Templates Library.....	991
Global Templates Explorer.....	992
Global Templates Folders	998
Global Templates	1000
Managing the Control Constituent Password.....	1002
Global Templates Editors	1006
Opening Global Templates Editors	1007
Opening Global Templates Editors.....	1007
Interface Editor	1008
Interface Editor	1008
Facet and Composite Template Editors.....	1013
Facet Editor	1014
Composite Editor	1018
Common Template Editor Components	1021

Common Template Editor Toolbars and Menus	1022
Common Template Editor Panes	1028
Common Template Editor Filters.....	1030
Using the Radar View	1031
Using the Simplified View in Template Editors.....	1032
Managing Global Templates	1034
Managing Global Templates	1035
Saving Changes In Global Templates	1035
Copying and Pasting Global Templates and Folders	1038
Exporting Templates.....	1039
Importing Templates	1042
Purging the Global Templates Library	1049
Updating, Replacing, and Duplicating Global Templates.....	1051
Template Modification Strategy	1052
Updating Global Templates and Templates of Elements	1054
Updating Global Templates at the Folder Level	1058
Replacing the Template of an Element of a Global Template	1060
Duplicating Global Templates	1065
Replace Dialog Box	1069
Editing and Extending Interfaces In-Place	1070
Updating References in Parent After Editing Child Element – Example	1075
EcoStruxure Process Expert Services.....	1079
System Documentation	1080
About System Documentation.....	1081
System Documentation Content.....	1083
Application Domain Data	1083
Project Domain Data	1084
Topology Domain Data	1086
Global Templates Domain Data	1088
Generating System Documentation	1089
Managing the Consistency of the Topology	1090
Analyzing the Topology	1091
Analyzing the Topology	1091
Analysis Rules	1094
Analysis Rules - 1000 ID Series - Logical Networks	1095
Analysis Rules - 2000 ID Series - IP Addresses	1096
Analysis Rules - 3000 ID Series - Logical Network Configuration	1098
Analysis Rules - 4000 ID Series - Routing	1100
Analysis Rules - 5000 ID Series - Physical Links.....	1101
Analysis Rules - 6000 ID Series - Services	1102
Analysis Rules - 7000 ID Series - M580 Controllers	1104
Analysis Rules - 9000 ID Series - I/O Profiles	1105
Analysis Rules - 10000 ID Series - Hardware	1106
System Monitoring.....	1107
Displaying the Communication Status of Entities in Physical Views.....	1107
Displaying the Identification Mismatch Icon	1111
Displaying Operating States of M580 Controllers.....	1112

Displaying Operating States of Altivar Process Devices	1114
Displaying Operating States of TeSys Devices	1115
Displaying Operating States of Generic Automation Devices	1116
Displaying Operating States of Hirschmann Switches	1117
Basic Diagnostics	1118
Basic Diagnostics for Altivar Process Devices	1118
Content Repository	1119
Content Repository Explorer.....	1120
Temporary Contents Repository Files	1124
Appendices	1125
Linking Instances by Using the Links Editor	1126
Links Editor	1127
Linking Instances by Using the Links Editor.....	1130
Glossary	1132
Index	1141

Safety Information

Important Information

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a "Danger" or "Warning" safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result in** death or serious injury.

CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

Please Note

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

About the Book

Document Scope

This document contains general information about EcoStruxure Process Expert and guides you through the stages to engineer a system.

This document is organized in five main parts:

- **Understanding EcoStruxure Process Expert:**
Contains information on the object model and the system engineering life cycle concepts that the implementation of automation systems is based on.
- **Working with EcoStruxure Process Expert:**
Describes the graphical user interface and how to use the different tools that are at your disposal.
- **Engineering a system:**
Provides step by step instructions to help you implement a system by following the system engineering life cycle.
- **Global Templates Library and editors:**
Describes the Global Templates Library, the templates that it can contain, and the editors that allow you to manage them.
- **EcoStruxure Process Expert services:**
Contains information on the content repository, the generation of system documentation, and various services for the topology of a system.

It is written for users who are familiar with the engineering of Control and Supervision systems.

Validity Note

This document has been updated for the release of EcoStruxure Process Expert 2023.

The characteristics that are described in the present document, as well as those described in the documents included in the Related Documents section below, can be found online. To access the information online, go to the Schneider Electric home page www.se.com/ww/en/download/.

The characteristics that are described in the present document should be the same as those characteristics that appear online. In line with our policy of constant improvement, we may revise content over time to improve clarity and accuracy. If you see a difference between the document and online information, use the online information as your reference.

Related Documents

Title of documentation	Reference number
EcoStruxure™ Process Expert, Security Deployment Guide	EIO0000004234 (ENG)
EcoStruxure™ Process Expert, General Purpose Library, User Guide	EIO0000004045 (ENG)
EcoStruxure™ Process Expert, General Purpose Library, Process Templates, Reference Manual	EIO0000004043 (ENG)
EcoStruxure™ Process Expert, General Purpose Library, Device Templates, Reference Manual	EIO0000004044 (ENG)
EcoStruxure™ Process Expert, General Purpose Library Classic, Process Templates, Reference Manual	EIO0000000987 (ENG)
EcoStruxure™ Process Expert, General Purpose Library Classic, Device Templates, Reference Manual	EIO0000001308 (ENG)

Title of documentation	Reference number
EcoStruxure™ Process Expert, General Purpose Library Classic, Communication Templates, Reference Manual	EIO0000001311 (ENG)
EcoStruxure™ Process Expert, General Purpose Library Classic, Diagnostic Templates, Reference Manual	EIO0000001526 (ENG)
EcoStruxure™ Process Expert, Foundation Application Templates, User Guide	EIO0000002403 (ENG)
EcoStruxure™ Process Expert, Control Participant Services, User Guide	EIO0000001524 (ENG)
EcoStruxure™ Process Expert, Supervision Participant Services, User Guide	EIO0000001525 (ENG)
EcoStruxure™ Process Expert, Runtime Navigation Services, User Guide	EIO0000001574 (ENG)
EcoStruxure™ Process Expert, Global Templates, Reference Manual	EIO0000001986 (ENG)
EcoStruxure™ Process Expert, Installation and Configuration Guide	EIO0000001255 (ENG)
EcoStruxure™ Process Expert, Licensing Guide	EIO0000001261 (ENG)
EcoStruxure™ Process Expert, Implementing System Server Redundancy, Application Note	EIO0000004854 (ENG)
EcoStruxure™ Process Expert, Infrastructure Virtualization, Application Note	EIO0000003796 (ENG)
EcoStruxure™ Control Expert, Operating Modes	33003101 (ENG) 33003102 (FRE) 33003103 (GER) 33003104 (SPA) 33003696 (ITA) 33003697 (CHS)
EcoStruxure™ Control Expert, Application Versions and Controller Firmware, Compatibility Rules, User Manual	EIO0000005075 (ENG)
Modicon M580, Hardware Reference Manual	EIO0000001578 (ENG) EIO0000001579 (FRE) EIO0000001580 (GER) EIO0000001581 (SPA) EIO0000001582 (ITA) EIO0000001583 (CHS)
Modicon M580, Change Configuration on the Fly, User Guide	EIO0000001590 (ENG) EIO0000001591 (FRE) EIO0000001592 (GER) EIO0000001594 (ITA) EIO0000001593 (SPA) EIO0000001595 (CHS)

mySchneider Support Portal

Visit <https://www.se.com/myschneider> for support, software updates, and latest information on EcoStruxure Process Expert.

Product Related Information

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use software approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

The examples in this document are given for information only.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Adapt examples that are given in this document to the specific functions and requirements of your industrial application before you implement them.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: Templates shown in examples throughout this manual may differ from the actual templates contained in the supplied Schneider Electric libraries.

Terminology Derived from Standards

The technical terms, terminology, symbols and the corresponding descriptions in this manual, or that appear in or on the products themselves, are generally derived from the terms or definitions of international standards.

In the area of functional safety systems, drives and general automation, this may include, but is not limited to, terms such as *safety*, *safety function*, *safe state*, *fault*, *fault reset*, *malfunction*, *failure*, *error*, *error message*, *dangerous*, etc.

Among others, these standards include:

Standard	Description
IEC 61131-2:2007	Programmable controllers, part 2: Equipment requirements and tests.
ISO 13849-1:2015	Safety of machinery: Safety related parts of control systems. General principles for design.
EN 61496-1:2013	Safety of machinery: Electro-sensitive protective equipment. Part 1: General requirements and tests.
ISO 12100:2010	Safety of machinery - General principles for design - Risk assessment and risk reduction
EN 60204-1:2006	Safety of machinery - Electrical equipment of machines - Part 1: General requirements
ISO 14119:2013	Safety of machinery - Interlocking devices associated with guards - Principles for design and selection
ISO 13850:2015	Safety of machinery - Emergency stop - Principles for design
IEC 62061:2015	Safety of machinery - Functional safety of safety-related electrical, electronic, and electronic programmable control systems
IEC 61508-1:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: General requirements.
IEC 61508-2:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Requirements for electrical/electronic/programmable electronic safety-related systems.
IEC 61508-3:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Software requirements.
IEC 61784-3:2016	Industrial communication networks - Profiles - Part 3: Functional safety fieldbuses - General rules and profile definitions.
2006/42/EC	Machinery Directive
2014/30/EU	Electromagnetic Compatibility Directive
2014/35/EU	Low Voltage Directive

In addition, terms used in the present document may tangentially be used as they are derived from other standards such as:

Standard	Description
IEC 60034 series	Rotating electrical machines
IEC 61800 series	Adjustable speed electrical power drive systems
IEC 61158 series	Digital data communications for measurement and control – Fieldbus for use in industrial control systems

Finally, the term zone of operation may be used in conjunction with the description of specific hazards, and is defined as it is for a hazard zone or danger zone in the Machinery Directive (2006/42/EC) and ISO 12100:2010.

NOTE: The aforementioned standards may or may not apply to the specific products cited in the present documentation. For more information concerning the individual standards applicable to the products described herein, see the characteristics tables for those product references.

Understanding EcoStruxure Process Expert

What's in This Part

Presentation	20
Capabilities.....	24
General Concepts	35

Overview

This part introduces EcoStruxure Process Expert and describes the concepts and underlying mechanisms that are applied to engineer a system.

Presentation

What's in This Chapter

Introducing EcoStruxure Process Expert	20
Operating Modes.....	22
EcoStruxure Process Expert Platform and Explorers	23

Introducing EcoStruxure Process Expert

Overview

EcoStruxure Process Expert is an engineering tool that facilitates and automates the design, maintenance, commissioning, and operation of Control projects for Modicon controllers and the associated Supervision projects for SCADA software regrouped in a system.

It uses a client/server architecture that allows several users to work concurrently, page 82 on the same or different systems.

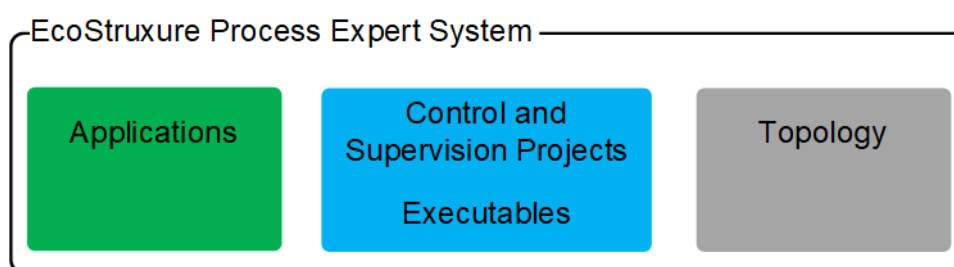
It displays systems and their components in separate Explorer windows, page 23 that you can open on engineering clients (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

System Description

An EcoStruxure Process Expert system, page 134 contains the data of the following components:

- The applications
- The Control and Supervision projects
- The topology

It also contains configurable properties and settings.



The system components are described further in this topic.

Terminology

The following terminology is used with EcoStruxure Process Expert.

Term	Description
Participant	<p>EcoStruxure Process Expert embeds the following software:</p> <ul style="list-style-type: none"> EcoStruxure Control Expert, herein the Control Participant (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>) AVEVA Plant SCADA, herein the Supervision Participant (see <i>EcoStruxure Process Expert, Supervision Participant Services, User Guide</i>) <p>You can open Participants on engineering clients to view and modify the program of a Control project and the data of a Supervision project. The Control Participant is also used for the configuration of controllers and certain devices of the topology.</p>
Global template	<p>Models the functionality of process equipment (such as a motor or a level switch) or hardware (such as the communication card of a motor starter or the software installed on a workstation). The Global Templates Library contains different types of templates that are organized in several libraries. Application templates are used to create applications in the system and contain the following constituents:</p> <ul style="list-style-type: none"> Code that generates the Control logic for the modeled functionality to create the program in the Control Participant. Preconfigured Supervision resources that create the Supervision data in the Supervision Participant.
Application	<p>Models the functionality of a process to be controlled and monitored. Applications, page 157 are created by configuring instances of application templates and organizing them in a folder structure that models the process hierarchy. Applications are created independently of the other components of a system (projects, topology).</p>
Control project	<p>Contains the sections, page 268 to which you can assign instances of an application to automatically create the corresponding program by using the code that the instances contain.</p>
Supervision project	<p>Contains the different containers, page 298 (such as tag containers and pages) to which you can assign instances of an application to automatically create the different types of tags (such as alarms or trends), animated graphics, and the Supervision infrastructure (such as clusters or I/O servers) by using the resources the instances contain.</p>
Topology	<p>Models the hardware (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>) that can be mapped to Control and Supervision projects and its communication networks. It contains the hardware, software, and communication configuration, page 512 of controllers, devices, and workstations. Workstations model computers running a controller simulator or Supervision software.</p>
Control executable	<p>The logic that is built, page 720 and deployed to the controller. It merges the program, controller configuration, and non-program data (such as Control project properties). A Control project can contain several executables, each targeting a different controller.</p>
Supervision executable	<p>The Supervision project that is built, page 763 and deployed to Supervision servers and clients. It merges the data from the Supervision project and the topology (the associated controller and Supervision workstations). A Supervision project can contain several executables that allow you to deploy it on various infrastructures.</p>
System engineering life cycle	<p>Regroups the various stages that you must complete to create and commission an EcoStruxure Process Expert system. The system engineering life cycle is represented in a stage diagram, page 46.</p>

Operating Modes

Overview

In EcoStruxure Process Expert, you can work with Control projects in offline and online mode. The mode applies to a [Control project executable](#), [page 720](#) and the controller or workstation that is mapped to it. You can use both modes simultaneously in a system.

Offline Mode

In offline mode, the engineering client is not connected to a physical controller. It allows you to use the Control Participant to configure controllers and refine the Control projects of the system.

Online Mode

In online mode, the engineering client is connected to a physical controller. This mode is used for commissioning and maintenance. It allows you to perform the following actions:

- Deploy a Control executable to the connected controller.
- Deploy changes you made to the system after deploying it.
- Refine the Control project residing in the controller.
- Update the topology and Control project of the system with changes you made to the Control project online.
- Start and stop the controller.

For details, refer to the chapter describing the [execution stage](#), [page 855](#).

NOTE: In online mode, you can also connect to the [controller simulator](#), [page 799](#) of the Control Participant. It allows you to perform similar actions as when connected to a controller but for testing purposes.

EcoStruxure Process Expert Platform and Explorers

EcoStruxure Process Expert Platform

The platform is the part of the engineering client user interface that excludes the Participants. It allows you to manage the system data that is not Participant-specific, such as templates, application instances, and some parts of the Control and Supervision projects.

Explorers

The following Explorers are available as part of the platform to help you complete the stages of the system engineering life cycle, page 46.

Explorer	Description
Global Templates Explorer	<p>Displays the Global Templates of the installed Libraries and those that you have created and/or imported.</p> <p>Templates are organized in a folder structure.</p> <p>You can search, create, modify (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>), and manage Global Templates.</p>
Systems Explorer	<p>Displays the systems that you create and that are stored in the database of the system server.</p> <p>You can back up and restore systems, configure system properties, and generate system documentation.</p> <p>The Explorer is the entry point to starting the system engineering life cycle.</p>
Application Explorer	<p>Displays the applications of a system in a tree view, which allows you to create a process hierarchy.</p> <p>You can search, create, configure, and manage the instances that model the applications of a system.</p>
Project Explorer	<p>Displays, at the platform level, the projects of a system that are managed by the Participants:</p> <ul style="list-style-type: none"> Control projects Supervision projects <p>You can create and manage the Control sections, Control facets of instances, the Supervision containers and tags, the Supervision infrastructure, and the executables of each project.</p> <p>NOTE: The Project Explorer allows you to open the Control and Supervision Participants.</p>
Topology Explorer	<p>Displays the topology of a system in a tree view or graphically in physical views.</p> <p>You can create, configure, and manage communication networks as well as the different topological entities that model the hardware infrastructure. The Explorer also allows you to interact with the controllers, devices, workstations, and networks of a system.</p> <p>NOTE: The Topology Explorer allows you to open the Control Participant.</p>

Capabilities

What's in This Chapter

Hardware and Software Capabilities	24
Supported Data Types	28
Naming Rules	31

Hardware and Software Capabilities

Overview

This chapter describes the main hardware and software capabilities of EcoStruxure Process Expert and additional capabilities of the embedded Control and Supervision Participants.

For more information on the specific capabilities of the Control Participant and Supervision Participant, refer to the [help of the respective Participants](#), page 95.

Hardware Platforms

For information on the integrated hardware architectures, refer to Configuration Stage (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

NOTE: You can include unsupported hardware platforms in EcoStruxure Process Expert systems, but no services are provided for these platforms.

Programming Languages

EcoStruxure Process Expert generates the program for controllers by using the Function Block Diagram (FBD) programming language.

The Control Participant supports additional [programming languages](#), page 26 for creating the user program, which can be used in Control projects managed by EcoStruxure Process Expert.

The Supervision Participant supports a structured, proprietary programming language for monitoring and controlling plant equipment, which can be used in Supervision projects managed by EcoStruxure Process Expert. For more information, refer to the [Supervision Participant help](#), page 95.

Template Libraries

EcoStruxure Process Expert uses object-oriented modeling for the engineering of systems. For this purpose, an extensive selection of Schneider Electric templates are provided in various Libraries, such as application templates modeling signals, hardware templates modeling the system infrastructure, and complex control module templates modeling the entire Control and Supervision functionality of process objects.

When a Library is installed, the templates are available in the **Global Templates Explorer** and can be used in explorers and editors. The encapsulated Control and Supervision constituents can be viewed in the corresponding Participants.

For information on architectures for which templates are available, refer to [Integrated Hardware Architectures](#) (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

Function and Function Block Libraries

The Control Participant includes an extensive Library, which encompasses function blocks for simple boolean operations, function blocks for strings and array operations, up to function blocks for controlling complex control loops.

The function blocks are provided as Libraries and grouped into families.

The function blocks can be used in the Control Participant with the FBD, LD, IL, and ST programming languages and added to Control projects managed by EcoStruxure Process Expert.

Controller Program

EcoStruxure Process Expert uses the following elements to organize the controller program:

- A Master task (MAST).
- A FAST task.
- FBD sections that are assigned to either task.

By using the Control Participant, you can add the following elements to the program:

- Sections that are assigned to MAST or FAST tasks
- AUX tasks (depending on the platform)
- Sections that are assigned to AUX tasks
- Sections for processing time controlled events (Timerx)
- Sections for processing hardware controlled events (EVTx)
- Subroutine sections (SR)

Some of these elements and their content cannot be viewed outside of the Control Participant (for example, the content of non-FBD sections).

Performance Scope

The table shows the main characteristics of EcoStruxure Process Expert and its Participants.

	EcoStruxure Process Expert	Control Participant	Supervision Participant
Programming languages			
Function Block Diagram (FBD)	+	+	–
Ladder Diagram (LD)	–	+	–
Instruction List (IL)	–	+	–
Structured Text (ST)	–	+	–
Sequential Function Chart (SFC)	–	+	–
Ladder Logic 984 (LL984)	–	+	–
Cicode	–	–	+
Schneider Electric Global Templates Libraries			
Foundation Library	+	–	–
General Purpose and segment Libraries	+	–	–
Function block Libraries⁽¹⁾			
Standard Library	–	+	–
Control Library	–	+	–
Communication Library	–	+	–
Diagnostics Library	–	+	–
I/O management Library	–	+	–
System Library	–	+	–
Obsolete Library	–	+	–
Memory card file management Library	–	+	–
General information			
Project browser/explorer	+	+	+
Template explorer	+	–	–
Instance editor	+	–	–
Create and modify object templates	+	–	–
Access control	+	+	+
Simulator	–	+	–
Context-sensitive and PDF help	+	+	+
Control			
Encapsulate DFB constituents in templates	+	–	–
Create and use data structures (DDTs)	+	+	+
Create and use derived function blocks (DFBs)	–	+	–
HMI operator screens	–	+	–
Supervision			
Encapsulate tags and animated graphics in templates	+	–	–
Design pages, animated graphics, faceplates	–	–	+
Use tags and alarms	+	–	+
Clusters	+	–	+

	EcoStruxure Process Expert	Control Participant	Supervision Participant
Server and client events	—	—	+
I/O devices	+	—	+
I/O, alarm, trend, and report servers	+	—	+
Redundancy	+	—	+
Load balancing (operation servers)	+	—	+
+ Available — Not available ⁽¹⁾ Availability of the function blocks depends on the hardware platform.			

Supported Data Types

Data Type Conversion

EcoStruxure Process Expert may use data types that are different from the types used by the Control Participant (see *EcoStruxure Process Expert, Control Participant Services, User Guide*). It converts them automatically to Participant-supported data types.

Standard Data Types

The following tables describes the data types that are used.

Data type	Description	Lower limit	Upper limit	Size
DateTime	A specific instance in time ⁽¹⁾ .	19-01-01 00:00:00.000	2100-12-31 23:59:59.999	8 bytes
Date	A Gregorian calendar date ⁽¹⁾ .	1900-01-01	2100-12-31	8 bytes
Time	An instant of time that recurs every day ⁽¹⁾ . Format: <i>hh:mm:ss.ms</i>	00:00:00.000	23:59:59.999	8 bytes
Duration	A duration of time as specified by Gregorian day, hour, minute, second, and millisecond components ⁽¹⁾ . Format: <i>D.hh:mm:ss.ms</i> Example: 1.05:10:30.245 If the value contains no day or ms component, they are not shown as 0 values. For example: 11:20:30 is displayed to represent 0 days, 11 hours, 20 min., 30 sec., and 0 ms.	0	9,223,372,036,854,775,807 in ticks NOTE: 1 tick = 100 ns	8 bytes
String	A sequence of unicode characters.	1 character of the standard ASCII table.	n characters of the standard ASCII table.	1 character = 1 byte
Boolean	A binary logic value.	False	True	1 byte
Float	Represented by using the 32-bit single-precision IEEE 754 format ⁽¹⁾ . 7-digit precision.	From approximately -3.4×10^{38} to -1.5×10^{-45}	From approximately 1.5×10^{-45} to 3.4×10^{38}	4 bytes
Double	Represented by using the 64-bit double-precision IEEE 754 format ⁽¹⁾ . 15 to 16-digit precision.	From approximately -1.7×10^{308} to -5.0×10^{-324}	From approximately 5.0×10^{-324} to 1.7×10^{308}	8 bytes
Decimal	A fixed-point decimal value with arbitrary precision ⁽¹⁾ . A maximum of 4 digits are allowed after the decimal separator.	0.0001	922337203685477.5807	12 bytes
Byte	8-bit signed integer.	-128	127	1 byte
Short	16-bit signed integer.	-32768	32767	2 bytes
Integer	32-bit signed integer.	-2,147,483,648	2,147,483,647	4 bytes
Long	64-bit signed integer.	-9223372036854770000	9,223,372,036,854,774,807	8 bytes
UnsignedByte	8-bit unsigned integer.	0	255	1 byte
UnsignedShort	16-bit unsigned integer.	0	65,535	2 bytes
UnsignedInt	32-bit unsigned integer.	0	4,294,967,295	4 bytes
UnsignedLong	32-bit unsigned integer.	0	18446744073709551615	4 bytes
(1) Microsoft Windows Regional Options may apply to the format of the type.				

User-Defined Data Types

Data type	Description	Lower limit	Upper limit	Size
Enum	An enumeration value is stored as String.	–	–	–
Variant	Any data type.	–	–	–

Naming Rules

Overview

When you change default parameters, such as identifiers or Supervision element properties, EcoStruxure Process Expert validates at different stages of the system engineering life cycle if they satisfy the naming rules that are described in this topic.

These rules can apply at the platform level and/or can be Participant-specific.

NOTE: Other Participant-specific rules may apply; however, EcoStruxure Process Expert does not validate them. For more information, refer to the respective Participant help, page 95.

Folder and Content Container Identifiers

Identifiers of application folders, system folders, Global Templates folders, and Content containers of the Content Repository must comply with the following rules:

- Must start with a letter⁽¹⁾ or the \$ character.
- Can only contain letters, digits, underscores, the \$ character, and spaces.

NOTE: Leading and trailing spaces are automatically removed.

NOTE: If used, the folder alias must start with a letter⁽¹⁾ and can contain only letters⁽¹⁾, digits, and underscores.

⁽¹⁾ For application folders and their alias, letters are restricted to ASCII codes 65 to 90 (A-Z) and 97 to 122 (a-z).

Instance Identifiers

Instance identifiers can only contain letters, digits, and underscores.

The maximum character length is set through the **Validation** property (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) of the **\$InstanceID** parameter of the template that the instance uses.

NOTE: Participants do not accept the use of certain reserved words for object names (for example, *DO* for the Supervision Participant). While EcoStruxure Process Expert allows using these reserved words as identifiers of instances in applications, doing so may prevent the corresponding Participant project from generating successfully when the reserved word is used as name of Participant project items (for example, *equipment*, page 437).

Topological Entity, Device, and Folder Names

Topological entity, device, and folder names must comply with the following rules:

- Must start with a letter.
- Can contain spaces, except for controller names.
- Cannot contain special characters other than underscore (_).
- Cannot exceed 32 characters.

NOTE: Leading and trailing spaces are automatically removed.

Control and Supervision Participant Project Identifiers

Identifiers of Control and Supervision Participant Projects and their components must comply with the following rules:

- Must start with a letter.
- Can contain spaces.
- Cannot contain special characters other than underscore (_).
- Cannot exceed 32 characters.

NOTE: Leading and trailing spaces are automatically removed.

Variable Names and Section Identifiers

Names of variables and identifiers of sections that are managed by the Control Participant must comply with the following rules:

- Maximum length: Refer to the maximum length of identifiers as specified under *Implementation-Dependent Parameters* in the IEC 61131-3 standard (*Appendices of Languages Reference* in the Control Participant help).
- Permitted characters: Refer to *naming conventions* in the glossary of the Control Participant help, *Operating Modes*.

Refer to the topic describing [data validation before generating](#), page 398.

Supervision Cluster Identifiers

Identifiers of Supervision clusters cannot exceed 16 characters in length.

Supervision Page Identifiers

Identifiers of pages must comply with the following rules:

- Maximum length: 64 characters
- Can contain spaces.
- Cannot contain the following characters: \ / : * ? " < > |

NOTE: Leading and trailing spaces are automatically removed.

Supervision Field Lengths

For variable tags:

Rule	Field	Value
Maximum field length	<i>Item Name</i>	79
	<i>Address</i>	254
	<i>Raw Zero Scale</i>	11
	<i>Raw Full Scale</i>	
	<i>Eng Zero Scale</i>	
	<i>Eng Full Scale</i>	
	<i>Eng Units</i>	8
	<i>Format</i>	11
	<i>Comment</i>	254
	<i>Deadband</i>	11
	<i>Data Type</i>	16

For trend tags:

Rule	Field	Value
Maximum field length	<i>Item Name</i>	79
	<i>Expression</i>	254
	<i>Trigger</i>	11
	<i>Format</i>	11
	<i>File Name</i>	253
	<i>No. Files</i>	4
	<i>Comment</i>	254
	<i>Period</i>	32
	<i>Storage Method</i>	64
	<i>Type</i>	32
	<i>Eng Units</i>	8

For advanced alarm tags:

Rule	Field	Value
Maximum field length	<i>Custom1... to Custom8...</i>	64
	<i>Paging Group</i>	80

For digital alarm tags:

Rule	Field	Value
Maximum field length	<i>Comment</i>	254
	<i>Custom1... to Custom8...</i>	64
	<i>Paging Group</i>	80

For reports (and facets assigned to **ServerEvents** containers):

Rule	Field	Value
Maximum field length	<i>Report Name</i>	79
	<i>Output Device</i>	16
	<i>Report Format File</i>	253
	<i>Comment</i>	48
	<i>Area</i>	16
	<i>Action</i>	16
	<i>Trigger</i>	254
	<i>Time</i>	32
	<i>Period</i>	32

For events (and facets assigned to **ClientEvents** containers):

Rule	Field	Value
Maximum field length	<i>Name</i>	16
	<i>Trigger</i>	254
	<i>Event (Action)</i>	64
	<i>Comment</i>	48
	<i>Time</i>	32
	<i>Period</i>	32

Supervision Fields Special Characters

The table describes the rules used during generation with regard to the contents of the following Supervision project field.

Field	Rule
<i>Name</i>	<p>Tag names must have the following syntax:</p> <pre>[<alpha> ' _ '] * [<alpha> <digit> ' \ ' ' _ ']</pre> <ul style="list-style-type: none"> Tag names must begin with either an alpha character (A-Z or a-z) or the underscore. The following characters must be either: <ul style="list-style-type: none"> Alphabetical characters (A-Z or a-z) Digits \ (backslash) _ (underscore) (vertical bar) <p>If you use other characters, the compile process cannot complete successfully.</p> <p>NOTE: Tag names can begin with a digit only if the <i>[General] TagStartDigit</i> parameter of the <i>citect.ini</i> file is set to 1 (the default value is 0).</p>

General Concepts

What's in This Chapter

Object Model Concept	36
System Engineering Life Cycle Concept.....	46

Overview

This chapter describes the object model concept and how it is applied within EcoStruxure Process Expert. It also describes the concept of engineering a system in stages.

Object Model Concept

Overview

This section describes the object model concept and the resulting templates that you can use to engineer systems.

Object-Based Engineering

Object Model Concept

Object models allow you to create the applications and the topology of a system.

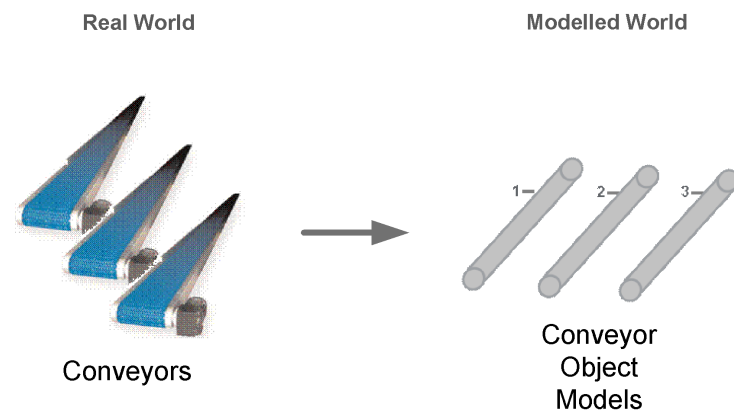
You can use the following types of object models to create applications:

- Process objects representing, for example, the functionality of pumps, valves, and sensors in several tanks.
- Device objects representing, for example, the control functions for a process object, such as a motor by using a drive.
- Communication objects representing, for example, the communication functions of devices communicating with the port of an M580 controller by using Modbus TCP explicit messaging.

Likewise, object models representing the topology are called topological objects. They can represent, for example, the controller module of an M580 controller.

The representation of a component or function of the real world by using object models is referred to as object-oriented modeling and is based on the abstraction principle: The process of reducing the information to only what is relevant for a particular component or function independently of the context.

The following figure illustrates the application object-oriented modeling of a conveyor.



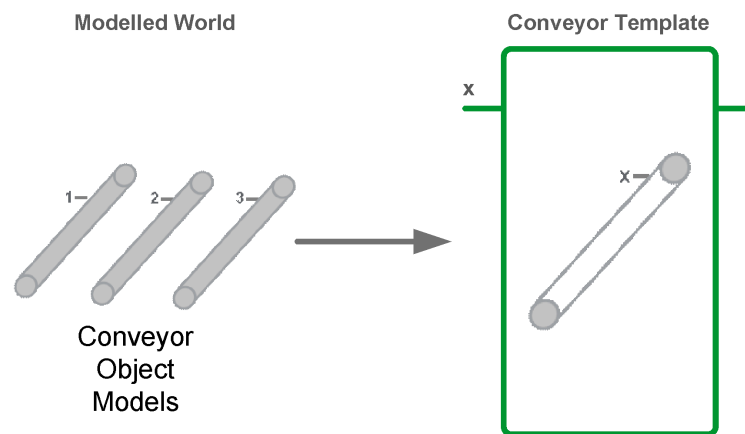
From Object Model to Object Template

Based on the object model, the next step consists in creating an object that models a component or a function of the real world. This object needs to be generic enough to be reused in different systems.

The standardization principle is used to achieve this objective: The process of finding the common characteristics of different components or functions.

The resulting object is called an object template.

The following figure illustrates the process of standardizing an object model representing a conveyor to create a template that allows modelling conveyors and similar items (such as mixers) in different systems.



Object Template Description

Overview

Schneider Electric object templates created from object models are called Global Templates (herein, templates) and are grouped in Libraries. When a Library is installed, its templates appear in the **Global Templates Explorer**, page 992. These templates can be used in systems created with EcoStruxure Process Expert.

Templates provide collaboration and data sharing between Participants and the platform. This results in reduced engineering effort and improved data consistency.

Template Classification

Typically, templates are combined by placing one inside the other and/or linking them to provide the diverse functionality of a physical device. There are three types of templates:

- **Facet templates:** They are the smallest templates available. They encapsulate a constituent that is provided by a specific Participant to deliver the corresponding service. The following types of facet templates exist:
 - Control facet templates providing Control services. For example, in the General Purpose Library, the service that is provided by the *\$MOTORGP_UL* facet template is the on/off motor management functionality. The functionality is implemented by a Control resource that is encapsulated inside the template. In this example, it is the *MOTORGP* function block.
 - Supervision facet templates providing Supervision services. For example, in the General Purpose Library, the service that is provided by the *\$BlowerLeftGP_m_CG* Genie facet template is the animated graphic representing a motorized blower that can be displayed on an operator screen. The functionality is implemented by a Supervision resource that is encapsulated inside the template. In this example, it is the animated graphic *Blower_Left*, which is part of the **Included** project supplied with the Library.
- **Composite templates:** They consist of one or more facet templates and/or other composite templates. Composite templates that are included inside another composite template are called composite references.

Depending on their composition, composite templates can be:

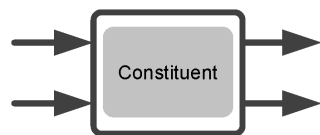
 - Control composite templates: Contain only Control references.
 - Supervision composite templates: Contain only Supervision references.
 - *Control module* templates: Highest level of composite template, which contains both Control and Supervision references.

NOTE: Facet templates that are included inside a composite template are called facet references.
- **Interface models:** They are considered a template because they are generic, configurable, and reusable. They are a means of linking the following:
 - Any combination of facet and composite templates.
 - Instances of templates.

There are different types of interface models depending on the objects that they can link.

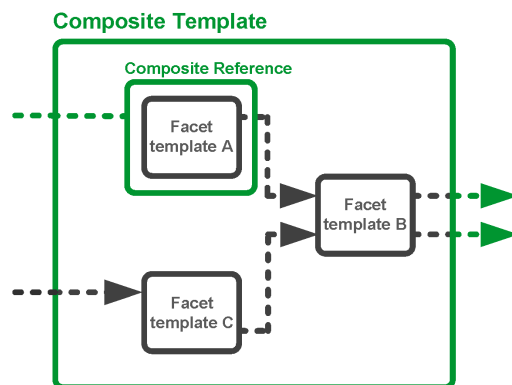
For details, refer to the EcoStruxure Process Expert help.

The following figure illustrates a single facet template.



The arrows represent the inputs and outputs of the facet which propagate the data.

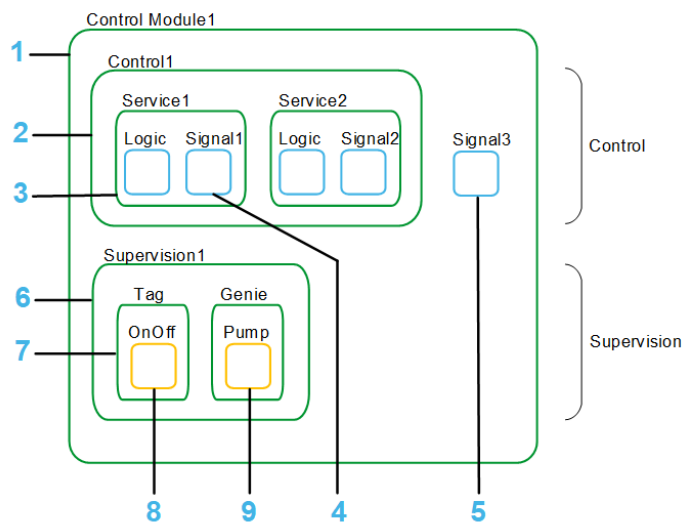
The following figure illustrates a composite template including another composite template (composite reference) and two facet templates.



- - - Interface links created by interface models




Template Terminology

The following figure shows the graphical representation of a sample control module template. The names of elements may vary.



NOTE: Interfaces are not represented since they are a means to manage data exchanges between references.

The following table explains the color coding that is used in the generic control module template *Control Module1*.

Color	Description
 Blue	Control facet template, which encapsulates a functionality that is provided by the Control Participant.
 Orange	Supervision facet template, which encapsulates a functionality that is provided by the Supervision Participant.
 Green	Composite template, which can contain: <ul style="list-style-type: none"> Control and/or Supervision facet templates Other composite templates

NOTE: This color coding is used in the user interface of EcoStruxure Process Expert to differentiate the different types of templates and related services.

The following table describes the items that compose the generic control module template *Control Module1* and the terminology that is used to designate these items depending on the context.

Item	Description
1	<i>Control Module1</i> is the highest level composite template and is called a control module because it provides both the Control and Supervision functionalities that are required to manage the object that is modeled by the template.
2	<i>Control1</i> is a Control composite template because it contains Control facet templates. It is also called a composite reference as it is referenced inside the composite template <i>Control Module1</i> . In the context of <i>Control Module1</i> , <i>Control1</i> is an element of <i>Control Module1</i>
3	<i>Service1</i> is a composite reference. In the context of composite reference <i>Control1</i> , <i>Service1</i> is an element of <i>Control1</i> . The same applies to composite reference <i>Service2</i> . Both <i>Service1</i> and <i>Service2</i> are Control composite templates since they contain only Control facets.
4	<i>Logic</i> and <i>Signal1</i> are Control facet references. In the context of the composite reference <i>Service1</i> , they are elements of <i>Service1</i> .
5	<i>Signal3</i> is a Control facet reference. In the context of the control module <i>Control Module1</i> , <i>Signal3</i> is an element of <i>Control Module1</i> .
6	<i>Supervision1</i> is a Supervision composite reference because it is contained in composite template <i>Control Module1</i> . It contains Supervision elements (composite references <i>Tag</i> and <i>Genie</i>). In the context of the control module <i>Control Module1</i> , <i>Supervision1</i> is an element of <i>Control Module1</i> .
7	<i>Tag</i> is a Supervision composite reference because it is contained in composite <i>Supervision1</i> and contains only a Supervision element (<i>OnOff</i>). Likewise <i>Genie</i> is a Supervision composite reference.
8, 9	<i>OnOff</i> and <i>Pump</i> are Supervision facet references and elements of templates <i>Tag</i> and <i>Genie</i> respectively.

Types of Templates

Templates are used to model the applications and the topology of the system. As a result, the following types of templates exist:

- Application templates:** Facet and composite templates that define the functional components of the applications of the system.

- *Topological templates*: Facet and composite templates that define hardware and software components of the topology of the system.

Application Templates

Overview

Control and Supervision facet templates encapsulate the respective Participant constituents:

- For Control:
 - **Logic** facet templates: They implement a specific functionality in the Control project. For example, discrete motor management by means of variables and/or function blocks.
- For Supervision:
 - **Data** facet templates: They allow creating tags and user interface messages in the Supervision project, which are required for runtime data acquisition.
 - **Genie** facet templates: They allow visualizing runtime data and interacting with process objects by adding an animated graphic to a Supervision page.

Based on a combination of these facet templates, application control module templates are created, which allow you to model the Control and Supervision services of the applications of a system.

The templates are available in the form of Libraries and once installed, they are displayed in the **Global Templates Explorer**, page 992 where they are grouped by category.

For details, refer to the EcoStruxure Process Expert help.

NOTE: A relation exists between Control and Supervision facet templates where, for a given Control service (for example, hours of operation counter for maintenance) that is provided by a **Logic** facet, the corresponding **Data** facet exists, which creates the necessary tags (for example, to acquire the counter value and allow resetting it from the faceplate of the animated graphic representing the process object). The relation is created by interface links.

Topological Templates

Introduction

Topological templates allow you to model the hardware and software infrastructure of the topology of a system by creating topological entities.

The templates are located in the Foundation Library and grouped by category.

Sub-Types of Topological Templates

The following subtypes of topological templates exist:

- **Network:** Models control and fieldbus networks.
For example: EtherNet/IP, Modbus TCP, CANopen, PROFIBUS DP, RIO S908, RIO Ethernet, device I/O for controllers and Advantys islands).
- **Network devices:** Models the field bus devices.
For example: Gateways.
- **Station node:** This template models the components of the workstations that are connected to the logical network.
For example: Network Interface Cards (NICs), Supervision software, OPC Factory Server (OFS) software, and Control software.
- **Controller:** Models the controller hardware that is managed by the Control Participant.
For example: Hardware of the M580 platform.
- **Device I/O:** Models devices and distributed I/Os including third party devices that can be connected as nodes to logical networks.
For example: Advantys Island I/Os, energy devices, motor starters, and instrumentation devices.

Interface Models

Introduction

Interface models allow you to define the following relations:

- Data exchange between two object instances or two object references.
- Dependencies between two object instances or two object references, such as compatibility or requirement rules.

Types of Interface Models

The following types of interface models are predefined:

- **Physical:** Allows data exchange at the hardware module level between:
 - Topological instances
 - Topological references
- **Communication:** Allows data exchange at the logical level (I/O scanner, OPC Factory Server software, Supervision I/O devices) between:
 - Topological instances
 - Topological references
- **Application:** Allows data exchange between:
 - Application instances/references
 - Topological instances/references
- **Mapping:** Allows data exchange during the mapping stage between application objects (modeling the logical projection of the hardware objects, such as signals or ports) and topological objects (modeling the physical hardware defined in the topology, such as I/O or communication modules).

Instances of Templates

From Templates to Instances

EcoStruxure Process Expert allows you to create application and topological instances from the corresponding templates.

You can customize an instance to fulfill specific requirements because control module templates integrate a range of Control and Supervision services, editable parameters, and expose a set of interfaces.

The process of creating instances from templates and configuring them is called instantiation.

To access and configure their parameters, instances generate facets that are created from the facet templates that each control module template contains.

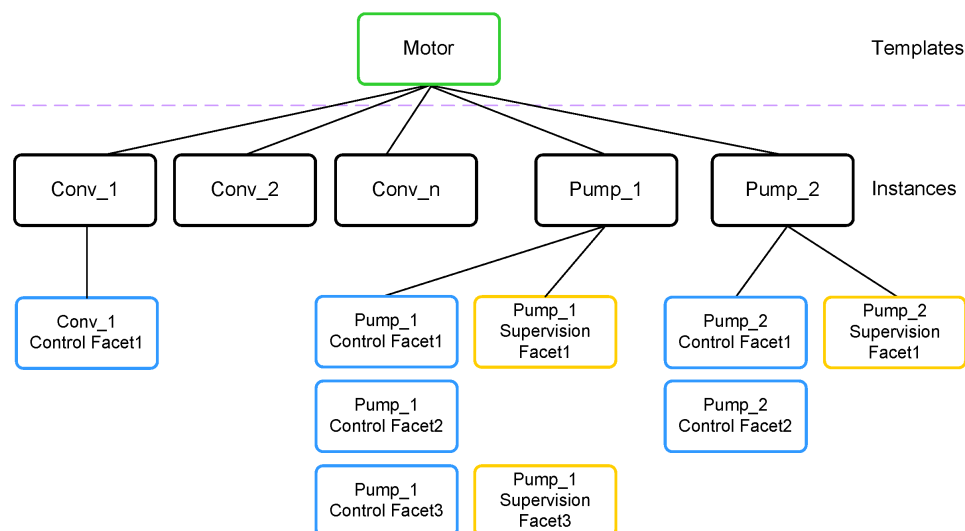
During the instantiation, you can do the following:

- Select the services that you want to enable among those provided by the template. For Schneider Electric templates, some services are mandatory (for example, **Logic**) and some are optional (for example, **Maintenance** or **Supervision**).
- Edit the values of parameters (for example, provide a high limit value for an analog level sensor or the name of a variable).
- Link instances to each other (for example, link a motor instance to the instance representing the functions of the variable speed drive that controls the motor).

If you enable (select) a service, EcoStruxure Process Expert generates a facet that is created from the facet template that provides the service.

Conversely, if you disable (clear) a service, the corresponding facet is not created or deleted.

The following figure shows an example of several instances with different configurations that you can create from one template.



Only some facets of instances *Conv_1*, *Pump_1*, and *Pump_2* are shown for illustration purposes.

System Engineering Life Cycle Concept

The system engineering life cycle consists of the various stages that you must complete to create and commission an EcoStruxure Process Expert system.

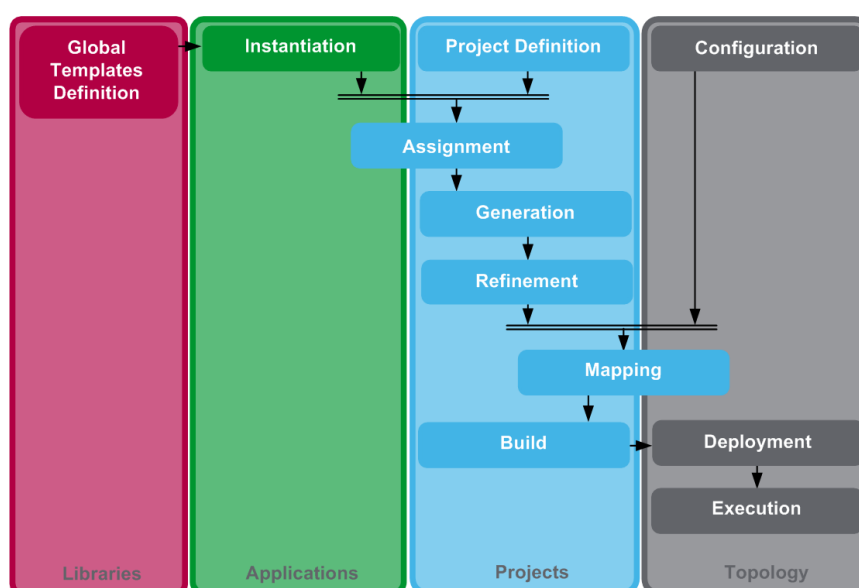
This section contains the following information on each stage:

- A stage diagram showing the position of the stage within the system engineering life cycle.
- A description of the purpose of the stage.
- A simple example of what is accomplished during the stage.

Overview

Stage Diagram

The following diagram shows the stages of the system engineering life cycle.



The stages are grouped into four domains where Applications, Projects, and Topology represent the components of a system, page 20. Libraries represent the Global Templates that are used in the other domains. An Explorer, page 23 is associated to each component and provides the functionality to complete the corresponding stages.

The arrows represent the order in which you must complete the stages. The horizontal double bars represent the stage gates where the previous stages must be completed before you can proceed further in the life cycle.

The Assignment and Mapping stages make the link between the different components of the system.

You can find a list of the stages and the main tasks that you can complete in each stage in the *System Creation Workflow and Tasks* sections, page 125.

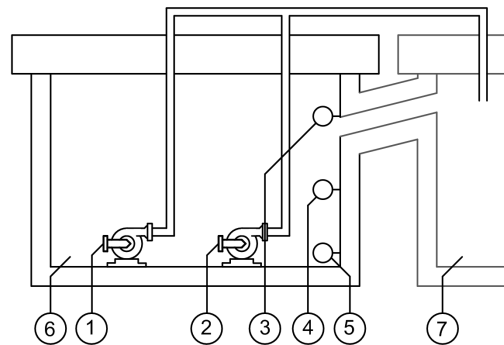
NOTE: Starting with EcoStruxure Process Expert 2021, a simplified system engineering life cycle, page 69 exists that allows you to perform changes and maintenance on an existing system faster.

Illustrating the Concept

This example is a water tank application that collects rain water. The installation consists of the following equipment:

- Two electrical pumps, which transfer the water from the tank into a separate reservoir.
- Three digital switches, which indicate the rain water level in the tank.

The following figure shows the water tank application:



Item	Description
1	Pump: Pmp1
2	Pump: Pmp2
3	High-level switch: Lsh1
4	Low-level switch: Lsl1
5	Very low-level switch: Lsll1
6	Rain water tank
7	Reservoir

Pump Management Operating Mode

In the example, one of the two pumps transfers the rain water into the reservoir. The other pump serves as a back-up pump in case the transfer pump requires maintenance or is out of order. Both pumps exchange their roles periodically and are equipped with a soft starter.

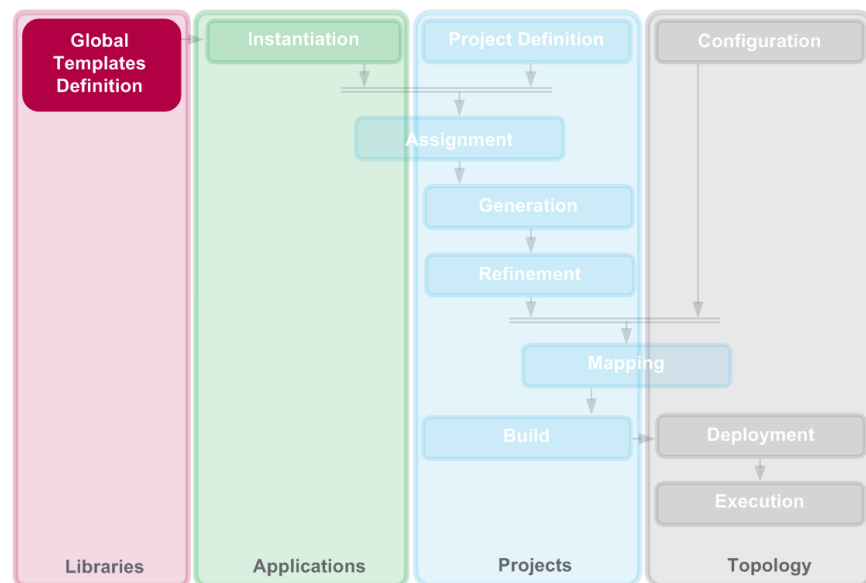
Level Management Operating Mode

In the example, the three digital level switches act on the pumps at three different water levels: Very low, low, and high level. The *Lsh1* high level and *Lsl1* low-level switches provide information on the level of the water in the tank. The *Lsll1* very low-level switch is directly wired to the power supply of both pumps so that they do not run dry.

Global Templates Definition Stage

Stage Diagram

The following figure shows the position of the Global Templates Definition stage within the system engineering life cycle.



Definition

The Schneider Electric application and topological templates that are provided in various Libraries allow you to engineer systems. Yet, you can create or modify templates.

The Global Templates definition is the process of creating and/or modifying templates either:

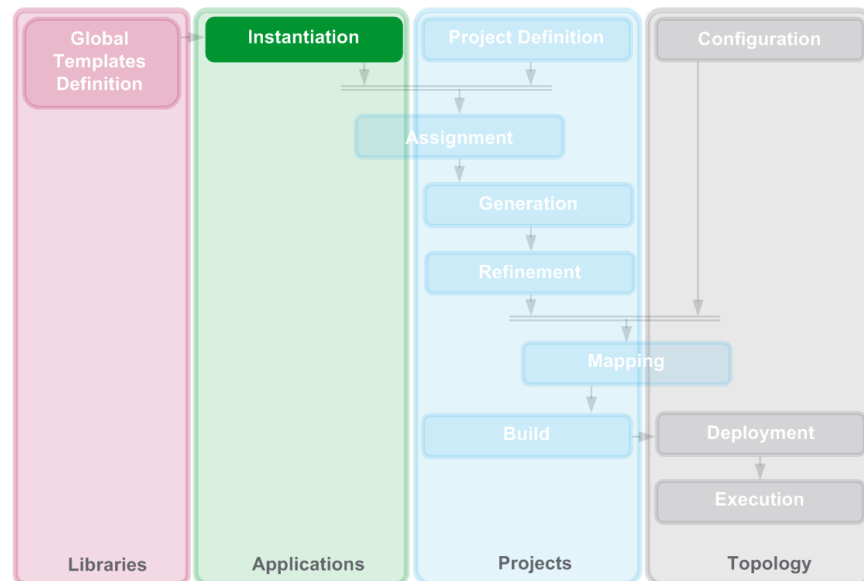
- By creating new templates from generic base elements with the help of template editors.
- By using existing templates from the Global Templates Library and modifying them to fulfill the requirements of your applications.

This process is described in *EcoStruxure Process Expert, Global Templates, Reference Manual*.

Instantiation Stage

Stage Diagram

The following figure shows the position of the Instantiation stage within the system engineering life cycle.



Definition

Instantiation consists of creating the applications of the system by modeling the following for each application:

- The process hierarchy by using a folder structure.
- The functional components of the process by using instances, page 45 of application templates.

Once you have created a [system](#), page 134, you can create instances from templates that you select among a choice of templates available in the Global Templates Library.

Depending on the composition of the template, it creates a set of Control and/or Supervision facets that are associated to the instance and provide specific services.

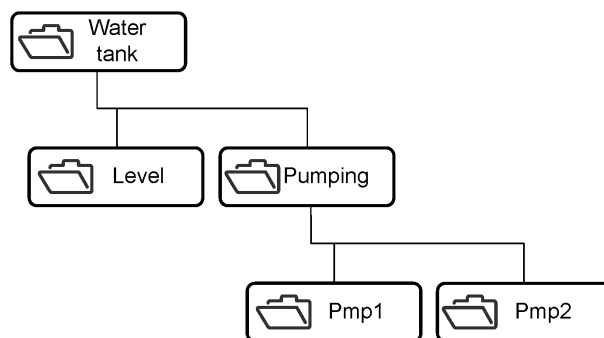
To fulfill the requirements of your applications, the instantiation process includes the step of configuring the application folders and instances by performing the following tasks:

- Selecting/deselecting optional Control and Supervision facets among those that are built into the instance template to enable/disable the corresponding services.
- Editing folder and instance properties and/or parameters.
- Creating links between instances.

For more information on completing the instantiation stage, refer to the chapter describing its implementation, page 157.

Example of the Process Hierarchy of the Water Tank Application

The following figure shows an example of how you can organize the water tank application, page 46.



Water tank: Water tank system root folder.

Level: Level management folder, which will contain the instances modeling the water level sensor functionality.

Pumping: Pump management folder.

Pmp1: Folder containing the instances modeling the functionality of the motor of pump 1.

Pmp2: Folder containing the instances modeling the functionality of the motor of pump 2.

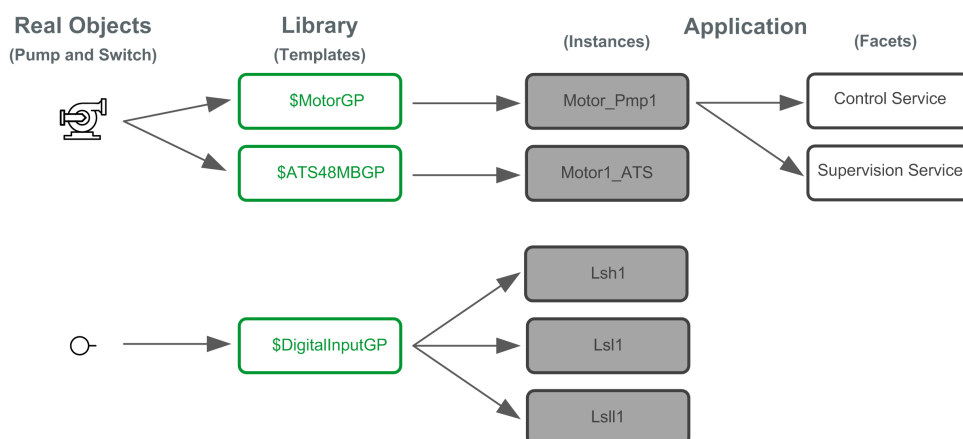
Selecting Templates to Model the Application

The following table shows examples of templates of the General Purpose Library that can be used to model the water tank functionality.

Template	Functionality
<i>\$MotorGP</i>	Motors of pump 1 and pump 2
<i>\$ATS48MBGP</i>	Altistart 48 Soft Starters for pump 1 and pump 2
<i>\$DigitalInputGP</i>	Digital switches (<i>Lsh1</i> , <i>Lsl1</i> , <i>Lsl1</i>)

Example of the Instances Modeling the Water Tank Application

The following figure illustrates the concept of instantiation for the water tank application.



NOTE: For simplicity, only the following items are represented:

- Pump 1 and level switch objects.
- Instances inside folders **Pmp1** and **Level**.
- Facets of the *Motor_Pmp1* instance.

The table describes the instances that are created during instantiation of the templates and gives an example of the configuration possibilities.

Selected template	Instances created from template ⁽¹⁾	Instance description	Purpose of the facet contained in the template ⁽²⁾	Configuration example ⁽³⁾
\$MotorGP	Motor_Pmp1	Model the electrical motors of pump 1 and pump 2.	Control facet for on/off logic.	The on/off logic facet is selected by default and cannot be deselected. The corresponding service is provided for the instance. You can configure various parameters for this Control service.
			Control facet to manage interlock conditions.	The optional interlock condition summary facet is selected, created, and the service is be available for the instance.
	Motor_Pmp2		Supervision facet to manage runtime motor on/off data.	The optional Supervision facets are selected, created, and the corresponding services are available for the instance.
			Supervision facet to manage interlock data.	You can configure several parameters for these Supervision services.
\$ATS48MBGP	Motor1_ATS	Model the management of motors for pump 1 and pump 2, each by an ATS 48 soft starter device.	Not detailed for the example.	
	Motor2_ATS			
\$DigitalInputGP	Lsh1	Models a digital input for the Lsh1 high-level detection switch.	Control facet to manage the logic.	The logic facet is selected by default and cannot be deselected. The corresponding service is provided for the instance. You can configure several parameters for this Control service.
			Supervision facet for signal data.	The optional Supervision facet is selected, created, and the corresponding service are available for the instance. You can configure several parameters for this Supervision service.
	Lsl1	Models a digital input for the Lsl1 low-level detection switch.	Control facet. (Same as for Lsh1)	The logic facet is selected by default and cannot be deselected. The corresponding service is provided for the instance. You can configure several parameters for this Control service.
			Supervision facet. (Same as for Lsh1)	The optional Supervision facet is selected, created, and the corresponding service is available for the instance. You can configure several parameters for this Supervision service.
	Lsl1	Models a digital input for the Lsl1 low-low-level detection switch.	Same as for Lsh1	Not detailed for the example.

(1) The name of the instance is user-defined.

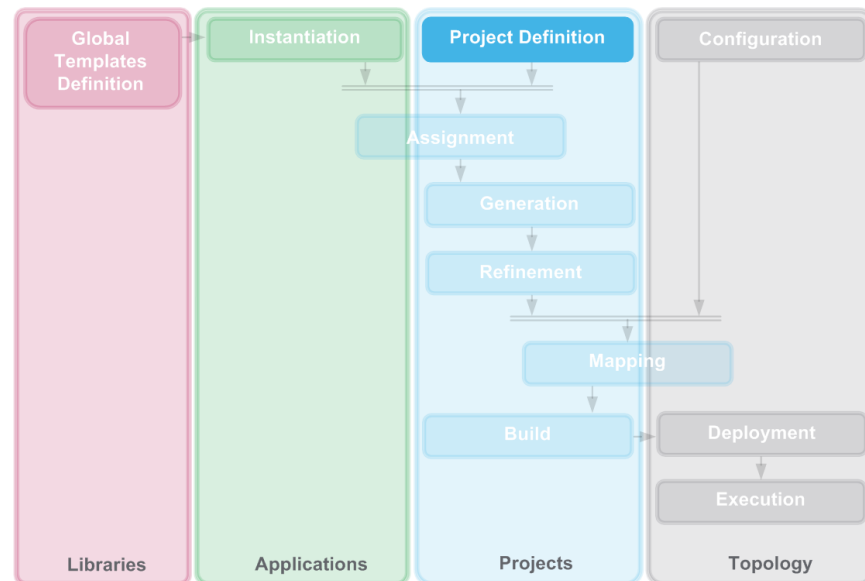
(2) The templates contain more facet templates than shown in the table.

(3) The facets are created for each instance. The example shows only the configuration for instances *Motor_Pmp1* and *Lsh1*. In fact, each instance of a template can be configured individually.

Project Definition Stage

Stage Map

The following figure shows the position of the Project Definition stage within the system engineering life cycle.



Definition

Project definition is the process of creating the functional structure of projects based on the application they are associated to. You define projects independently from specific hardware components, which allows you to use one project with different hardware infrastructures.

The content of projects is created and managed with the help of the Participants in subsequent stages. Towards the end of the system engineering life cycle, one or more executables are associated to projects, which you can deploy to the topology of the system.

For each system, you can define as many projects as required.

Each project is associated to a Participant and is therefore referred to as a Participant project.

A system is composed of the following Participant projects:

Control	This type of project is specific to a hardware platform (for example, Modicon M580 platform) and manages the controller program.
Supervision	This type of project allows you to manage data from the applications, represent it graphically, and provide the operator with a user interface to interact with the controller program.

NOTE: Supervision projects are optional and you create them depending on your requirements.

Project Structure

The organizational structure of a project is specific to the Participant that it is related to because it reflects the components that the Participant manages.

A Control project consists mainly of nodes representing:

- The program structure: A **Containers** node containing FBD and non-FBD program sections, which are referred to generically as containers.

- The program: An **Executables** node providing access to program files and related services.

A Supervision project consists mainly of nodes representing:

- Data: A **Containers** node containing tag containers for each cluster.
- Graphic elements: A **Pages** node that contains the user interface visible to the operator.
- Components: A **Cluster** node for managing servers and communication for data acquisition and representation by cluster.
- The program: An **Executables** node providing access to services related to the program files.

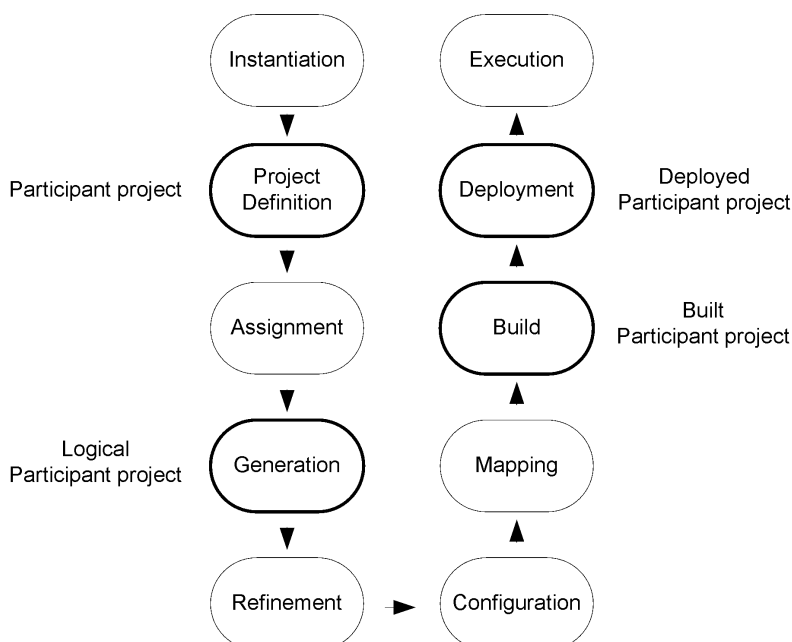
Types of Projects

As you progress through the system engineering life cycle, a Participant project evolves. To distinguish between the stages that projects go through, specific terminology is used to describe them.

EcoStruxure Process Expert manages three types of Participant projects throughout the system engineering life cycle.

Logical Participant projects	Contain the information of the applications and the organizational structure of the project with no relation to the hardware infrastructure of the system. EcoStruxure Process Expert creates these projects during the generation, page 58 stage.
Built Participant projects	Contain the information of the logical Participant project and associate it to the hardware infrastructure of the system. EcoStruxure Process Expert creates these projects during the build, page 65 stage.
Deployed Participant projects	Are the built Participant projects that you have deployed to engines of the system (for example, controllers or workstations). You deploy projects during the deployment stage, page 67.

The following figure shows at which stages of the system engineering life cycle the different types of projects are managed.



For more information on completing the project definition stage, refer to the chapter describing its implementation, page 262.

Example

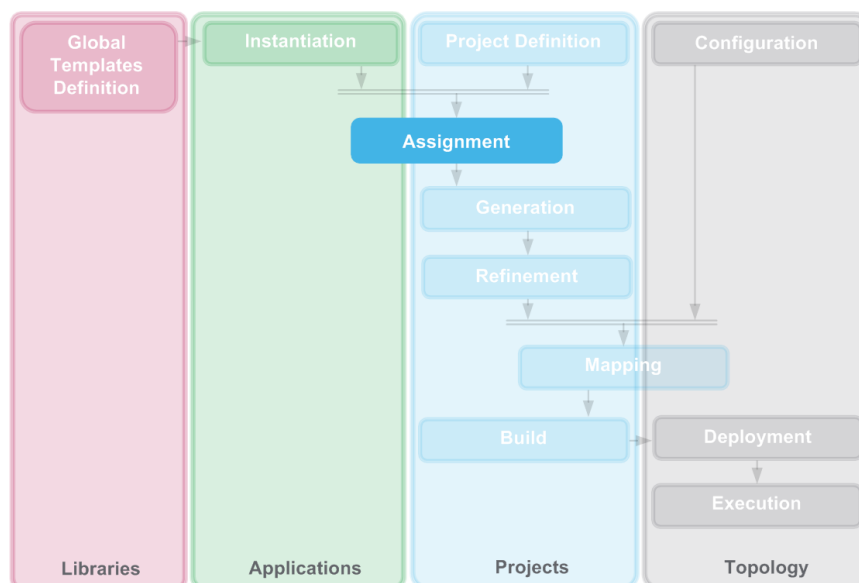
The following projects are defined for the water tank application, page 46:

- A Control project to manage the logic (for example, variables and DFBs,) of the instances of the application. It contains three containers in the MAST task folder corresponding to the folder structure of the application:
 - Level
 - Pmp1
 - Pmp2
- A Supervision project to manage the Supervision services (tags, alarms, supervision pages) of the instances of the application. It contains the following components:
 - One tag container for the Supervision data facets of the instances of the application.
 - One page for the Supervision genie facets of the instances of the application.
 - Components: One of each I/O, alarm, and trend server, and I/O device with its communication port.

Assignment Stage

Stage Map

The following figure shows the position of the Assignment stage within the system engineering life cycle.



Definition

Assignment is the process of associating Control and Supervision facets of instances of an application to containers of the corresponding projects of the system.

The purpose is to define where the constituents that are encapsulated in the facets will be generated in the Participant project during subsequent stages of the system engineering life cycle.

You must assign the two types of facets to the corresponding containers:

- Control facets: To containers of Control projects, which are FBD sections.
- Supervision data facets: To containers of Supervision projects, which are tag containers.

You can assign a facet only to one container per project but you can assign a facet to several projects.

NOTE: Supervision genie facets are assigned to a page of a Supervision project during the refinement stage.

For more information on completing the assignment stage, refer to the chapter describing its implementation, page 340.

Example

The table shows an example of assignment of the facets of the instances *Motor_Pmp1* and *Lsh1* of the water tank application, page 46 that you can perform.

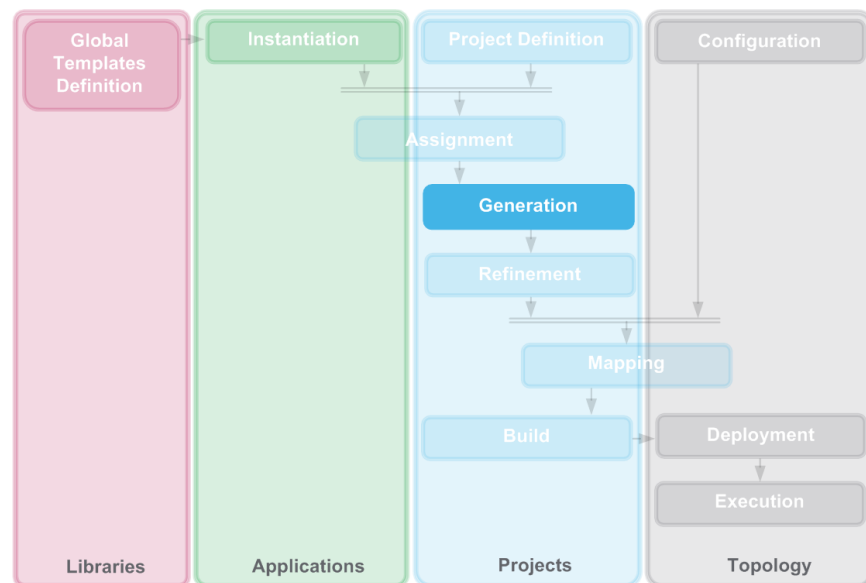
Instance	Facets	Assigned To
<i>Motor_Pmp1</i>	Control facets: <ul style="list-style-type: none"> • <i>\$MOTORGP_UL</i>: On/off logic data facet • <i>\$INTERLOCKONGP_UL</i>: Optional interlock condition summary facet 	FBD section <i>Pmp1</i> of the Control project.

Instance	Facets	Assigned To
	Supervision facet: <ul style="list-style-type: none">• <i>\$MOTORGP_CD</i>: Optional on/off data facet	Tag container of the Supervision project.
<i>Lsh1</i>	Control facets: <ul style="list-style-type: none">• <i>\$MOTORGP_UL</i>: Logic data facet• <i>\$DISignal_UL</i>: Optional digital input signal	FBD section <i>Level</i> of the Control project.
	Supervision facet: <ul style="list-style-type: none">• <i>\$DINPUTGP_CD</i>: Optional data facet	Tag container of the Supervision project.

Generation Stage

Stage Map

The following figure shows the position of the Generation stage within the system engineering life cycle.



Definition

Generation is an automatic process that creates a logical Participant project, page 54 by using the constituents of the assigned facets and taking into account the structure of the project. At this point, the logical Participant project has no relation with the topology of the system.

For more information on completing the generation stage, refer to the chapter describing its implementation, page 396.

Example

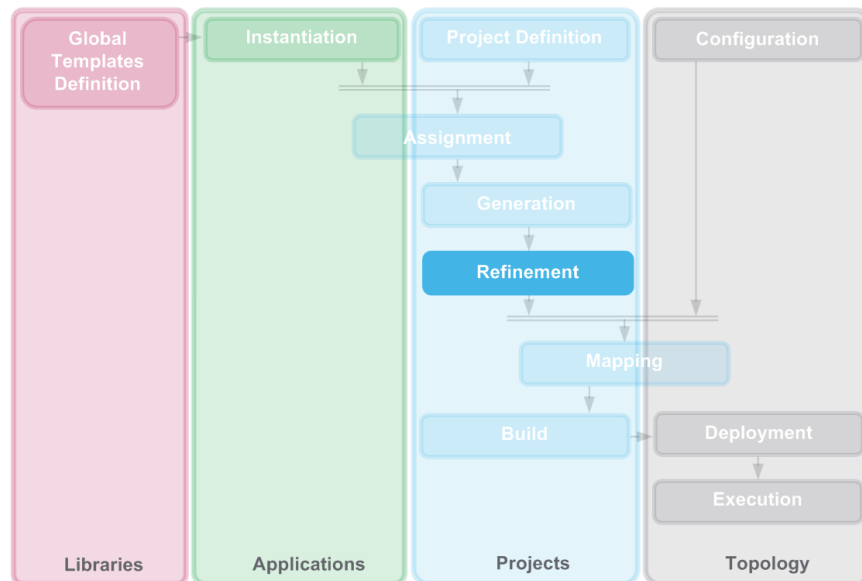
During the generation of the facets of the water tank application, page 46 that are assigned to projects, EcoStruxure Process Expert creates the following:

- For Control: The types, variables, and program in the Control Participant.
- For Supervision: Populates the different database files that are used by the Supervision Participant.

Refinement Stage

Stage Map

The following figure shows the position of the Refinement stage within the system engineering life cycle.



Definition

Refinement is an optional process during which you can modify and/or add information to a logical Participant project, page 54 that EcoStruxure Process Expert created during the generation stage.

For example, you can refine a Control project to:

- Add logic in existing sections.
- Create new relationships with other constituents such as interlocks or control sequences.
- Add sections in languages supported by the Participant other than FBD.

To facilitate the refinement of Control projects, EcoStruxure Process Expert locks and identifies visually code that it has generated.

You can refine a Supervision project, for example, to:

- Assign genie facets to a Supervision page.
- Configure user privileges.

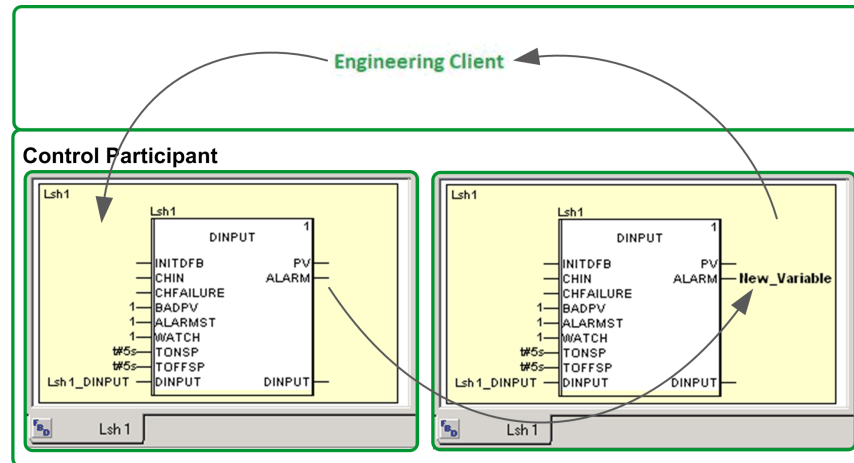
The respective Participant opens allowing you to use the interface and the necessary functionality of the embedded software.

For more information on completing the refinement stage, refer to the chapter describing its implementation, page 445.

Example

During the refinement of the Control project of the water tank application, page 46, you can access and modify the control logic by using the Control Participant.

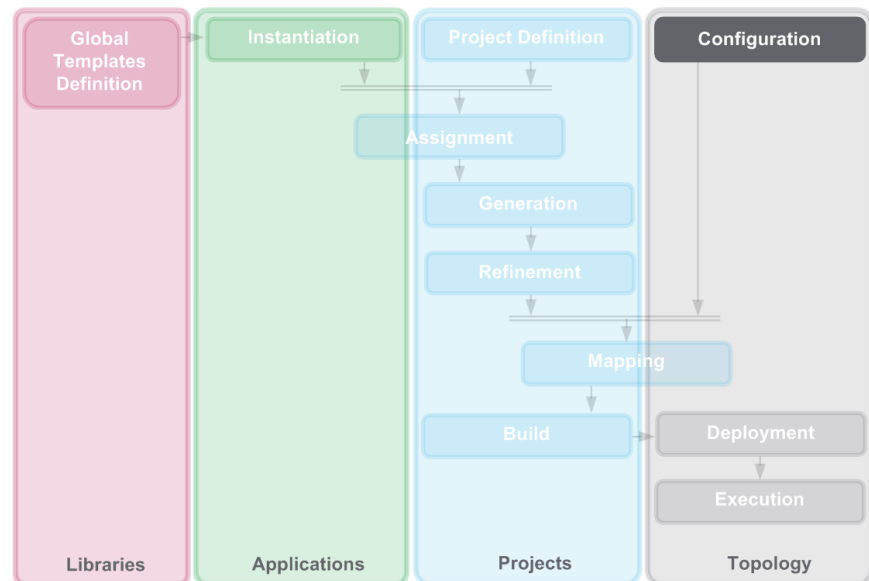
The following figure illustrates the refinement of the constituents generated by the Control logic facet of the *Lsh1* high-level switch sensor instance. During this refinement, a new variable is added.



Configuration Stage

Stage Map

The following figure shows the position of the Configuration stage within the system engineering life cycle.



Definition

Configuration consists in defining the topology of the system, which models the hardware and software infrastructure.

You can create topological entities by using either of two methods depending on the entity that you are defining:

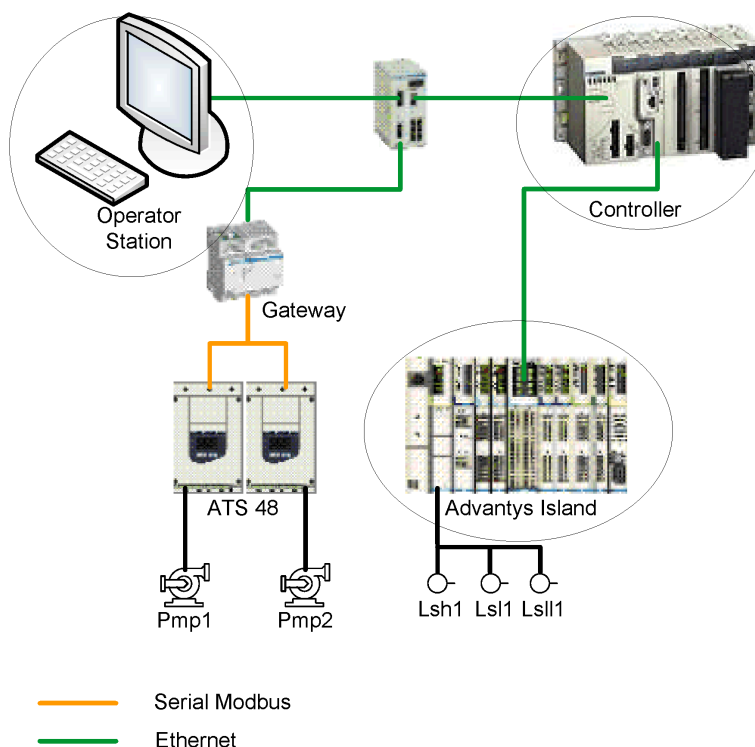
- In the engineering client user interface by selecting manually the templates to be used to create entities.
- In the Control Participant by using the functionality of the embedded software. Saving the configuration in the Participant creates the corresponding folders and devices in the **Topology Explorer** by selecting automatically the appropriate templates of the Foundation Library.

For more information on completing the configuration stage, refer to the chapter describing its implementation, page 512.

Example

For the water tank application, page 46, during the configuration stage, you can define the hardware that will run the program, the Ethernet control network, and the operator station from which the operator will operate the pump, monitor the pumping process, and the water level.

The following figure shows examples of topological entities of the water tank system that you can create during the configuration stage.



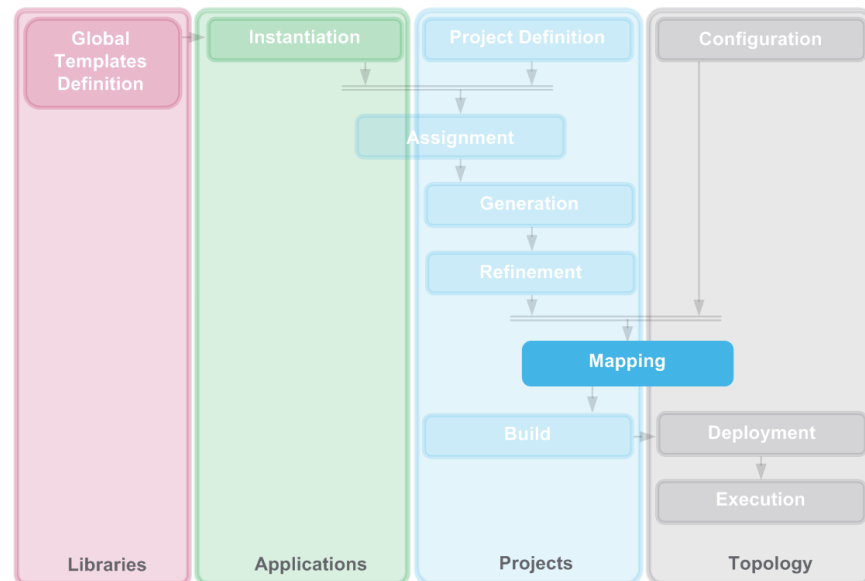
Controller	Local rack with an Ethernet based architecture.
STB Island	Distributed I/Os to which level switches <i>Lsh1</i> , <i>Ls11</i> , and <i>Ls111</i> are connected, communicating with the controller through I/O scanning.
Operator Station	Supervision station to control and monitor the system, configured with its network interface card and the supervision software.
Ethernet network	Logical networks to which the controller, the STB Island, the gateway, and the workstation are connected.
Gateway	The gateway communication instance is created in the application of the system modeling the communication functions. The device may not be represented in the topology.
ATS 48	The soft starter instances are created in the application of the system modeling the motor management functions. The device itself may not be represented in the topology.

NOTE: When communication with devices is performed by using Modbus TCP explicit or implicit messaging, topological instances of devices are created also in the topology to manage TCP/IP addressing.

Mapping Stage

Stage Map

The following figure shows the position of the Mapping stage within the system engineering life cycle.



Definition

Mapping consists in associating facets of a project to resources of the topology of the system.

You can perform three types of mappings independently for each project:

- **Service mapping:** Defines which engine executes the project (for example, which controller runs the program or to which workstation with an I/O server is the Supervision project deployed).
- **Hardware mapping:** For Control projects only. Links the logical projection of the hardware defined in the application (for example, a variable representing an I/O signal of a motor) to the hardware defined in the topology (for example, an I/O channel in an I/O module of the controller).
- **Communication mapping:** Defines runtime communication (for example, which controller will communicate with the I/O device and through which communication module), communication between controllers (peer to peer communication), and between controllers and devices (for example, communication with STB islands or Modbus TCP Ethernet devices by using I/O scanning).

NOTE: You can map a project to several different engines, for example:

- To a controller in the laboratory for test purposes.
- To a controller in the production plant.

For more information on completing the mapping stage, refer to the chapter describing its implementation, page 719.

Example

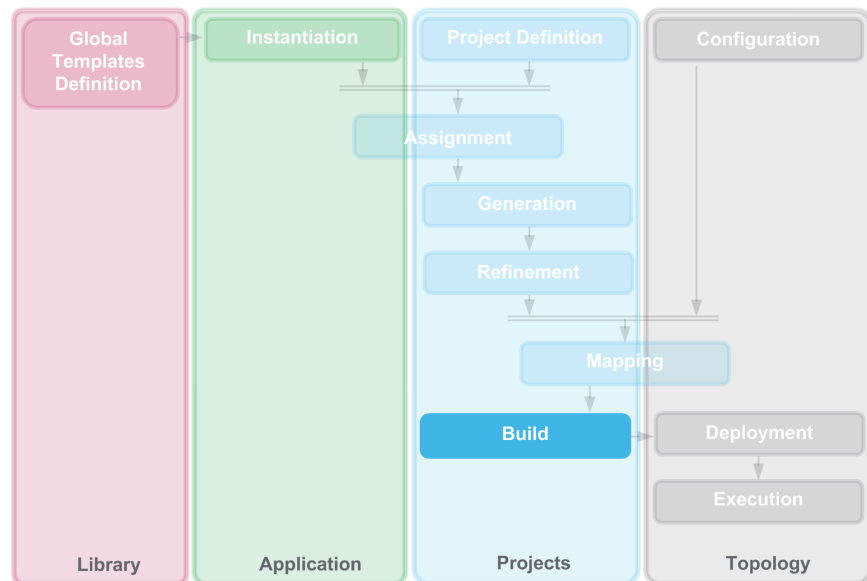
The following table gives examples of the mappings that you can perform for the water tank application, page 46.

Mapping	Description
Control project	
Service	Defines to controller the program is associated for deployment and execution.
Hardware	<p>If for instance <i>Motor_Pmp1</i>, you select the optional Running Control service, the corresponding facet is created. It creates, for example, a boolean variable that indicates the state of the motor switch.</p> <p>You can map this variable of the application to the channel of an input module of the STB island.</p>
Communication	Defines the I/O scanner channels that are created for communication between the controller and the STB island.
Supervision project	
Service	Defines the I/O, alarm, and trend servers that are used for the Supervision project and on which operation stations of the topology they are installed.
Communication	Defines from which controller I/O data is acquired and which communication module of this controller is used.

Build Stage

Stage Map

The following figure shows the position of the Build stage within the system engineering life cycle.



Definition

A build consists in starting an automated process during which a Participant merges the following information in order to create the built Participant project, page 54:

- The logical Participant project
- Mappings
- The topology

The built Participant project is the executable file that you can deploy to the associated engine of the system.

EcoStruxure Process Expert stores the executable files resulting of this process in the content repository.

At this point, you can view the complete Control and Supervision projects in the corresponding Participant.

For more information on completing the build stage, refer to the chapter describing its implementation, page 773.

Example

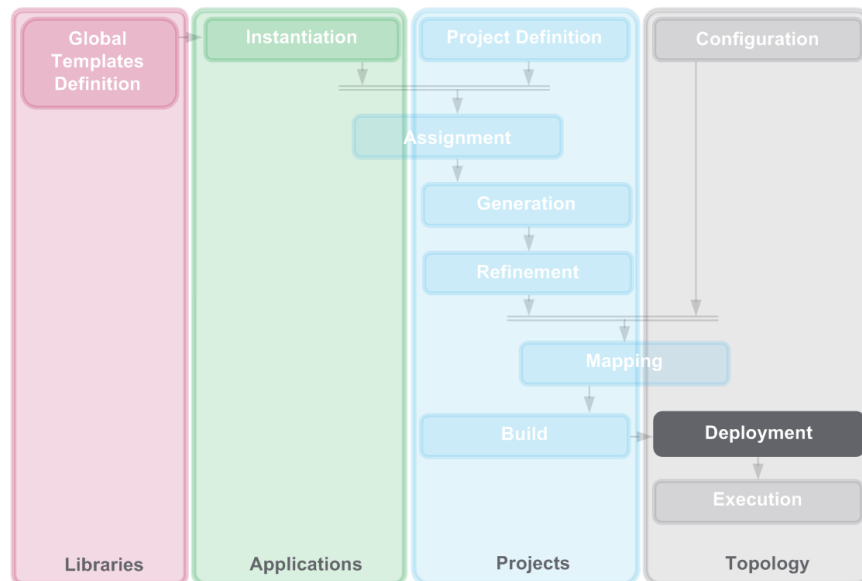
During the build process of the water tank application, page 46, the Participants create the following files.

Participant	Generated files
Control	The executable file, which is the logic to be deployed to the controller.
Supervision	The master project file (.ctz) to be deployed to the operation server and operator stations.
	Files of the included projects (.ctz) to be deployed to the operation server and operator stations.
	The OFSOPC/OPC UA configuration files (.xml) to be deployed to the operation server.

Deployment Stage

Stage Map

The following figure shows the position of the Deployment stage within the system engineering life cycle.



Definition

Deployment consists in transferring a built Participant project, page 65 to the engine that exists in the topology of the system and that is mapped to the executable.

Depending on the project, an engine can be:

- For Control:
 - A controller
 - A workstation running the Controller simulator
- For Supervision: A workstation acting as operation server or operator station.

Once deployed, a project is referred to as the **deployed Participant project**, page 54, which is the project that is loaded into the engine for execution.

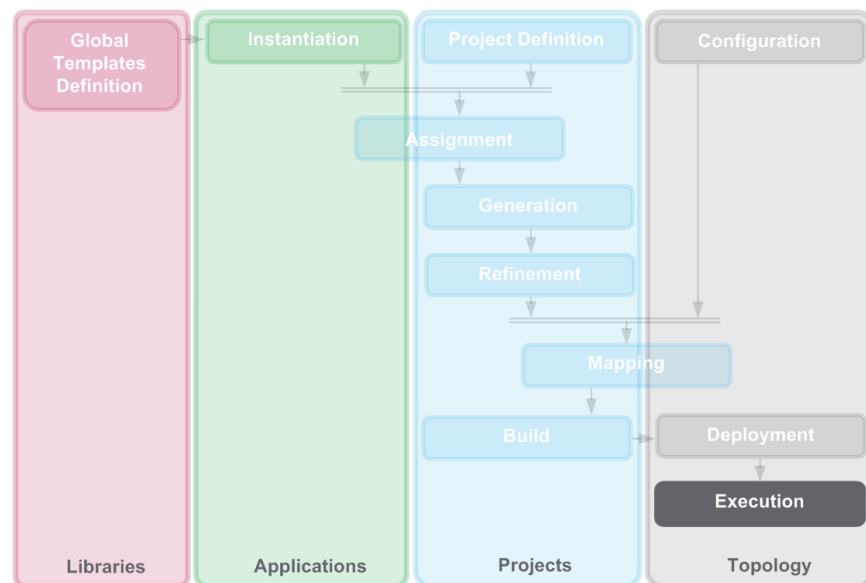
The stage also includes the steps of redeploying an already deployed Participant project and to update a deployed Participant project with changes that you made to the corresponding logical Participant project.

For more information on completing the deployment stage, refer to the chapter describing its implementation, page 793.

Execution Stage

Stage Map

The following figure shows the position of the Execution stage within the system engineering life cycle.



Definition

Execution includes the following steps:

- Starting and stopping the execution of a deployed Participant project, page 54.
- Making online changes to a deployed Participant project.
- Updating the corresponding logical Participant project, page 54 with the changes that you have made online.

The execution stage takes into account actions that you can perform with and without stopping the controller to keep you informed of the impact of your changes.

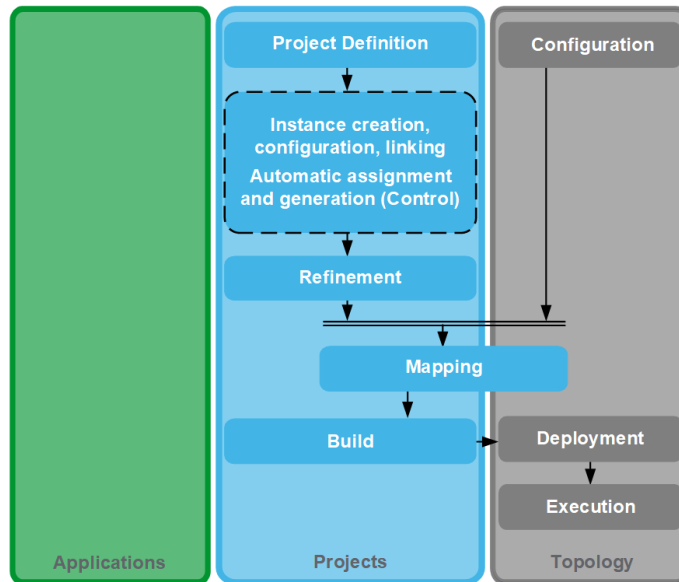
For more information on completing the execution stage, refer to the chapter describing its implementation, page 854.

Additional runtime services are described in the *EcoStruxure Process Expert Runtime Navigation Services User Guide*.

Simplified System Engineering Life Cycle

Graphical Representation

The following figure shows the simplified system engineering life cycle. The Instantiation, Assignment, and Generation stages are grouped into one, which is completed by using the **Project Explorer** and the **Assignment Editor** for Control projects. The change compared to the regular system engineering life cycle is outlined with a dashed line.



NOTE: The Global Templates definition stage is not part of the simplified system engineering life cycle.

Definition

The purpose of the simplified workflow is to make it possible to perform small changes to the Control project of a system during the maintenance phase with little effort. Such changes can be, for example, adding one or more I/O signals.

As part of the simplified workflow, you can perform the following actions from a Control project:

- Create, page 348, configure, copy, and link, page 390 one or more instances and assign their facets automatically.
- Generate changes automatically, page 407.

Working with EcoStruxure Process Expert

What's in This Part

Starting and Quitting System Server and Clients	71
Multiusers Functionality	82
Action Traceability	84
Using the Help	94
User Interface	102

Overview

This part explains how to open and use the engineering client user interface. It describes the functionality of EcoStruxure Process Expert and contains information that helps you use it effectively.

Starting and Quitting System Server and Clients

What's in This Chapter

Logging into and out of the System Server and Clients	72
Locking and Unlocking EcoStruxure Process Expert	74
Starting the System Server Again	75
Client/Server Connection	76
Quitting EcoStruxure Process Expert	80

Overview

This chapter describes how to start and quit the system server and clients after a first start, and describes the client/server connection.

Logging into and out of the System Server and Clients

Overview

To log into and use the EcoStruxure Process Expert system server and/or a client, a user must be enabled and associated to one or more Process Expert profiles.

The users and profiles can be managed in Security Editor by the system administrator.

For example, to use the engineering client, the user must be associated to the *ESX PE Engineer* profile.

NOTE: For details, refer to the topic describing role-based access control (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

Authentication Mode

To authenticate users, the system server uses *SecurityService*, which is configured by using Security Editor. If a centralized authentication mode is selected, ensure that the remote server on which the directory service or Security Editor is installed is reachable from the system server computer.

You can configure local authentication as a fallback mode if the centralized mode cannot be used.

Single Login and Logout

EcoStruxure Process Expert implements a single-login mechanism. You need to log in when you open the first component (for example, an engineering client). Thereafter, you are logged in automatically when you open other components on the same computer.

If several components are open on the same computer, you are logged into all of them at once. You need to log in again only after you log out of a component on the computer.

EcoStruxure Process Expert also implements a single-logout mechanism. Logging out of a component also logs you out of the other components to which you are logged in on the same computer.

NOTE: Only the system server can be running without a user being logged in when it was started automatically (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

Logging In

To log into a component, enter in the login window the name and password of a user that is enabled in Security Editor. The name is case-sensitive. The login process may take a moment to complete.

The login window opens when you open the first component on a computer or by selecting **Log In** from the user menu, page 104.

If you close the login window without logging in, the component remains open.

Logging Out

To log out of a component, select the **Log Out** command from the user menu, page 104.

The operations that are in progress for components running on the computer must have completed.

Logging out of a component results in the following:

- It does not quit the component.
 - When you log out of the system server, it is not stopped. However, you cannot stop the system server when no user is logged in.
 - When you log out of clients, explorers and other tabs are closed. If you had opened additional windows by using the **Open in New Window** command, page 110, they are closed.
- It also logs you out of the other EcoStruxure Process Expert components that are open on the same computer.

NOTE: You are automatically logged out of EcoStruxure Process Expert after you quit the last component that is open on the computer.

Viewing Connected Clients

From a client that is connected to a system server, you can view information about the other clients that are connected to this server in the **Client Details** tab of the about dialog box.

For a description of the dialog box, refer to the topic describing the **About EcoStruxure Process Expert** dialog box, page 106.

Tracing Login Attempts and Log Outs

The following actions are recorded:

- Attempts to log into a component even if they are unsuccessful. The user name that is entered is recorded.
- Locking and unlocking, page 74 a component.
- Logging out of a component.

Actions performed in the system server console appear as notifications in the console and are recorded in the system server log files.

Actions performed in a client appear in the notification panel and in the system server console.

For details, refer to the topic describing the traceability of actions, page 84.

Locking and Unlocking EcoStruxure Process Expert

Locking Components

To lock a component, select the **Lock** command from the user menu, page 104.

Locking the server still allows clients running on other computers to connect or remain connected.

When components are locked, the log-in window opens on top of the main window. The username of the logged-in user shown is in the **User Name** field in read-only mode. Only the user who has locked EcoStruxure Process Expert can unlock it.

Impacts of Locking Components

Locking a component has the following impacts:

- Locking any component also locks all the other components that are open on the computer and to which you are logged in.
- Dialog boxes and notifications in the notification panel are queued and suppressed. These messages are shown once the component is unlocked.
- You cannot quit a component that is locked.
- You cannot open other EcoStruxure Process Expert components on the computer.

Unlocking Components

To unlock a component, enter the password in the login dialog box.

Unlocking any component also unlocks all the other components that were locked on the computer.

To unlock EcoStruxure Process Expert when **Domain** is the configured **Authentication Type** requires that the domain controller is reachable from the computer running the system server.

Starting the System Server Again

Temporary Content Repository Files

When you start the system server, it creates folders to store temporary files, page 1124 from the content repository to speed up access to these files.

Using the Auto-Start Functionality

You can configure the system server to start automatically (see *EcoStruxure Process Expert, Installation and Configuration Guide*) when you restart the computer.

Starting the System Server Manually

To start the system server manually after you stopped it, page 80, proceed as follows.

Step	Action
1	Ensure that InterSystems Caché is running by verifying the presence and the color of the Caché icon in the notification area. When Caché is running, its icon is blue.
2	Ensure the engineering and operation clients that were connected are closed.
3	If the system server tray icon is not visible, from the Windows Start menu, click EcoStruxure Process Expert > System Server . Result: The log-in window opens.
4	Enter your username without domain or prefix, your password, and click Log In . Result: The system server console opens.
5	Click Action > Start . Result: The system server starts. NOTE: During the system server start, several Participant windows may open briefly. Do not close them manually; otherwise, the functionality of EcoStruxure Process Expert may be impacted.
6	Wait until the system server tray icon turns green or the console displays the following message at the bottom of the window to start engineering and operation clients (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>): Server is ready.

Client/Server Connection

Overview

Clients connect to the system server for the following operations:

- Access the database.
- Access software Participants.
- Start tasks that are performed by the system server.
- Receive notifications from the system server.
- Verify digital certificates and license rights.

You cannot use clients while the connection to the system server is interrupted or while the system server is not running.

You can test the client/server connection (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

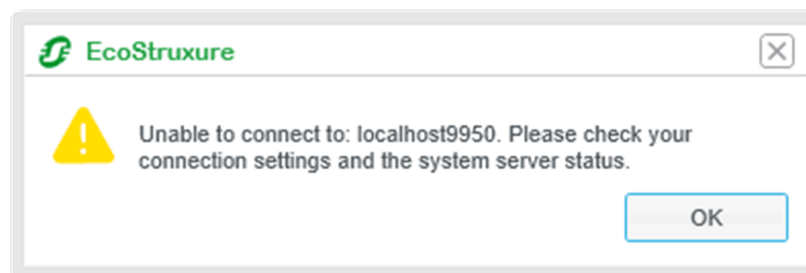
Notifications About Client/Server Connection Interruptions

The following table describes the various notifications that are displayed client-side for different scenarios where a client cannot connect to the system server or a software Participant.

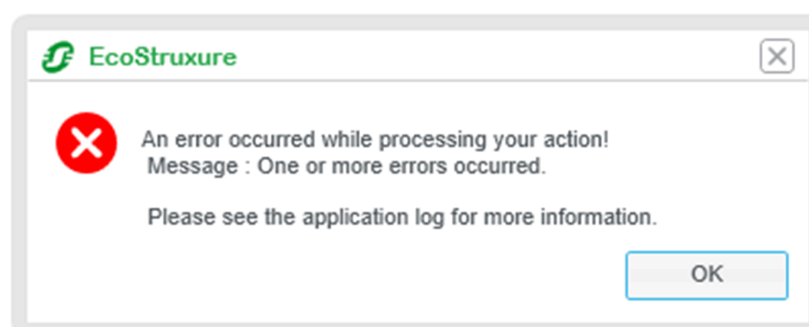
Scenario	Operation in progress at the time of the interruption	Client notification	Corrective action
1	No	No	None
2	You start a task while the connection to the system server is interrupted.	Notification that the client is unable to connect to the system server.	Restart the task when the connection is re-established.
3	Yes	Notification about an interruption of the client/server communication.	If the system server is running, the action completes server-side and the notification of clients is updated when the connection is re-established. If the system server has stopped, an action is required, page 78.
4	Yes. An action is performed in an open software Participant window.	Notification that the action in the Participant cannot be performed and indication of the possible causes. NOTE: If you are refining a Control project online (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>) and clicked the button to build and deploy changes after the connection was interrupted, the changes are not deployed nor saved.	If the connection to the system server is functional, save changes, close the Participant window, and start over once the cause has been corrected. If the connection to the system server is interrupted, refer to scenario 3. If the reason for the connection interruption cannot be determined, contact your local Schneider Electric service representative.
5	The system server is stopped by user intervention.	Notification in the notification panel, page 85 to inform that the server is shutting down.	From this point on, you cannot perform actions in the client. Close clients and restart the system server.
6	Yes. An action is performed in an open software Participant window.	None. A message is displayed in the system server console.	Contact your local Schneider Electric service representative.

NOTE: Also refer to the topic containing information to help troubleshoot client/server communication (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

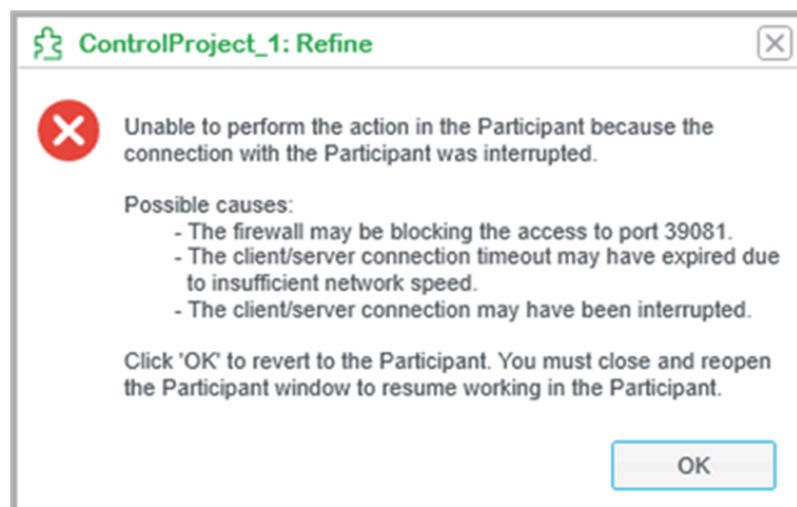
The following figure shows an example of the notification that is displayed for scenario 2. The name or IP address and the port of the system server computer are indicated.



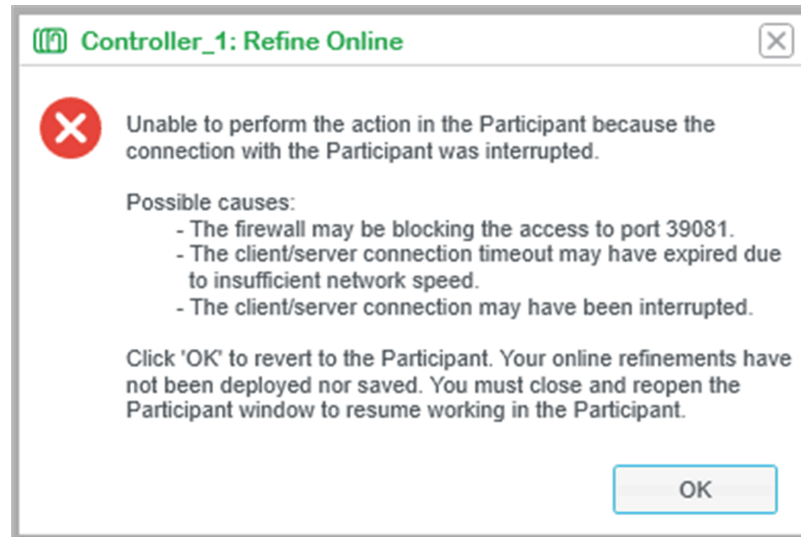
The following figure shows an example of the notification that is displayed for scenario 3.



The following figure shows an example of the notification that is displayed for scenario 4.

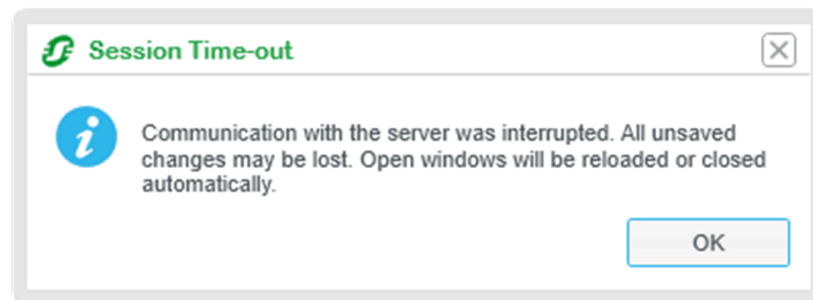


The following figure shows an example of the notification that is displayed for scenario 4 if you are refining a Control project online.



After Re-establishing the Connection to the System Server

Clients automatically reconnect to the system server when the connection is re-established. After reconnecting, clients display a notification to inform you of the communication interruption. Unsaved changes are discarded and open windows are automatically refreshed or closed.



If the system server stopped, the notification panel is empty when the system server restarts. To view the history of activity up to the interruption of the communication, open the log files, page 92.

After an interruption of the client/server connection, the states of the engineering client and the system server may be inconsistent. This can happen, for example, if at the time of the interruption an operation was in progress, such as the import of several Participant projects or the pasting of many instances.

NOTICE

LOSS OF COMMUNICATION

After an interruption of the client/server connection, ensure that the last operation executed by EcoStruxure Process Expert completed successfully and, if necessary, repeat the last operation.

Failure to follow these instructions can result in equipment damage.

Restarting Clients After Closing Unexpectedly

If an engineering client closes unexpectedly (for example, because the computer malfunctioned), you do not need to restart the system server. You can restart the

client after 60 seconds, which is the time required by the system server to terminate a client session that ended abnormally.

Restarting a client earlier may display messages about a missing engineering client license.

Quitting EcoStruxure Process Expert

Overview

To quit EcoStruxure Process Expert, close the components in the following order:

- Clients that are connected the system server that you are stopping.
- The system server.

You cannot quit a component if it is **locked**, page 74 and you cannot stop the system server when no user is logged in.

To restart the system server, quit EcoStruxure Process Expert in the same order and then, **start the system server**, page 75.

NOTE: You are logged out of EcoStruxure Process Expert on a computer after you quit the last component that you were logged into on this computer.

Quitting Clients

When you quit a client, it does not close if a task is still in progress.

To quit a client, proceed as follows.

Step	Action
1	Ensure that processes are completed by referring to the information in the notification panel.
2	Click the client close button. Result: The client closes and is disconnected from the system server.

Stopping the System Server

When you stop the system server, tasks in progress are completed before the server is stopped. If a client is connected to the system server, a notification is displayed to indicate the identity of the logged-in user by using the format **<User name> (<computer name>)**.

You cannot use a client when the communication between the system server and the client is interrupted, for example, if the system server is not running.

NOTICE

LOSS OF COMMUNICATION

Stop the system server only when no client is running.

Failure to follow these instructions can result in equipment damage.

To stop the system server, proceed as follows.

Step	Action
1	From the menu bar of the system server console, click Actions > Stop . Result: A confirmation dialog box opens.
2	If a client is connected, click No ; otherwise, proceed to step 4. Result: The dialog box closes without stopping the system server.
3	Quit the client, page 80 that is connected and restart the procedure.
4	Click Yes . Result: The system server completes on-going tasks and stops.

Multiuser Functionality

What's in This Chapter

Multiuser Functionality	82
-------------------------------	----

Multiuser Functionality

Overview

Several clients can connect to a system server simultaneously, allowing several users to work on the same component or different components of a system at the same time.

For example, User A and User B can work on the application of the system, or User A can work on the application and User B on the projects of the system. These operations could also be performed by a single user.

In both cases, EcoStruxure Process Expert manages system-wide concurrent user access to editors and other tools to preserve the integrity of data of the system.

It may allow you to view data by granting access to certain editors in read-only mode.

When a user is finished working on a system component (for example, an instance) by closing the corresponding tool (for example, the **Instance Editor**), other users are allowed to access components that manage or use data of that component.

As such, users do not need to check out or lock the component that they are working on.

Managing User Activity

At the following stages of the system engineering life cycle, EcoStruxure Process Expert indicates the status of objects by using icons:

- Instantiation, page 199
- Project Definition, page 338

At any stage, when the command that you have selected cannot be completed, you can obtain information about the reason through:

- Dialog boxes that clients display when you try to access system data that is being used or edited by another user.
- Information in the notification panel on the activity of any user that is logged in to a client of the same system server.

Examples

By using examples, the table illustrates some of the locking mechanisms that EcoStruxure Process Expert implements when two users work on the same system.

Action executed by User A	Examples of restrictions for User B
Editing properties of <i>Instance A</i> .	Cannot edit <i>Instance A</i> but can open the Instance Editor in read-only mode.
	Can generate the section to which facets of <i>Instance A</i> are assigned but the facets of

Action executed by User A	Examples of restrictions for User B
	<i>Instance A</i> are not generated. If the section contains other facets, they are generated. Information is displayed in the notification panel.
Refining the Control Participant project in the Project Explorer .	Cannot refine the same Participant project or any section of it. Information is displayed in a dialog box.
	Cannot generate the same Participant project.
	Cannot build the same Participant project.
Configuring a controller entity in the Topology Explorer .	Cannot configure the same controller entity.
	Cannot do mapping (hardware and communication) of the Control executable to which the engine is mapped.
	Cannot build the Control Participant project executable to which the engine is mapped.

NOTE: The locking mechanisms apply also when a single user is performing these operations (if applicable).

Action Traceability

What’s in This Chapter

Notification Panel	85
Audit Trail	88
Activity Log Files	92

Overview

This chapter describes how to obtain information related to the activity of the system server and clients.

Notification Panel

Overview

The notification panel allows you to view and trace engineering activity. The panel is displayed automatically when you start an engineering client. You can modify its height and show column headers.

The information that is displayed is the same for each engineering client that connects to the same system server. When you open a second engineering client, you can see in its notification panel the history of actions starting from the moment the first engineering client was opened.

Certain system server tasks are displayed also.

An entry is logged when you launch a task and it is updated as the execution of the task progresses. When the task is complete, the status of the task and the actions that EcoStruxure Process Expert executed are indicated.

For each task, a color icon allows you to see the status at a glance.

At any time, the notification The information remains available while the system server to which the client is connected is running. Closing the client or the notification panel does not delete the entries.


You can show/hide the notification panel from the engineering client toolbar, page 104.

Information Availability

At any time, the notification panel can display a maximum of 2000 entries. When this limit is reached, the most recent entries replace the earliest ones on a first-in first-out basis.

The entries are displayed while the system server to which the client is connected is running. Closing the client or the notification panel does not delete the entries.

Aborting Tasks

When you select or confirm certain commands, the  icon appears in the **Action** column of the notification panel. You can click it to cancel the task. Once the task is near completion and cannot be canceled anymore, the icon appears disabled








Only the user who has initiated the command is can abort it.

The abort functionality and the behavior when you click the icon is described in the topics documenting the commands for which the functionality is available. For example, the **Paste** command, page 214 of the **Application Explorer**.

Aborted tasks are identified in the notification panel in the **Message** column with (*Aborted*).

Data Description

The table describes the data that is displayed in the notification panel.

Column header ⁽¹⁾	Description
–	<p>Icons indicate the status of the task:</p> <ul style="list-style-type: none">  Task in progress (spinning).  Task completed successfully (green).  Task did not complete (red).  Task completed with notifications (yellow).  Information (blue).
Start Time	Date and time when the task was initiated.
Elapsed Time	The time it took to complete the task independently of its final status.
Message	<p>Short description of the task that was executed. The status of the task is indicated in brackets.</p> <p>A task can have messages for sub-tasks.</p> <p>Expand the row by clicking the expand icon on the outer left of the row.</p>
Action	Keyword describing the action that was executed. Typically, this corresponds to the command you selected.
User Name	<p>Name of the user who is logged in to the client from which the task was executed.</p> <p>Format: <Full name> (<Username>)</p>
Station Node	Name of the computer from which the task was executed.
System	<p>Identifier of the system in which the task was executed.</p> <p>When the task is not related to a system, the entry is Global.</p>
Domain	<p>Identification of the domain or project component on which the task was executed.</p> <p>When the task is not related to a system component (for example, when a user logs in), the entry is Global.</p>
Entity Path	Path the entity on which the task was executed.
Entity Type	Type of the entity on which the task was executed.
Attribute	Identifier of the modified attribute (if applicable).
Old Value	Value of the attribute before the modification was performed (if applicable).
New Value	Value of the attribute after performing and/or saving the modification (if applicable).
(1) Headers are hidden by default. To show column headers, right-click a row inside the notification panel and click Show Table Headers .	

Using the Notification Panel

For each task that you launch, it is good practice to verify in the notification panel if the task is completed and its final status.

Select a task that has sub-entries and press **Enter** (or double-click it) to expand/collapse it. Sub-entries provide more detailed information about the task.

NOTE: When performing activities on a large number of entities, expanding entries in the notification panel may increase the response time of the engineering client temporarily. Such activities can be, for example, importing or generating 500 or more objects at once.

Filtering and Sorting Activity Data

Use the filter function, page 120 on columns to view, for example, only activity of one user by filtering on the **User Name** column.

By default, the most recent activity is displayed at the top of the list.





Exporting Activity Data

To save the history of actions, export activity data regularly and before stopping the system server.

To export to file (.csv) data that is displayed in the notification panel for a given period, proceed as follows.

Step	Action
1	Right-click any row inside the notification panel and click Export . Result: The Export dialog box opens.
2	In the Destination field, enter a file name and location, or use the default file name and location.
3	In the Period field, select a start date and time, and an end date and time to define the time span of activity data to export. NOTE: Only activity data that appears in the notification panel can be exported.
4	Click OK . Result: The export file is created and the folder containing it is opened. NOTE: Click Cancel to close the Export dialog box without exporting data. NOTE: Click Reset to reset the values of the Destination and Period fields to their default values.

In the export file, the icons of messages are replaced by the following keywords.

Icon	Keyword
	Success
	Not successful
	Warning
	Informational

NOTE: For aborted tasks, it is the most recent sub-task that is identified with *Aborted* in the **Action** column, not the task itself.

Audit Trail

This topic describes the Audit Trail functionality for system-engineering activities performed by users and the data that is sent to the syslog server.

Configuring Audit Trail

You can enable and configure Audit Trail (see *EcoStruxure Process Expert, Installation and Configuration Guide*) from the **Audit Trail** section of the **System Server Configuration Wizard**.

Working Principle

Once you have enabled and configured Audit Trail, when the system server is started, it connects to the syslog server and sends Audit Trail messages. Messages are sent in chronological order for actions that users perform on systems by using the engineering clients of the infrastructure.

Messages for certain user actions that are performed in the system server or operation clients are also sent.

The Audit Trail functionality of EcoStruxure Process Expert is independent of the **Event Logging** setting of the Control Participant (**Project Settings > PLC Diagnostics**).

NOTE: Syslog server software needs to be purchased, installed, and configured separately.

Scope of the Audit Trail

Typically, Audit Trail messages that are sent to the syslog server correspond to a user-action entry in the notification panel, page 85 of the engineering client.

For each child notification pertaining to a parent entry, one message is also sent. Messages for the main entry and its child notifications have the same **Domain** value.

No message is sent for system-generated entries and when a user action is canceled.

The following tables outline the type of actions for which messages are sent to the syslog server.

Actions Performed in Engineering Clients

Actions performed at the platform level	Comment
Logging in and out of the client.	—
Locking and unlocking the client.	—
Actions on systems.	For example, creating, deleting, or restoring systems.
Actions on system components.	Applications, topology, projects.
Import operations.	—
Deployment operations.	—
Actions on global templates.	—
Actions on content repository data.	—

Actions performed in a Participant	Comment
Control Participant: <ul style="list-style-type: none"> Refinement offline of project and sections Controller and PRM configuration 	Participant-generated messages are sent when changes are saved. Typically, this information does not appear in the notification panel of the engineering client.
Control Participant: Refinement online.	<p>Messages are sent when the Build and Deploy command is used.</p> <p>A message is sent also when you perform an action that impacts the controller directly (for example, initializing an animation table) even if you close the refinement window without building and deploying changes.</p>
Supervision Participant: <ul style="list-style-type: none"> Refinement Advanced settings Page editing 	<p>The information that is available is the same as that appearing in the notification panel.</p> <p>No details are provided about Participant data that is modified.</p>

Actions Performed in Operation Clients

Actions	Comment
Logging in and out of the client.	—
Locking and unlocking the client.	—
Connecting to a controller when using runtime navigation services.	A message is sent for obtaining the controller status when opening the Operation Client Viewer .

Actions Performed in the System Server

Actions	Comment
Starting and stopping the system server.	—
Logging in and out of the system server.	—

Audit Trail Message Format

The content of the message portion (message text or details) of a message that is sent to the syslog server uses the format defined by the syslog protocol (RFC 5424 specification), which is as follows:

<Time stamp> <Machine name> <Application name> – <User> : <Domain> :
<Action> : <Entity> : <Message> : <Attribute> : <Old value> → <New value>

Element	Description
<i>Time stamp</i>	Date and time when the action was performed. Format: YYYY-MM-DDThh:mm:ss:ms Milliseconds (ms) are indicated with 7 digits (0000000 if not available). The time stamp is based on the system time of the computer on which the EcoStruxure Process Expert system server is installed in UTC. Nevertheless, Schneider Electric recommends that the system time of computers on which EcoStruxure Process Expert clients are running be synchronized to the system time of the system server computer. NOTE: For actions performed inside the Control Participant, the time stamp corresponds to the Participant system time in UTC.
<i>Machine name</i>	Name of the computer on which the action was performed.
<i>Application name</i>	<i>EPE</i>
<i>User</i>	Name of the user who is logged in to the EcoStruxure Process Expert component on which the action was performed. Format: <i>Full name (Username)</i> . If the full name is not defined, the username appears instead.
<i>Domain</i>	Domain or project component on which the action was performed. For example, <i>Systems</i> or <i>Application</i>
<i>Action</i>	Description of the action that was performed.
<i>Entity</i>	Name of the object on which the action was performed.
<i>Attribute</i>	Identifier of the modified attribute (if applicable).
<i>Message</i>	Description of the action. Typically, the information that is available in the notification panel of the engineering client.
<i>Old value</i>	Value of the attribute before the action was performed.
<i>New value</i>	Value of the attribute after the action was performed.

NOTE: If an element contains no value, it is not shown.

Syslog Server Language Setting

When EcoStruxure Process Expert is installed in a language other than English, make sure to set the *Output Encoding* parameter of the syslog server to *Unicode (UTF8)*. Otherwise, the content of messages may not be readable.

NOTE: The name of the parameter to configure may vary depending on the syslog server software that you use.

Connection to the Syslog Server

The table describes the actions of the system server when it connects to the syslog server and when the connection is interrupted.

Connection of system server to syslog server	Description
Initial connection	<p>A message is shown in the notification panel and sent to the syslog server to indicate the status of the connection.</p> <p>In the notification panel, messages related to the connection status show <i>Global</i> in the System and Domain columns and <i>System</i> in the User Name column.</p>
Connection is interrupted	<p>The system server detects a disconnection when it is not able to send messages to the syslog server. In this case, it attempts to reconnect automatically up to three times. If unsuccessful, a message is shown in the notification panel.</p> <p>Thereafter, the system server attempts to reconnect at the configured interval.</p>
Connection is reestablished	<p>Two messages are shown in the notification panel and sent to the syslog server to indicate the following:</p> <ul style="list-style-type: none">• Date and time the disconnection was interrupted.• Date and time the connection was reestablished.

NOTE: Actions logged while the connection to the syslog server is interrupted are discarded. Notifications continue to show in the notification panel.

Activity Log Files

Overview

EcoStruxure Process Expert records the activity of each component (system server, engineering client, and operation client) by using log files.

In addition, a log file records information related to activated licenses for troubleshooting purposes.

The file size and number of files that are maintained for each component is limited. This is done to make the log files more user-friendly to view and to save space.

NOTE: For clients, the data that is logged is different from the activity data that is displayed in the notification panel, page 85.

Log File Location

Log files are located at the following paths:

- System server: *C:\ProgramData\Schneider Electric\Process Expert X\Logs*
- Clients: *C:\Users\Username\AppData\Roaming\Schneider Electric\Process Expert X\Logs*

Where:

- *X* corresponds to the version that is installed.
- *Username* corresponds to the username that was entered to log on to the current session of Windows, independently of the username that was entered to log in to the client, page 72.

Log File Name Syntax

The names of log files have the following syntax: *Component_Name.File_number.log*

Component_Name can be:

- *Server* for the system server.
- *Client* for engineering clients.
- *OperationClient* for operation clients.
- *License* for license-related information.

File numbers:

- The most recent log file has no file number.
- The second latest log file has number 0.
- The log files thereafter have incremental numbers from 1 to 9 where 9 represents the earliest log file containing the oldest information.

For example:

- *Server.log* is the current log file where EcoStruxure Process Expert records server activity.
- *Client.1.log* is the log file where EcoStruxure Process Expert recorded past engineering client activity. The file content is not updated anymore because *Client.log* is currently recording activity. *Client.1.log* was created when *Client.log* reached its size limit for the second time (the first time, *Client.0.log* was created).

Log File Size

The size of any log file cannot exceed 4 Mb. When the size of a log file has reached this limit, EcoStruxure Process Expert creates a new file. Existing log file numbers are incremented.

Number of Log Files

EcoStruxure Process Expert keeps up to 10 log files per component. When a new log file needs to be created because the current one has reached the size limit, and the number of files for a component is reached, EcoStruxure Process Expert deletes the earliest file for that component before creating the new one.

Using the Help

What's in This Chapter

Software Participant Help.....95

Using Help for Different Versions of Schneider Electric Templates.....96

Help on User-Created Templates.....98

Software Participant Help

Overview

You can open the Help of a Participant by clicking **Help** or the **?** button in the menu of the Participant when the Participant window is open (for example, after you click **Configure** in the context menu of a topological entity).

You can open the same Help also from the **Help EcoStruxure Process Expert** dialog box without having to open the Participant. To access the dialog box, click the **?** button in the toolbar of a client.

Help is available for these Participants:

- The Control Participant
- The Advantys Configuration Software (ACS)
- The Supervision Participant

NOTE: ACS is used to configure STB islands during the configuration stage. For simplicity, in this document, ACS is assimilated to the Control Participant.

NOTE: The dialog box lets you also open:

- The Help describing how to use the Modicon IEC 61850 Configuration Tool.
- The Help of OPC Factory Server.
- The Help of OPC UA Server Expert.

Opening the Help of a Participant

To open the Help of a Participant from a client window, proceed as follows.

Step	Action
1	In the toolbar, page 104 of the client window, click the Help button (?). Result: The Help EcoStruxure Process Expert dialog box opens.
2	In the Software Participant Help section, double-click a Participant name. Result: The Help of the Participant opens in a separate window.

Using Help for Different Versions of Schneider Electric Templates

Overview

For the templates that are included with EcoStruxure Process Expert, each software version contains only one version of each template (the latest) and installs the Help only for this version. This overwrites the Help of an existing previous version if you are upgrading EcoStruxure Process Expert. The Help is provided in compiled HTML (CHM) format.

However, your applications can contain several versions of a same template, for example, if you have migrated your database and have imported a later version of the template afterwards.

Starting from version 4.3 of Process Expert, you have the possibility to retain the Help that is included with any version to continue using it after you upgrade to a later version.

By verifying the version of a template when you use the context-sensitive Help button, EcoStruxure Process Expert opens the Help for this version if available.

This topic provides guidelines for using several versions of the Help and describes how to proceed.

Considerations When Using Several Versions of the Help

The Help of a given version of EcoStruxure Process Expert documents the templates and functionality provided by that version only.

If you open the Help for a given template and then, within the same Help window, you look up another template or a platform functionality, it may be that the version of the template or the functionality that is documented differs from the version/functionality that you are using.

This can be the case, for example, if:

- You have migrated your database and have not updated the templates used by instances of applications. In this case, the Help that is provided may document a later version of templates.
- You are opening the Help of an earlier version of EcoStruxure Process Expert. In this case, the templates that are available in the Library and the functionality provided by the platform may be of a later version than those documented.

It is a good practice to:

- Open the Help for a template only by using the context-sensitive Help button, which takes into consideration the version of the template that you are selecting.
- Close the help after each use of any given template.
- Verify the version of the Help you are viewing and which is indicated in the title of the Help window.

Location of the Help Files

On the computer running the engineering client, the Help files are located in the *User Guides* folder at the path *C:\Program Files\Schneider Electric\EcoStruxure\Process Expert*.

The *User Guides* folder also contains *ContextualHelpRepository.** files, which are required for the context-sensitive Help on templates to work.

Preserving the Help Files Before Upgrading

So that the Help of the installed version remains available after you upgrade to a later version, before proceeding with the upgrade, you need to copy the complete set of Help files and their auxiliary files to a specific location on the computer running the engineering client.

Proceed as follows:

Step	Action
1	<p>Create the following folder structure if it does not exist yet: <i>%localappdata%\Schneider Electric\Process Expert\User Guides</i>.</p> <p>NOTE: By default, the value of the <i>%localappdata%</i> environment variable is <i>C:\Users\username\AppData\Local</i>.</p> <p>A <i>Process Expert x</i> folder, where <i>x</i> corresponds to the installed major version (the <i>R</i>• version suffix does not appear), may already exist in the <i>Schneider Electric</i> folder but do not use it.</p>
2	<p>In the <i>User Guides</i> folder, create a subfolder with a name and a version number. For example, <i>Template Help Files Vx</i>.</p> <p>Mentioning the version helps you identify for which version you are opening the Help in case the same template version is provided and documented in different versions of EcoStruxure Process Expert.</p>
3	<p>Copy the files contained in the <i>User Guides</i> folder, page 96 to the subfolder that you have created in step 2.</p>

Opening the Help for Templates of an Earlier Version

To open the Help of a template for which several versions exist, click the context-sensitive Help button on the engineering client toolbar and select a template, an instance of a template, or a facet of an instance of a template.

The software verifies the version of the template and opens the Help file from:

- the installed version if the template is provided in the software.
- an earlier version in which the template was provided (given you have copied the necessary Help files to the [required location](#), page 97).

NOTE: Contextual Help is not available if the software version for the template is missing or incorrect. The **Contextual Help Conflict** dialog box is displayed if the template is in both versions. You must select a help version.

Help on User-Created Templates

Overview

You can configure EcoStruxure Process Expert to open the Help in CHM format of a user-created template.

You can also manage the CHM Help files of several versions of the same template so that clicking a given version of the template opens the corresponding Help.

Only one Help window opens at a time.

This topic describes the prerequisites and the steps required to configure contextual Help.

NOTE: By using this feature, you can also associate your own CHM Help to a Schneider Electric global template in addition to the existing Help. You can then select which Help file you want to open.

NOTE: You can also associate Help files to templates and their instances that operators can open from the operation client when using runtime navigation services (see *EcoStruxure Process Expert, Runtime Navigation Services, User Guide*).

Prerequisites

The following are the prerequisites to be able to open Help for user-created templates:

- You need to have Help files documenting user-created templates in the Microsoft® compiled HTML format (.chm).
For more information on creating a Help project, refer to the Windows [online Help](#).
- You need to know the topic ID, page 99 of the topic documenting the template, which you want to open when you use the context-sensitive Help button of the engineering client. If the Help file (.chm) documents several templates, you need to have the topic ID that corresponds to each template.
For more information on topics, refer to the Windows [online Help](#).

Creating a Repository for Help Files of User-Created Templates

So that the Help of user-created templates remains on the computer and can be opened after you upgrade to a later version, you need to copy your complete set of Help files and the *ContextualHelpRepository.xml* file to a specific location on the computer running the engineering client.

Proceed as follows.

Step	Action
1	Browse to %localappdata%\Schneider Electric\Process Expert\User Guides. NOTE: By default, the value of the %localappdata% environment variable is C:\Users\{username}\AppData\Local.
2	In the <i>User Guides</i> folder, create a copy of the folder named <i>Sample-To be copied</i> in the same location and rename it by using a meaningful name describing its contents and the version that was used to create the template. For example, <MyReactor Template Help Files for V2020>. NOTE: You can create as many copies of this folder as needed in case you want to keep the Help files of different templates in distinct folders.
3	Copy all your Help files to the subfolder that you have created in step 2.

NOTE: Uninstalling EcoStruxure Process Expert removes the folder named *Sample-ToBeCopied* but not the copies of it.

Identifying the ID of a Topic Inside a CHM Help File

The procedure describes how to identify the ID of a topic inside a CHM Help file.

Step	Action
1	Open the user-created CHM Help.
2	Select the page that you want to open when using the context-sensitive Help button on the template and right-click somewhere on the page. Result: The context menu of the page opens.
3	Select Properties . Result: The Properties dialog box opens.
4	Copy the topic ID. The topic ID of the page is the part of the string that starts after : / in the Address (URL) attribute. For example, if the Address (URL) attribute is <i>mk:@MSITStore:C:\PROGRA~2\XMLNOT~1\Help.chm::/Overview.htm</i> , the topic ID is <i>Overview.htm</i> .

Editing the XML File

To edit the *ContextualHelpRepository.xml* file, proceed as follows.

Step	Action
1	In the folder, page 98 to which you have copied your Help files, open the ContextualHelpRepository.xml file by using Notepad or an XML editor software. Result: The file that opens contains the following entry, which serves as an example only. <ContextualHelp Version="1.0"> <Item HelpKey="MyMotor3" FileName="PELbPrTp" MappingId="PELbPrTp_D-SE-0014236.htm" FileType="CHM" Description="" Version="2.7.2"/> </ContextualHelp>
2	Edit the following parameters of the entry: <ul style="list-style-type: none"> • HelpKey: Is the identifier of the user-created template. • FileName: Is the name of the CHM file without file extension. • MappingId: Is the ID, page 99 of the topic in the CHM file that documents the user-created template. Include the <i>.htm</i> file extension. • FileType: Do not edit this parameter. Default value: CHM. • Description: Optional. You can enter a short description by using free form text. It is used to describe the entry in the XML file and can appear in the Contextual Help Conflict dialog box, page 97. To use no description, leave it as "". • Version: Is the version of the user-created template as it appears in the Global Templates Explorer or in the Templates Browser pane of the Application Explorer.
3	Add a new entry for each user-created template for which you want Help to open. To manage the Help for different versions of the same template, refer to the next topic, page 100. NOTE: The </ContextualHelp> element must be the last entry in the file.
4	Save your changes.

Updating the Help for a New Version of a User-Created Template

The table describes the actions that you need to perform when you create a new version of a template and want Help to open for the various versions that exist. Two versions of a template are used as an example. *MyReactor 1.0.0* is considered the previous version and *MyReactor 1.0.1* the new version.

An updated Help file is available to document the new version	Action	Comment
No	<ol style="list-style-type: none"> 1. In the folder, page 98 to which you have copied your Help files, edit the <i>ContextualHelpRepository.xml</i> file. 2. Add a new entry that corresponds to the new version of the template by using the same parameter values as for the previous version; except for <i>Version</i>, which needs to correspond to the version of the new template. 	<p>The same Help file opens when you use the context-sensitive Help button on <i>MyReactor 1.0.0</i> or <i>MyReactor 1.0.1</i>.</p> <p>If you remove the entry for template version 1.0.0, Help cannot open when you select it.</p>
Yes	<ol style="list-style-type: none"> 1. Browse to the <i>User Guides</i> folder, page 98 located at the path %localappdata%\... 2. Create a copy of the folder in which the Help files of <i>MyReactor 1.0.0</i> and the <i>ContextualHelpRepository.xml</i> file are located. 3. Rename the copy of the folder by using a name that allows you to distinguish the previous and the new version. For example, <MyReactor 1.0.1 Template Help Files for V2018>. 4. In this folder, edit the entry in the <i>ContextualHelpRepository.xml</i> by modifying the value of <i>Version</i> to be equal to the new version (for example, <i>Version="1.0.1"</i>). 	<p>Clicking either template version when context-sensitive Help is enabled opens the Help file that is located in the folder, which also contains the XML file with an entry of the same template identifier and version.</p> <p>NOTE: It is assumed that the Help file name and the topic ID are identical for both template versions.</p>

Opening the Help for User-Created Templates

To open the Help of a user-created template, proceed as follows.

Step	Action
1	<p>Click the context-sensitive Help button in the toolbar, page 104 of the engineering client and select the user-created template.</p> <p>Result: EcoStruxure Process Expert verifies the version of the template and opens the corresponding Help file according to your configuration, page 99.</p>

NOTE:

- If the value of any parameter in the *ContextualHelpRepository.xml* file is incorrect (except for description), if the *ContextualHelpRepository.xml* file is missing or its location incorrect, or if the location of the folder containing the Help files and the XML file is incorrect, no Help is shown.
- If the template identifier and version appear in more than one *ContextualHelpRepository.xml* file contained in different subfolders of the *User Guides* folder, the **Contextual Help Conflict** dialog box opens, which lets you select from which location you want to open the Help.

Associating Other CHM Help Files to Schneider Electric Global Templates

By using an example, this topic describes how to associate your own Help file in CHM format to a sample control module template of a Global Templates Library for which the EcoStruxure Process Expert Help is already available.

The data used for the example is the following:

- Control module template: *\$Motor* version 1.0.0.
- Name of your Help file: *MyMotorHelp.chm*

- ID of the topic that you want to open: *mymotor_description.htm*

The table describes the steps that you need to perform to implement the Help and select, which file to open.

Step	Action
1	Create a repository for your Help, page 98 and copy the <i>MyMotorHelp.chm</i> file to this folder.
2	Edit the <i>ContextualHelpRepository.xml</i> file, page 99 located in this folder and add the following entry before the <code></ContextualHelp></code> element. <Item HelpKey="\$Motor" FileName="MyMotorHelp" MappingId="mymotor_description.htm" FileType="CHM" Description="" Version="1.0.0"/>
3	Click the context-sensitive Help button in the toolbar, page 104 of the engineering client and select the <i>\$Motor</i> template version 1.0.0, and instance, or facet of it. Result: The Contextual Help Conflict dialog box opens and shows the two Help files that are associated to the template: The EcoStruxure Process Expert Help and your Help.
4	Select the Help file that you want to view and click OK .

User Interface

What’s in This Chapter

- Engineering Client User Interface Description 103
- About** Dialog Box..... 106
- Engineering Client and Explorer Work Areas 107
- Organizing Work Areas 110
- Working with Panes 113
- Using Menus..... 117
- Software Participant Windows 118
- Filtering and Sorting Data..... 120
- Keyboard Shortcuts 123

Overview

This chapter describes the engineering client user interface, its main components, and the common tools that allow you to engineer and navigate a system.

Engineering Client User Interface Description

Overview

When you start the engineering client, it opens one main work area in which the various explorers and editors are displayed.

The engineering client user interface is composed of five main elements:

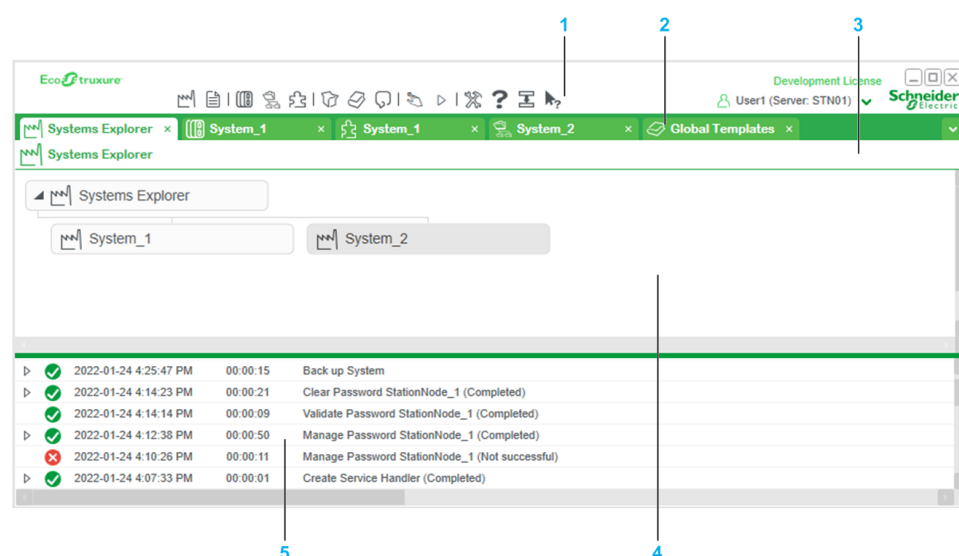
- The client toolbar
- The tabs bar of the main work area
- Explorer windows
- The notification panel, page 85

You can divide the main work area, page 110 of the engineering client.

The explorer that opens when a client is started depends on the group (see EcoStruxure Process Expert, Installation and Configuration Guide) that the logged-in user is a member of.

User Interface Description

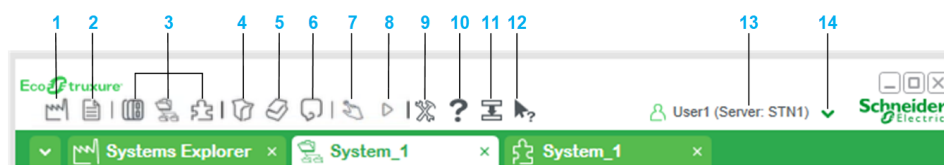
The following figure shows an example of the engineering client user interface.



Item	Description
1	Engineering client toolbar.
2	Tabs bar showing the tabs of the explorers and editors that are open.
3	Systems Explorer window.
4	Workspace of the Systems Explorer .
5	Notification panel.

Engineering Client Toolbar Description

The following figure describes the toolbar of the engineering client.




Item	Description
1	<p>Opens the Systems Explorer window, page 135, which lets you create, organize, and navigate systems.</p> <p>From the Systems Explorer you can open the explorers that are needed to create and manage system components:</p> <ul style="list-style-type: none"> • Application Explorer • Project Explorer • Topology Explorer
2	<p>Opens the Generate Report window, which lets you create and save to disk a report, page 1080 that contains engineering information about the selected system or one or more of its domains (such as topology or Control projects), including Participant project documentation.</p> <p>All windows in the engineering client, except the Systems Explorer, need to be closed to use the command.</p>
3	<p>Explorer navigation buttons to open or show the explorer of a system.</p> <p>From left to right:</p> <ul style="list-style-type: none"> • Topology Explorer • Application Explorer • Project Explorer <p>Clicking a button opens the corresponding explorer for either:</p> <ul style="list-style-type: none"> • The system that is selected in the Systems Explorer when this explorer is active. • The system for which an explorer or editor is active. For example, if the Assignment Editor for <i>System_1</i> is active, clicking the first button opens/shows the Topology Explorer for <i>System_1</i>. <p>If the explorer is already open, it is shown.</p> <p>NOTE: Navigation does not work from the Global Templates Explorer nor the Content Repository explorer by using these buttons.</p>
4	<p>Opens the Content Repository explorer, page 1120, which contains files related to Global Templates and the systems that you are engineering.</p>
5	<p>Opens the Global Templates Explorer, page 992, which you can use to browse and manage templates of the Global Templates Library.</p> <p>NOTE: In this example, the button is shown unavailable because the profile associated to the logged-in user, page 72 does not have the rights to use this explorer.</p>
6	<p>Opens the notification panel, page 85.</p>
7	<p>Lets you release the Supervision Participant so that you can open it (see <i>EcoStruxure Process Expert, Supervision Participant Services User Guide</i>) outside EcoStruxure Process Expert.</p> <p>Releasing is not required to launch the Plant SCADA runtime from the engineering client.</p>
8	<p>Launches the Supervision runtime (see <i>EcoStruxure Process Expert, Supervision Participant Services, User Guide</i>) with the Supervision project that is set as active on the local computer after you confirm the command.</p> <p>If the runtime is already launched, you can restart it by using the command.</p> <p>The runtime is not launched if the Supervision Participant is being used by EcoStruxure Process Expert (for example, if a Supervision project is being refined or its executable built).</p>
9	<p>Opens the About EcoStruxure Process Expert dialog box, page 106.</p>
10	<p>Opens a dialog box, which lets you access:</p>

Item	Description	
	<ul style="list-style-type: none"> The global EcoStruxure Process Expert CHM Help. It describes how to install, license, and use EcoStruxure Process Expert as well as the Global Templates that are included with it. The global Help of each software Participant, page 95. Other Help files. <p>Double-click an entry to open the corresponding Help.</p>	
11	<p>Opens the CHM Help at the topic describing the system engineering life cycle and the system creation workflow.</p> <p>It gives you an overview of the various stages of the life cycle and helps you identify what is the next step that you need to complete.</p>	
12	<p>Opens context-sensitive CHM Help.</p> <p>Clicking the Help button enables context-sensitive Help and a question mark appears next to the cursor.</p> <p>Selecting an item opens the Help when available. If no Help is available for the selected item, the functionality is disabled.</p> <p>To disable the functionality while the question mark is shown next to the cursor, press Esc.</p> <p>For an overview of user interface items for which context-sensitive Help is available, refer to the topic describing how to open context-sensitive Help.</p>	
13	<p>Full name of the logged-in user and name of the computer running the system server to which the client is connected. If the full name is not defined, the username appears instead.</p>	
14	<p>Opens the User menu, which contains the following commands.</p>	
	Lock	Locks, page 74 the components to which you are logged in on the local computer.
	User Information	Opens the User Information dialog box (see EcoStruxure Process Expert, Installation and Configuration Guide).
	Log In/Log Out	Lets you log in or out, page 72 of EcoStruxure Process Expert.

About Dialog Box

Opening the About EcoStruxure Process Expert Dialog Box

To open the dialog box, click either:

- The  icon in the toolbar of the engineering or operation client, page 104.
- **Help > About** in the menu bar of the system server.

General Tab

The **General** tab displays information related to:

- Installed EcoStruxure Process Expert and third-party software versions.
- The activated application size license and the total instance count for your database.
- The installation and expiration dates of entity certificates (see EcoStruxure Process Expert, Installation and Configuration Guide) installed on the local computer and of the corresponding root CA.

If no entity certificate is installed, no information is shown.

System Details Tab

The **System Details** tab displays the instance count by system, page 180 for the mounted database.

Client Details Tab

The **Client Details** tab displays information about the clients that are currently connected to the system server with a user being logged in. It shows the computer name, user name, and the type of client.

Support Tab

Contains the contact information of the Schneider Electric support.

The **Logs Folder** link lets you access the activity log files, page 92.

Hotfix Details Tab

The **Hotfix Details** tab displays information about installed hotfixes.

If no hotfix is installed, the tab is hidden.

Engineering Client and Explorer Work Areas

Overview

By default, in an engineering client only one work area is visible at a time.

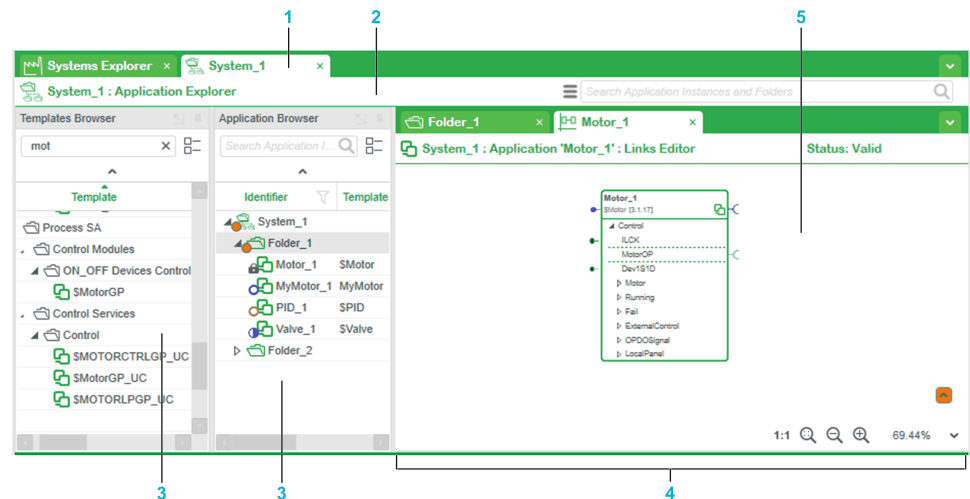
The work area is the display area that is below the tabs bar, page 103 where explorers and certain editors open, such as the **Systems Explorer** and **Composite Editor**.

Certain explorers (for example, the **Application Explorer** or **Project Explorer**) also feature a work area in which various windows and editors open, such as **Properties** windows or the **Instance Editor**.

You can create additional work areas, page 110 and organize them.

NOTE: When no window opens in the work area, it is called a *workspace*. This applies to editors and certain explorers (for example, the **Global Templates Explorer**).

The following figure shows an example of the work area of the **Application Explorer**.



Item	Description
1	Tab of the Application Explorer .
2	Toolbar of the explorer.
3	Panes of the explorer, which are open in the work area.
4	Remaining work area of the explorer in which a window and an editor are open. You can select them by using their tab.
5	Workspace of the editor.

Tabs Bar Description

For each window, explorer, or editor that you open in a work area, a tab is displayed in the tabs bar, which features:

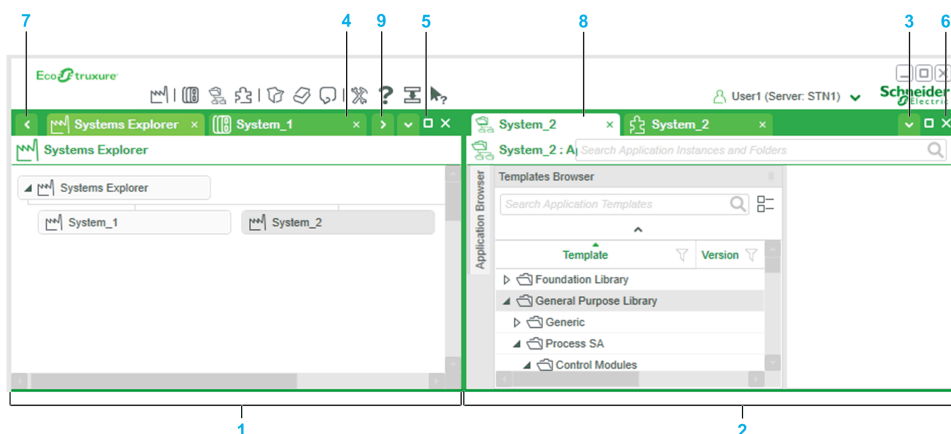
- An icon representing the type of window, explorer, or editor (for example, **Properties** window, **Application Explorer**, or **Links Editor**).
- The identifier of the system that is displayed in the corresponding window (except for the **Content Repository** explorer, **Global Templates Explorer**, and **Systems Explorer**).
- A button to close it.

You can rearrange the order of tabs on the tabs bar by dragging them to a new position.

NOTE:

- Closing the last tab in a tabs bar closes the work area (except if the work area is the last one to be open).
- You cannot close the last tab of the last work area that is open.
- To close all windows except one, right-click the tab of the window that you want to leave open and select **Close Other Tabs**.

The following example describes the tabs bars of the engineering client in which two work areas are open.



Item	Description
1	Main work area.
2	Additional work area.
3	Display all Tabs button. Displays a list of the tabs that are open in this work area.
4	Inactive tab.
5	Maximize/Restore Down button. The button is available when more than one work area, page 110 is open. It maximizes the work area in the window or reverts the display to a multiple work area view. NOTE: When you maximize a work area, the other open work areas are hidden. Use the Restore Down button to show them again.
6	Button to close the work area. The button is available when more than one work area, page 110 is open. It allows you to close the other work areas. You cannot close the main work area. NOTE: Closing the last tab in a work area closes it.
7	Scroll left button. Click the button to view hidden tabs that are to the left of the first tab. The scroll increment is one tab.
8	Active tab.
9	Scroll right button. Click the button to view hidden tabs that are to the right of the last tab. The scroll increment is one tab.

Moving the Workspace (Panning)

In editors (for example, the **Links Editor**), click an empty area of the workspace and hold down the left mouse button. The pointer turns into a hand icon. You can now drag the workspace in any direction by moving the pointer. This is useful to view items that are outside the visible workspace.

In certain explorers (for example, the **Global Templates Explorer**), you can use the following methods to move the workspace.

Action	Direction in which the workspace is moved
Mouse wheel up	Down
Mouse wheel down	Up
Shift + Mouse wheel down	Left
Shift + Mouse wheel up	Right

Magnifying and Reducing the Workspace View

To magnify or reduce the workspace view gradually, use the following method.

Ctrl + Mouse wheel up	Zoom in. The position of the pointer becomes the center of the magnification.
Ctrl + Mouse wheel down	Zoom out. The position of the pointer becomes the center of the reduction.

NOTE:

- A zoom selection menu may be available at the bottom of some workspaces. You may also be able to use the radar view.
- In some workspaces, starting out from the default zoom level, you can only zoom in.
- Zooming out beyond a certain level may switch the display to **simplified view**, page 1032.

Selecting Multiple Items

The method to select multiple items varies depending on the editor or explorer you are working in:

- In editors (for example, template editors), you can select multiple items in the workspace by using either or a combination of the following methods:
 - Select an item with the pointer, press and hold **Ctrl**, and select other items.
 - Press and hold **Ctrl**, click the workspace, and hold down the mouse button. Then, drag the pointer to draw a rectangle. Items that overlap with the rectangle are selected when you release the mouse button. Last, release **Ctrl**.
- In explorers, browsers, and various containers, you can select multiple items by selecting an item with the pointer, pressing and holding **Ctrl**, and selecting other items. To select contiguous items, select the first one, press and hold **Shift** and select the last item. Release **Shift**.

To unselect an item, press **Ctrl** and click the item.

NOTE: While you are still holding **Ctrl**, click a selected item again to unselect it.

Organizing Work Areas

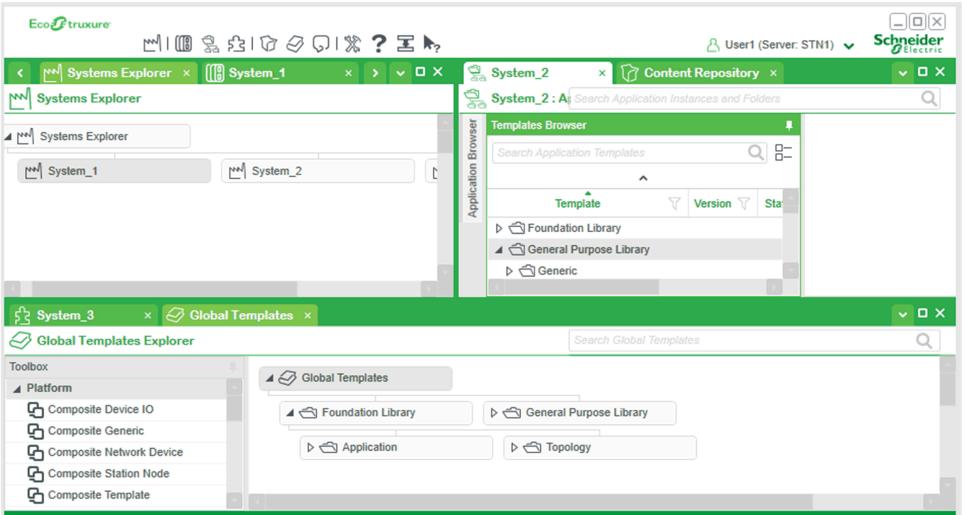
Overview

To work more effectively, you can customize the client user interface by creating additional work areas, page 110 left, right, or below the main one. Inside each work area you can open various explorers and editors.

Each work area has its own tabs bar, page 107 and toolbar (if available).

NOTE: You can also divide the work area of explorers, page 111.

The following figure shows an example of an engineering client user interface that is divided into three work areas.



Viewing Client Windows on Separate Monitors

To open a new window for the same client, so that you can view it on a separate monitor, proceed as follows.

Step	Action
1	Verify that, at least, two windows are open in the same or separate work areas. For example, the Systems Explorer and the Project Explorer windows.
2	Right-click the tab of a window in the tabs bar and select Open in new Window . Result: The tab opens in a second, separate client window.
3	Drag any of the two client windows to the other monitor.

NOTE:

- You can repeat the procedure from any tab of either client window given that two tabs are open in the work area. (The functionality is not available for tabs that are open inside the **Application Explorer** or **Project Explorer**.)
- Notifications about actions that you perform in the additional client windows appear in the notification panel of the main client window.

Creating Additional Work Areas

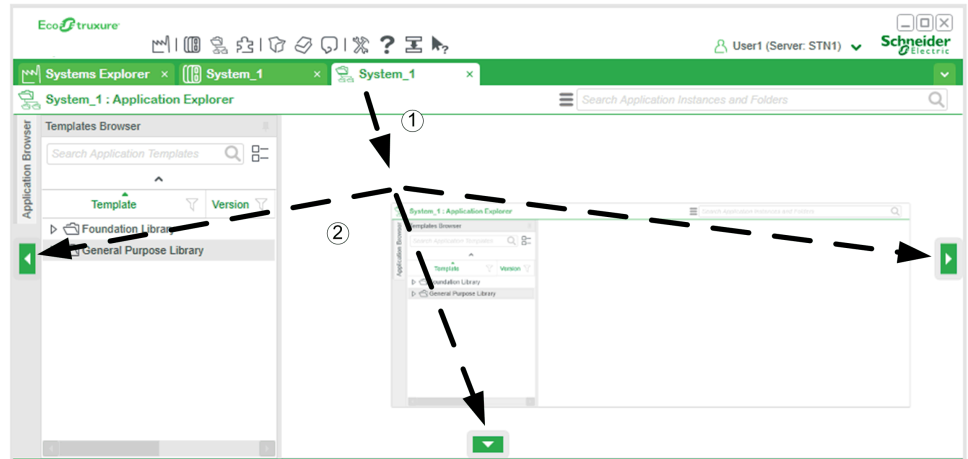
You can create a new work area with an already open explorer by using the docking tabs. The new work area opens on the left, right, or at the bottom of the existing one depending on which docking tab you use.

Once you have created several work areas, you can:

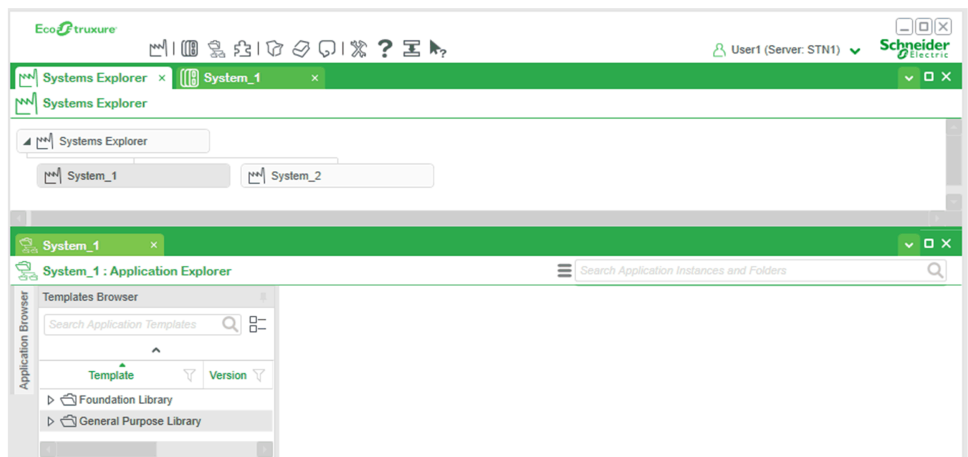
- Drag tabs from one work area to the other.
- Reposition work areas by using the docking tabs.

Refer to the topic describing the tabs bar, page 107 for information on how to manage windows inside a work area.

The following figure illustrates the use of the external docking tabs to create a new work area.



Example of result when you use the bottom docking menu.



To open a new work area, proceed as follows.

Step	Action
1	Verify that at least two explorers are open in the work area.
2	Drag a tab from the tabs bar towards the side of the work area where you want to create the new one. Result: The explorer follows the pointer and the external docking tabs appear.
3	Keep dragging the pointer over the docking tab where you want the new work area to open and release the mouse button. Result: A new work area, which contains the explorer opens at the selected location. NOTE: To cancel the operation, drag the pointer away from the docking tab and release the mouse button.

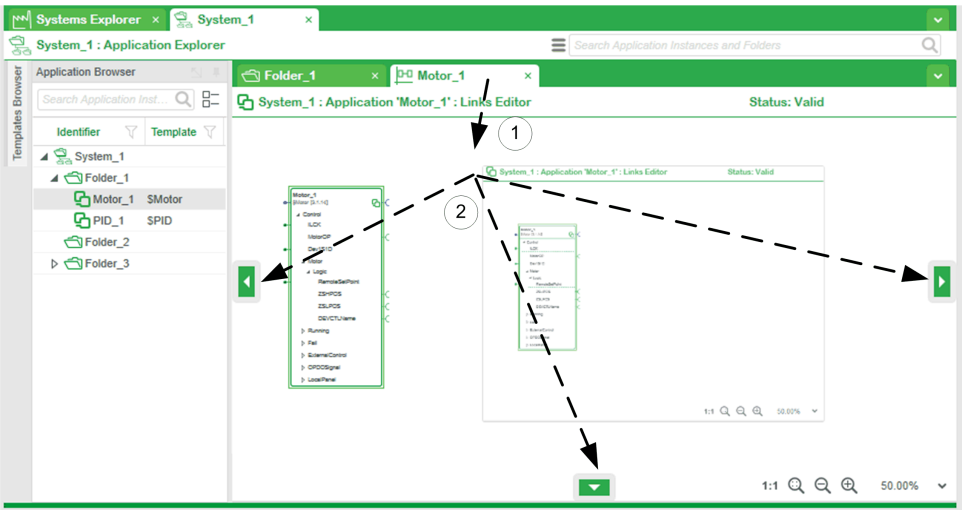
Creating Additional Work Areas in Explorers

When several windows are open in the work area of the **Application Explorer** or **Project Explorer**, you can create additional work areas by dragging either of the windows to a docking tab.

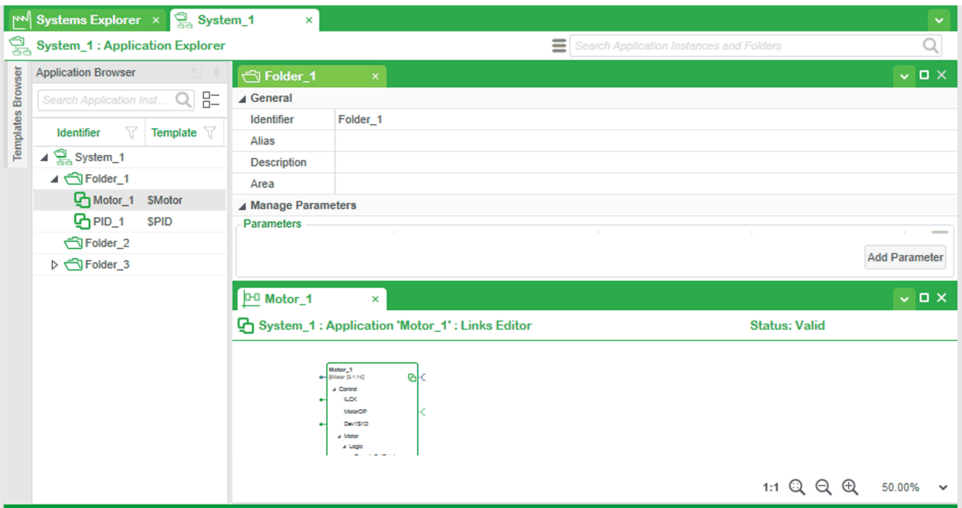
Refer to the topic describing the tabs bar, page 107 for information on how to manage windows inside work areas.

NOTE: You cannot create a new work area with the pane of an explorer. Refer to the topic describing how to work with panes, page 113.

The following figure illustrates the use of the external docking tabs to create a new work area inside the **Application Explorer**.



The following figure shows an example of the **Application Explorer** with two work areas that are open.



To open a new work area, proceed as follows.

Step	Action
1	Verify that at least two windows or editors are open in the work area of the explorer.
2	Drag a tab towards the side of the work area where you want to create the new one. Result: The item follows the pointer and the external docking tabs appear.
3	Keep dragging the pointer over the docking tab where you want the work area to open and release the mouse button. Result: A new work area, which contains the window opens at the selected location. NOTE: To cancel the operation, drag the pointer away from the docking tab and release the mouse button.

Working with Panes

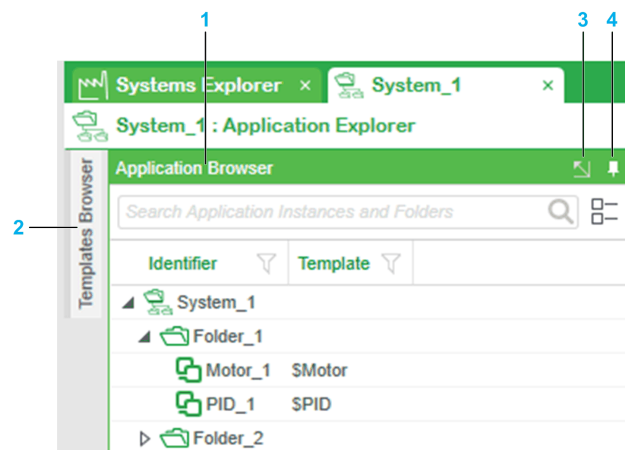
Overview

The software displays panes in various explorers and editors.

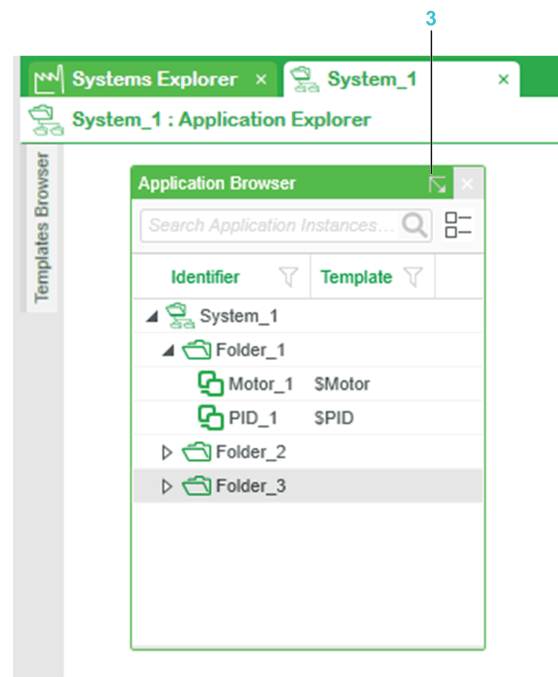
You can minimize them automatically when you do not need to see them, resize, and move them as required.

Pane Description

The following figure shows a partial view of two panes in the **Application Explorer**.



The following figure shows a floating pane. The pin button is not available.



Item	Description
1	Pane that is pinned and always visible.
2	Pane that is minimized to a tab.
3	Extract/retract button. Turns it into a floating pane that you can dock or returns a pane to its previous position (docked or tabbed).
4	Pin button. Lets you show and hide the pane.

Minimizing, Showing, and Restoring Panes

To minimize, show, and restore a docked or tabbed pane, proceed as follows.

Step	Action
1	Click the pin button in the title bar of the pane. Result: The pin button is displayed in a horizontal position and the pane will be minimized to the edge of the screen as a tab.
2	Move the pointer over the tab. Result: The pane is shown entirely while the pointer is inside the pane. NOTE: Click inside the pane so that it remains visible even you move the pointer away from it. Click outside the pane to minimize it again as a tab.
3	Click the pin button again. Result: The pin button is displayed in a vertical position and the pane is restored. If there are several overlapping panes, it may be shown only partially.

Extracting Panes

To make a pane floating, click the extract button or drag it by the title bar.

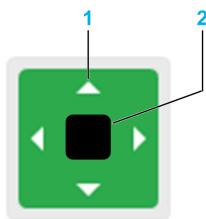
When panes are docked on top of each other so that their tabs appear at the bottom, dragging either pane by its title bar extracts the complete group. To extract only one pane, drag it by its tab or use its extract button.

Docking Panes

You can move docked panes to another area of the workspace or dock floating panes by using the [external docking tabs](#), [page 110](#).

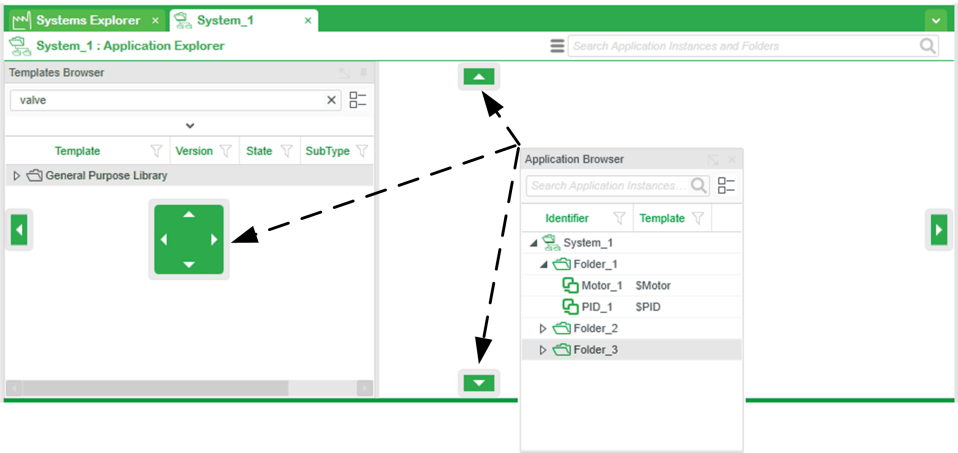
When the workspace contains more than one pane and you drag one pane on top of the other, a docking menu appears in the center of the other pane. It lets you position panes in a relative position to each other.

The following figure shows the docking menu.



Item	Description
1	The up, right, down, and left arrows let you position the pane on top, to the right, below, or to the left of the other one respectively.
2	The center pad positions the pane on top of the other one as a tabbed pane. The center pad appears when you move the pointer over the square docking menu but outside one of the arrows.

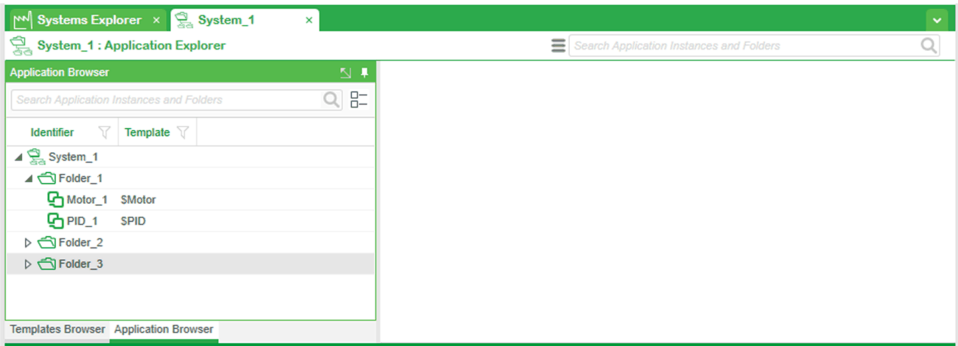
The following figure illustrates the use of the docking menu or external docking tabs to reposition a pane (docked) inside the work area of an explorer.



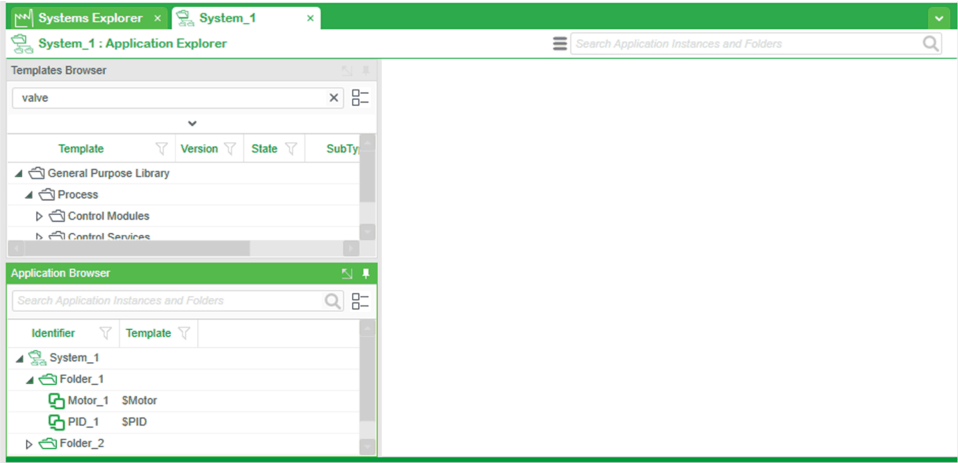
To dock a pane, proceed as follows.

Step	Action
1	Drag the pane by its title bar. If the pane is already docked, extract it, page 114 first. Result: The docking menu and/or the external docking tabs, page 110 appear.
2	Keep dragging the pointer over the docking menu or tab where you want the pane to be positioned and release the mouse button. Result: The pane is docked.

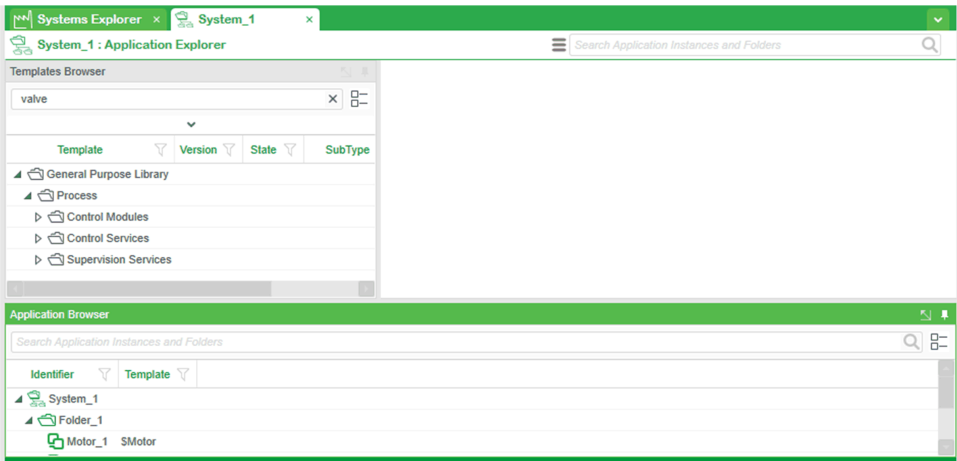
The following figure shows an example of panes that have been tabbed by using the center pad of the docking menu.



The following figure shows an example of a pane that has been docked below another one by using the down arrow of the center pad of the docking menu.



The following figure shows an example of a pane that has been docked to the bottom edge of the engineering client window by using the bottom external docking tab.



Resizing Panes

Typically, panes do not feature a maximize button. Instead, resize them to occupy the entire work area.

To resize a pane, proceed as follows.

Step	Action
1	Move the pointer over an edge of the pane. Result: The pointer turns into a double-ended arrow.
2	Drag the pointer away from the edge. Result: The pane is resized.

Closing Panes

To close a pane, click the close button in the title bar of the pane (when available). To open it again, select the corresponding command from the menu of the explorer or editor.

If no close button is available or if it does not allow closing the pane, minimize the pane, page 114 to a tab.

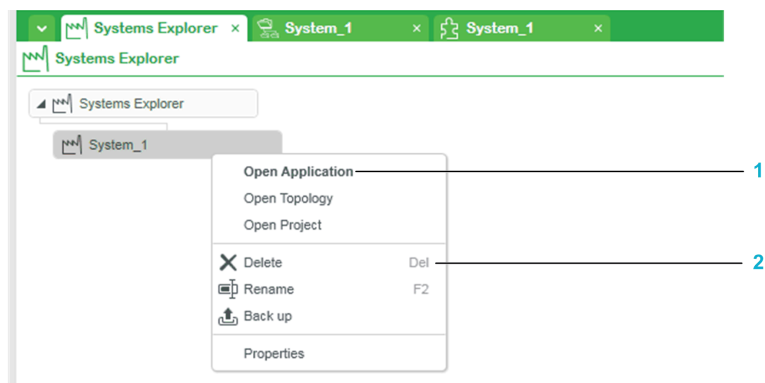
Using Menus

Context Menus

For many items of the user interface (for example, folders, instances, elements of projects), right-clicking the item opens a context menu containing several commands.

The commands are described in the topic documenting the item.

The figure shows an example of the context menu that opens when you right-click a system in the **Systems Explorer**, which lets you perform actions on the system.



Item	Description
1	You can execute commands that appear in bold also by pressing Enter when the item is selected or double-clicking the item itself (in this example, the item is System_1).
2	For certain commands, you can also use the keyboard shortcut, page 123 that is indicated to the right of the command when the item is selected.

NOTE: To close a context menu, press **Esc**.

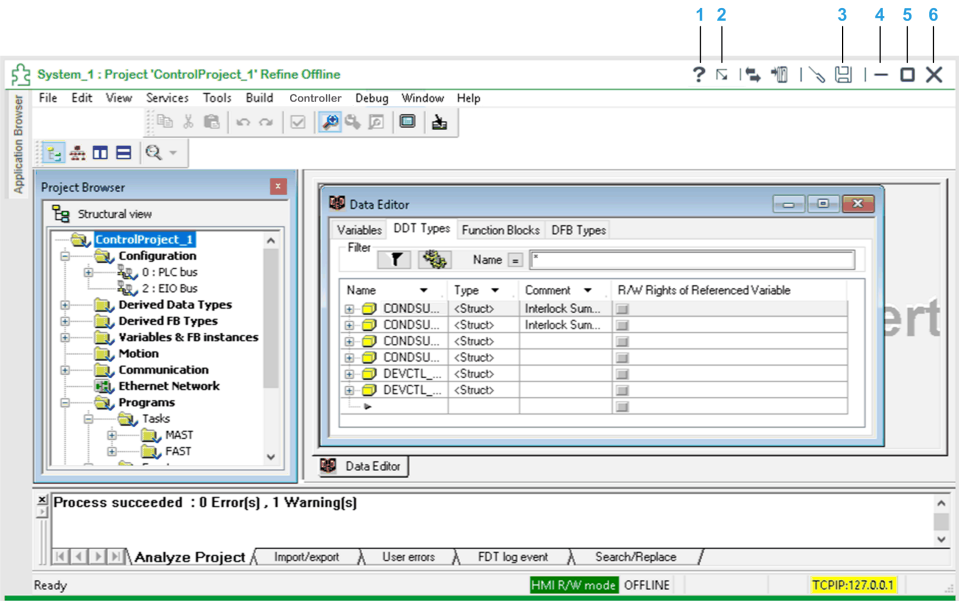
Software Participant Windows

Floating Software Participant Windows

When you open a Participant window, it opens as a floating window and maximized.

You can resize this window, display it on another monitor, or embed it in the client window (except in the **Topology Explorer** where Participant windows cannot be embedded).

You can open several floating windows simultaneously.



The following figure shows an example of a floating Control Participant window. The window is shown restored down.

Item	Description
1	Open the EcoStruxure Process Expert Help.
2	Embed the floating Participant window in the client.
3	Save changes that were made in the Participant.
4	Minimize the floating window.
5	Maximize the floating window (or restore down the floating window when it is maximized).
6	Close the floating Participant window.

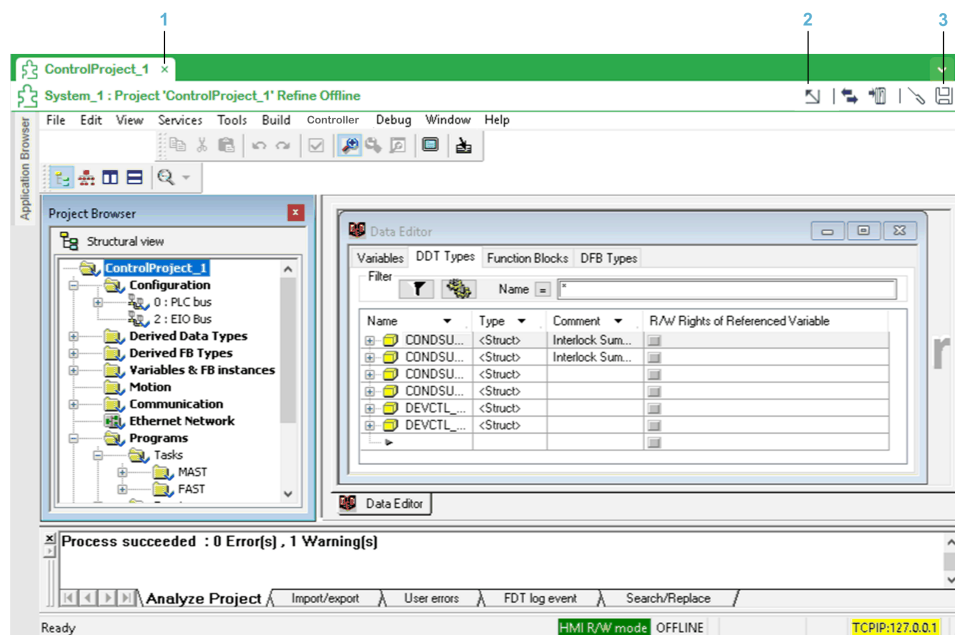
NOTE: The other buttons of the Participant window toolbar are task-specific and are described in the topic documenting the task.

While the floating Participant window is open:

- The embedded Participant window is not refreshed and displays the view that was shown when you extracted the floating Participant window.
- You cannot work in the embedded Participant window nor close it.
- Use the icon in the taskbar to display the client user interface when the floating Participant window is maximized.
- The Windows snap feature is enabled.

Embedded Software Participant Windows

The following figure shows an example of a Control Participant window, which is embedded in the engineering client by using the restore button. The functionality is not available in the **Topology Explorer**.



Item	Description
1	Close the Participant window
2	Extract and maximize the Participant window
3	Save changes that were made in the Participant

Showing Software Participant Windows on Separate Monitors

To show a software Participant window on another monitor, extract it, page 119 and drag it to the other monitor.

If you experience issues viewing the extracted Participant window on an extended monitor (for example, if the window turns black), try disabling hardware acceleration for Windows Presentation Foundation (WPF). The required setting may already be present in the registry and you can modify it; otherwise, you need to add it.

To disable hardware acceleration for WPF, proceed as follows. Follow this procedure only if you are experiencing issues viewing an extracted Participant window on an extended monitor.

Step	Action
1	By using the Windows Registry Editor of the computer on which the client is installed, go to HKEY_CURRENT_USER > SOFTWARE > Microsoft > Avalon.Graphics
2	Add a new registry key by right-clicking the Name column and selecting New > DWORD (32-bit) Value . NOTE: If the DisableHWAacceleration registry key is already present, proceed to step 4 if necessary.
3	Enter DisableHWAacceleration as name.
4	Right-click DisableHWAacceleration , select Modify , and enter 1 as data value.
5	Restart the computer.

Filtering and Sorting Data

Overview

For most columns inside folders and panes, you can open a filter menu that lets you refine data that is displayed according to criteria and/or conditions that you define.

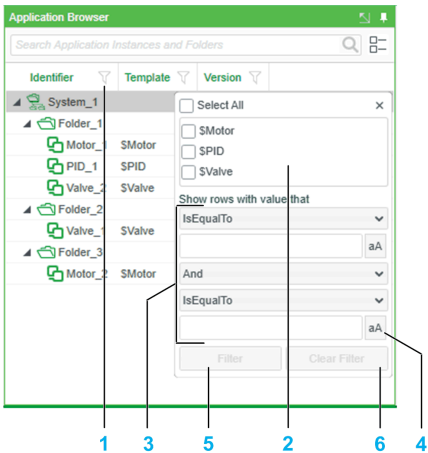
In addition, you can sort data in ascending or descending alphanumeric order.

Filters and sorting are reset when you close the pane, window, or editor in which they appear.

NOTE: When a filter is active, commands that you execute at a parent node level may also execute on data that is not displayed. This can be the case, for example, when you execute a command at the folder level in the **Application Explorer** while a filter is applied on the **Identifier** column of the folder and, as a result, not all the instances contained in the folder are displayed.

Filter Menu Description

The following figure shows an example of the filter menu.

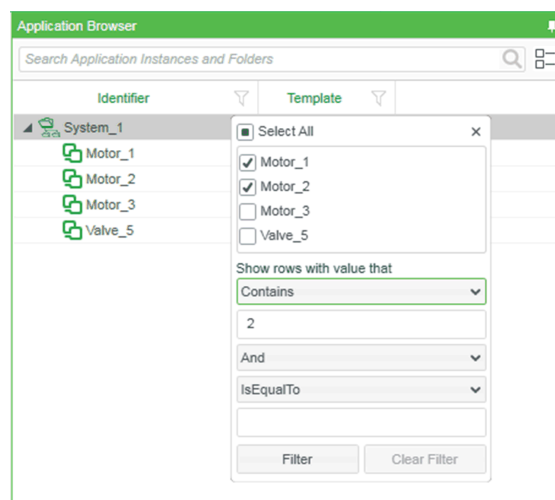


Item	Description
1	<p>Column filter icon.</p> <p>Click the icon to open the filter menu.</p> <p>The icon turns dark to indicate that you have applied a filter on the column.</p>
2	<p>Data field.</p> <p>The field displays in a flat list one of each of all the entries that the column contains. That is, if an entry appears more than once in the column, it is displayed only once in the data field. If the column contains only partial data because, for example, a keyword search has been performed, the data field displays also only the partial data.</p> <p>Select one or more entries. Only entries that you select will be displayed when you apply the filter.</p> <p>NOTE: If the column contains a tree view (for example, the Templates Browser pane of the Application Explorer), the content of the data field depends on the type of lowest-level child elements shown in the column:</p> <ul style="list-style-type: none">• A single type of child elements (for example, only templates): The data field displays only child elements that are not part of the tree structure (for example, only templates but no folders).• Several types of child elements (for example, the Name column of the Import window in the Projects Explorer for a Control project, which contains services and sections as lowest-level child elements): The data field displays only the parent nodes (for example, the system root folder).
3	<p>Expression builder.</p>

Item	Description
	<p>Lets you build custom filter expressions by selecting one or two conditions and an operator from a list.</p> <p>The conditions apply to any data contained in the column even if it is not shown in the data field. The list of available conditions varies and is adapted to the data type of the column.</p> <p>NOTE: If you build an expression and select an entry in the data field, your selection is used as an additional condition in the expression with the AND operator. If you select several entries in the data field, each one is treated as OR condition before the result is used as an additional condition.</p>
4	<p>Match case button.</p> <p>Click the button to enable case-sensitive filtering as an additional condition.</p> <p>The button appears only for columns showing data of String type.</p>
5	<p>Click the Filter button to apply the filter to the column.</p> <p>NOTE: When the filtered data is modified (for example, new objects have been created or imported, objects have been renamed), the filters do not apply to the modified data. To apply them, open the filter menu again and click Filter.</p>
6	<p>Click the Clear Filter button to clear the filter for the column.</p> <p>NOTE: To clear filters for all columns, right-click any column header and select Clear Filters.</p>

Example

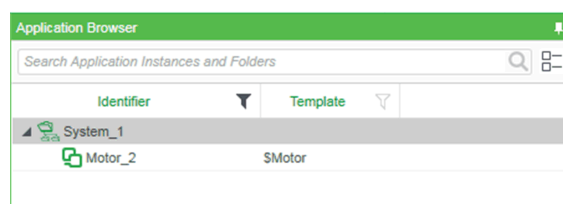
The following figure shows a filter example consisting of two items selected in the data field and one expression. It is used to filter the contents of the **Identifier** column.



The resulting filter expression is:

(Motor_1 OR Motor_2) AND Identifier Contains 2.

The following figure shows the search result once the filter is applied; instances for which the expression is true.



Using Filters

To use filters, proceed as follows.

Step	Action
1	Click the filter icon of the row that contains the data that you want to filter. Result: The filter menu opens.
2	Select entries in the data field or create a filter using the expression builder.
3	Click Filter . Result: The column displays data that matches the criteria and the filter icon turns dark. NOTE: When columns contain tree views, the parent structure of filtered items is shown.
4	Close the filter menu.

NOTE: You can use filters for several columns simultaneously. Repeat the above procedure on each column.

Sorting Data

The sorting feature is represented by a green arrow head inside the column header.

Click a column header to enable sorting of the data that the column contains.

Clicking the column header repeatedly toggles between the following states:

- Arrow up: Sorts in ascending alphanumeric order.
- Arrow down: Sorts in descending alphanumeric order.
- No arrow: No sorting. Data is displayed by using the default display order defined for the column.

NOTE:

- Hold **Shift** to enable sorting of multiple columns.
- To clear sorting on all columns, right-click any column header and select **Clear Sorting**.
- If a rule has been defined to display contents of a column, enabling the sorting feature does not override this rule. For example, in the **Application Browser** of the **Application Explorer** folders are shown first, followed by instances. Applying sorting sorts only items within their respective category. It does not change the display order of categories.

Clearing All Filters and Sorting

To clear filters and sorting for all columns, right-click any column header and select **Clear All**.

Keyboard Shortcuts

Available Keyboard Shortcuts

You can use the following keyboard shortcuts in place of the command in the context menu or the button.

Keyboard shortcut	Corresponding command or button
Alt+A	Navigation button, page 104. Opens or shows the Application Explorer for the system. NOTE: The keyboard shortcut is available from the Topology Explorer only if the tab of this window is selected.
Alt+P	Navigation button, page 104. Opens or shows the Project Explorer for the system. NOTE: The keyboard shortcut is available from the Topology Explorer only if the tab of this window is selected.
Alt+T	Navigation button, page 104. Opens or shows the Topology Explorer for the system.
Ctrl+A	No corresponding command. Use this keyboard shortcut to select the entire contents of a folder or container, such as instances or facets.
Ctrl+C	Copy
Ctrl+F	Find (Global Templates) Places the cursor in the search field of the Global Templates Explorer .
Ctrl+L	Edit Links. Opens the Links Editor of the Application Explorer .
Ctrl+V	Paste
Del	Delete
Enter	View (Global Templates)
F2	Rename
Shift+F12	Edit (Global Templates)

Engineering a System

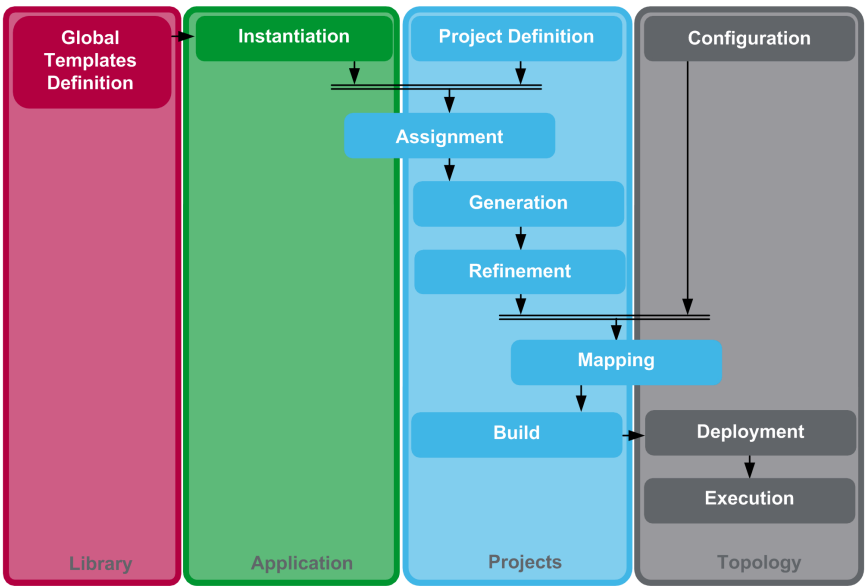
What's in This Part

System Creation Workflow and Tasks	125
Managing Systems	134
Instantiation Stage	157
Project Definition Stage	262
Assignment Stage	340
Generation Stage	396
Refinement Stage	445
Configuration Stage	512
Properties of Topological Entities	673
Mapping Stage	719
Build Stage	773
Deployment Stage	793
Execution Stage	854
Engineering M580 Safety Systems	908
Managing Communication	935

Overview

This part describes how to complete the different stages of the system engineering life cycle to engineer a system.

The following figure shows the stages of the system engineering life cycle.



System Creation Workflow and Tasks

What's in This Chapter

System Creation Workflow	126
System Creation Tasks	130

System Creation Workflow

Navigation

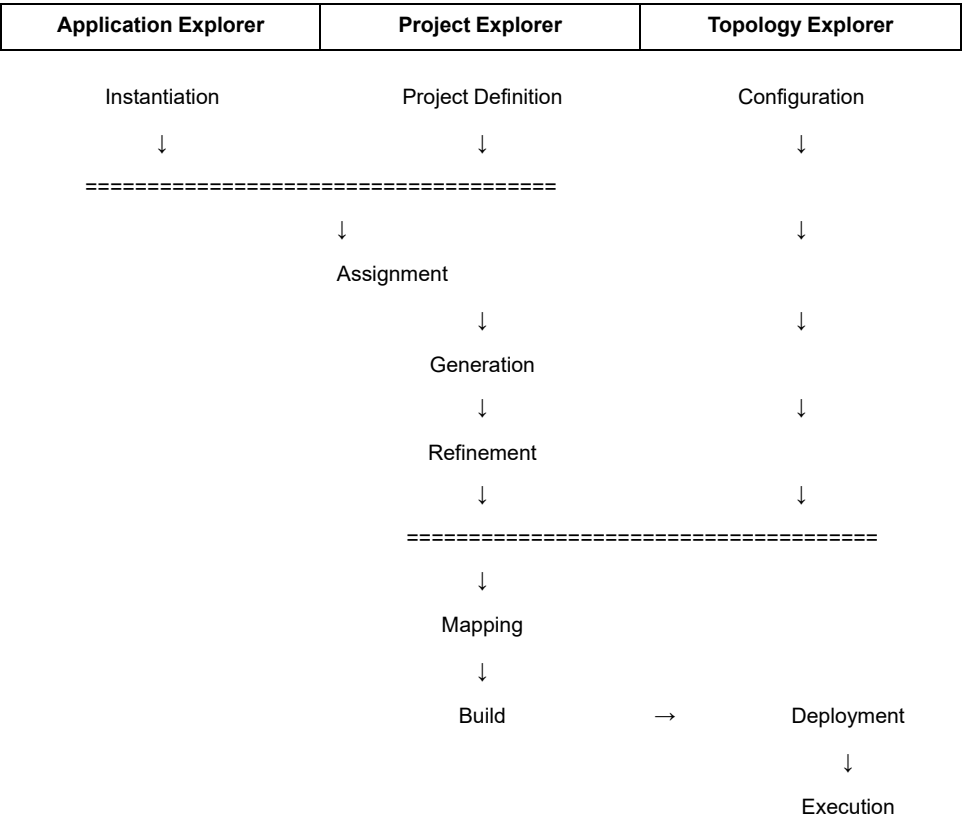
This page shows the stages of the system engineering life cycle and which Explorer is used to complete them.

In the CHM Help:

- Click an Explorer to view the main tasks of the workflow.
- Click a stage for detailed information on how to complete it.

NOTE: In the CHM Help, use the **Back** button in the toolbar of the Help window to revert to this page.

System Engineering Life Cycle

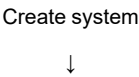


NOTE: Specific transversal workflows are also provided describing how to implement communication, page 935.

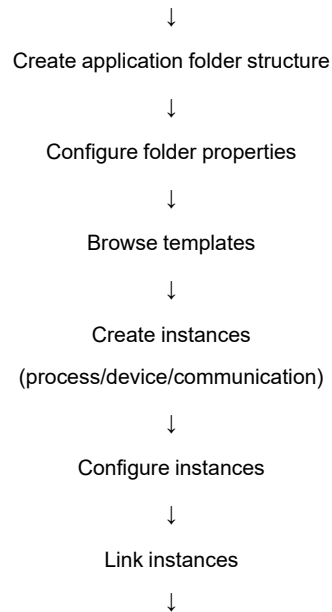
System Creation Workflow

The following workflow shows the main tasks that you need to perform to complete the system engineering life cycle. Click a task to navigate to the topic describing how to complete it.

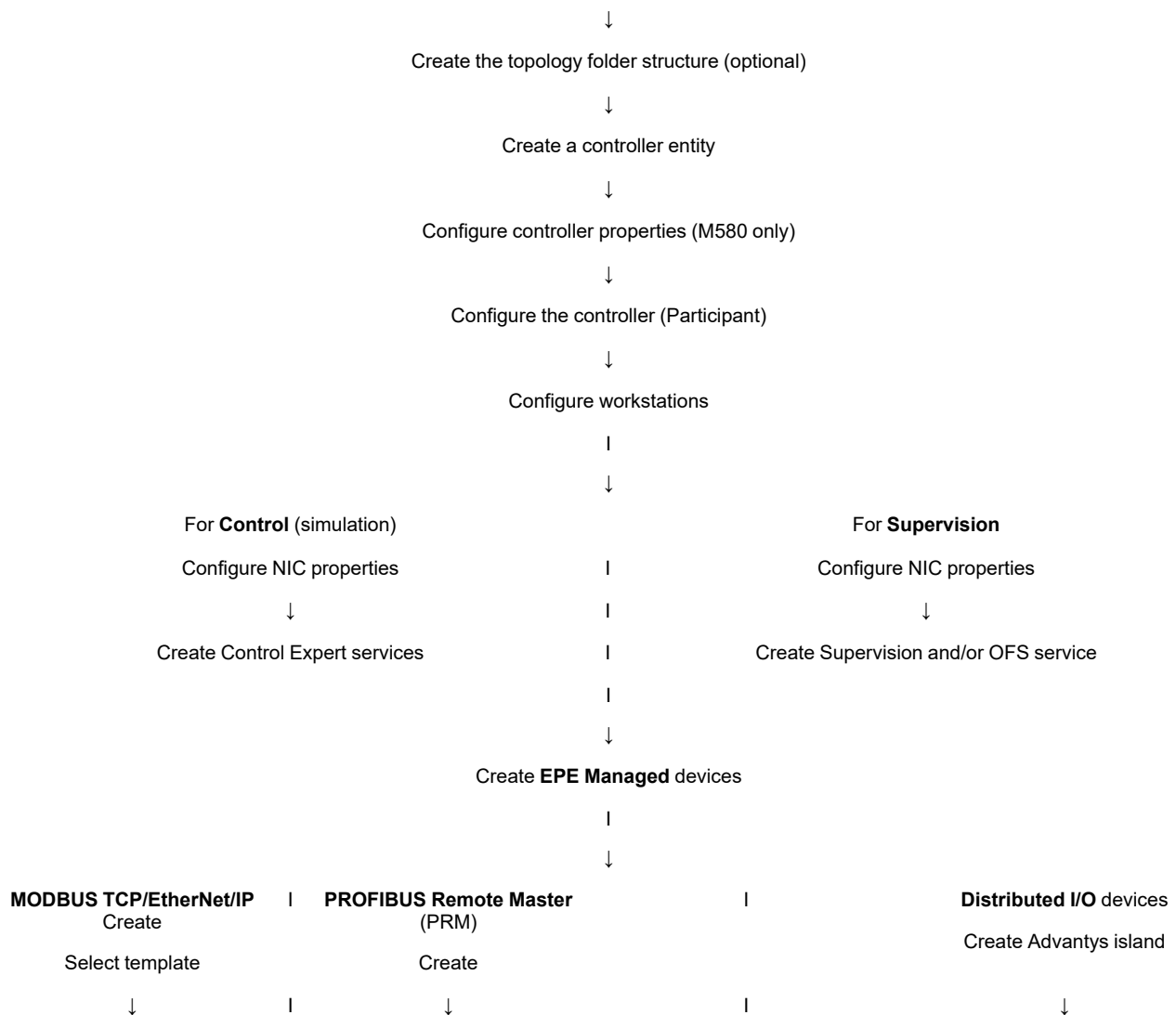
Systems Explorer

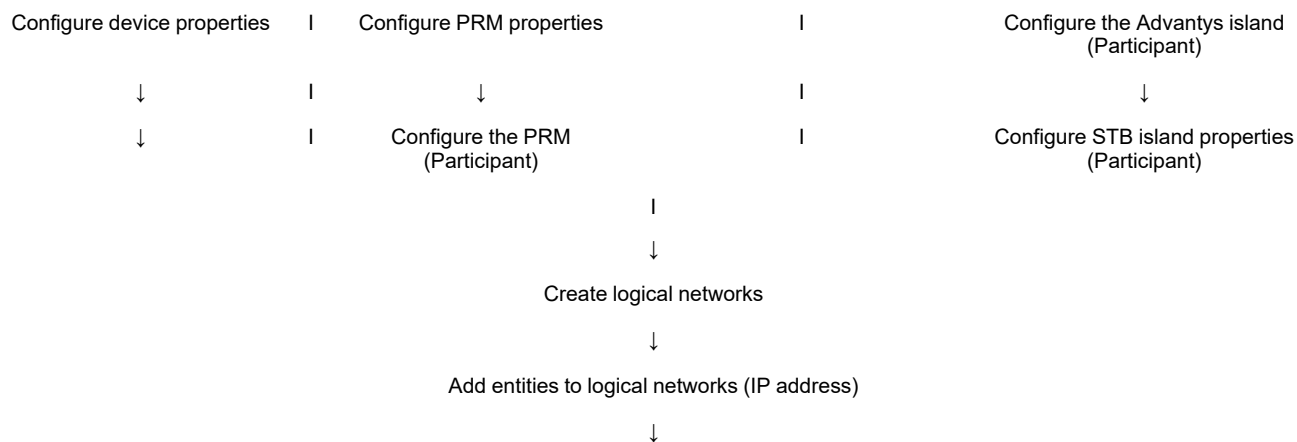


Application Explorer

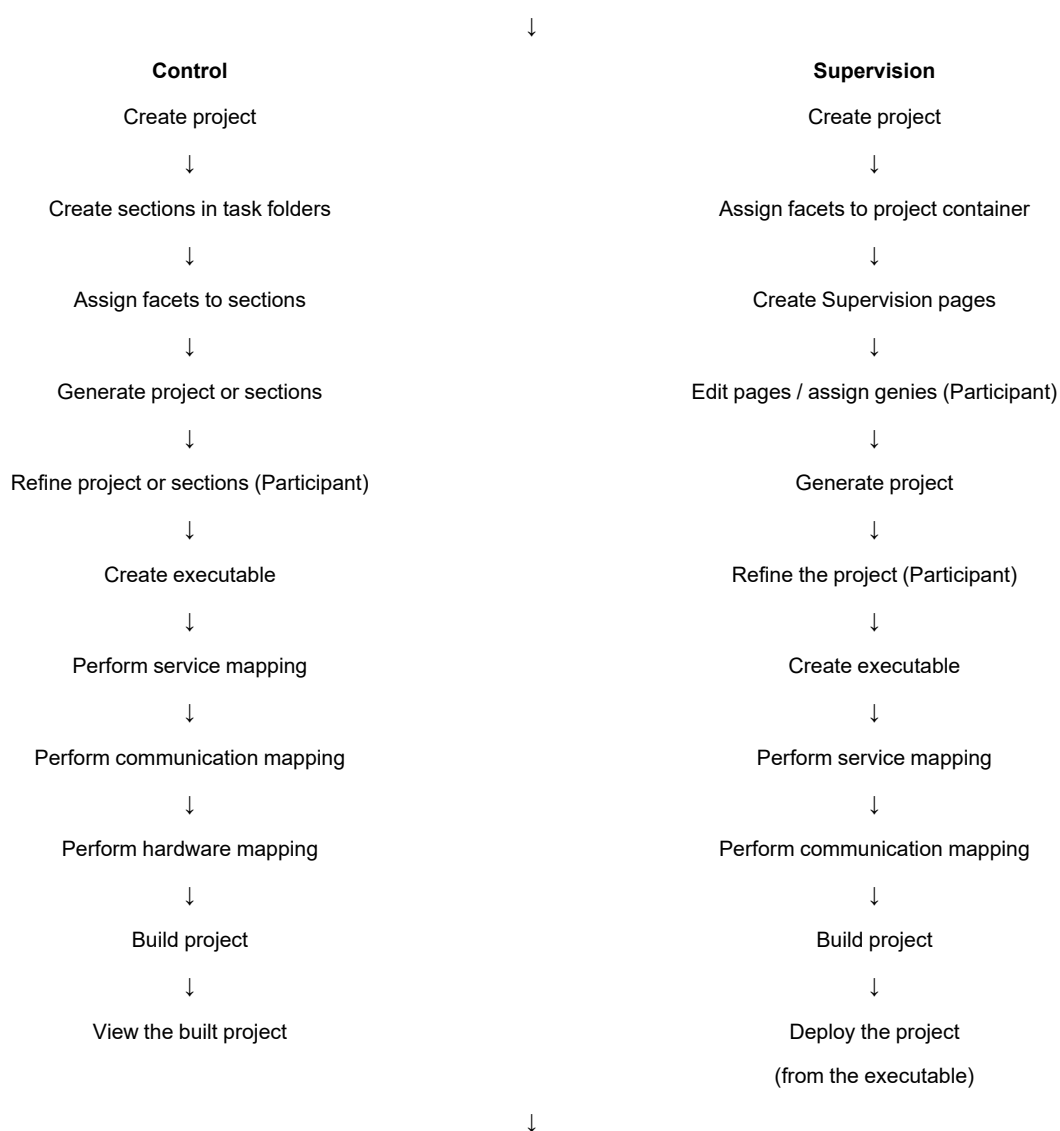


Topology Explorer

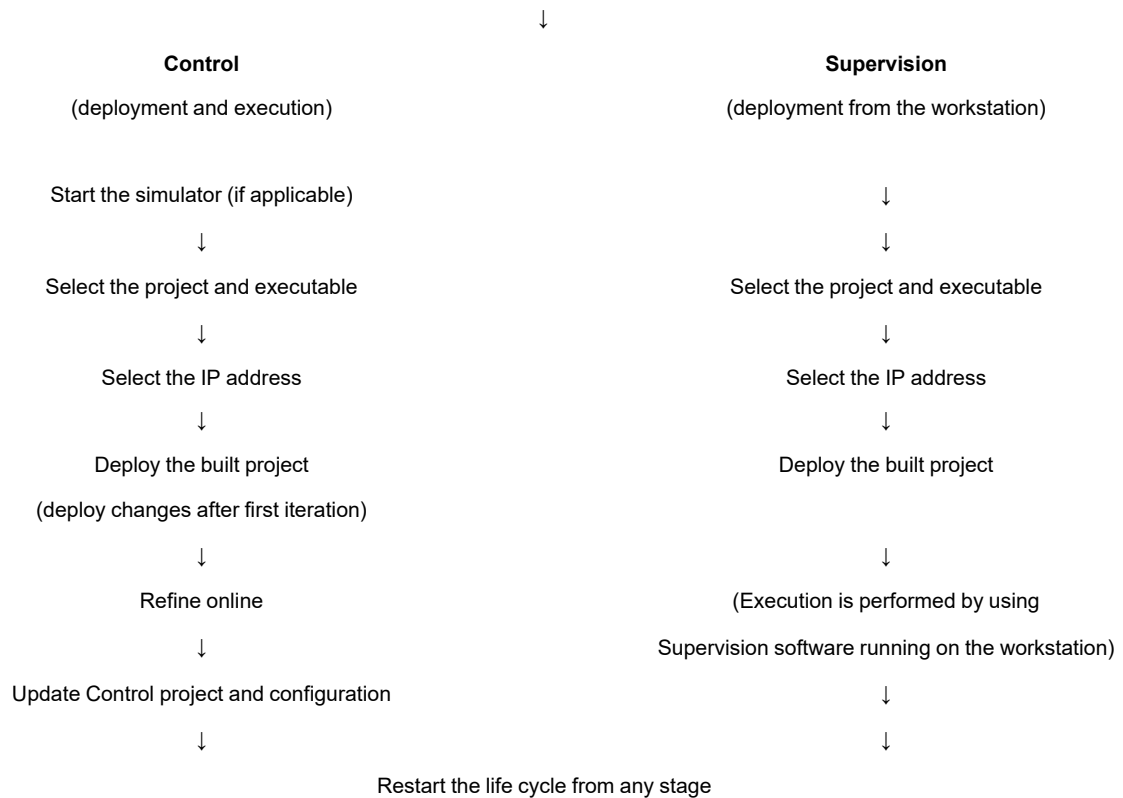




Project Explorer



Topology Explorer



NOTE: Specific transversal workflows are also provided describing how to implement communication, page 935.

System Creation Tasks

Overview

The tables in this topic provide an overview of the main tasks that you can perform at each stage of the system engineering life cycle, page 46 and where to find the corresponding information.

Stages are grouped by Explorer.

Systems Explorer Tasks

Task
Creating systems, page 134 (and configuring runtime navigation service (RTNS) parameters)
Creating system backups, page 145
Scheduling system backups, page 150
Restoring systems, page 152
Generating system engineering documentation, page 1080
Enabling and managing the System Access Password for a system, page 139.

Application Explorer Tasks

System Engineering Life Cycle Stage	Task
Instantiation stage, page 157	Browsing templates, page 163
	Creating instances, page 177
	Browsing instances and folders, page 166
	Counting instances, page 180
	Configuring instances, page 188 (and associating documents and URLs to instances for use with RTNS)
	Linking instances, page 200
	Viewing facet assignment, page 392
	Managing instances, page 210 (copy/paste, move, update/replace templates)
	Exporting and importing application objects, page 227

Project Explorer Tasks

System Engineering Life Cycle Stage	Control Projects	Supervision Projects
Project Definition stage, page 262	Creating projects, page 269 and M580 safety-related projects, page 908	Creating projects, page 300
	Managing projects and project sections, page 274 (order, export, import, manage data backup files)	Creating project components, page 304
		Managing Projects, page 317 (export, import)
		Adding and using Supervision page templates, page 359
Assignment stage, page 340	Creating sections, page 348 (automatic and manual)	Creating containers (automatic and manual)
	Assigning facets, page 375	Assigning facets, page 377
	Viewing the application structure in projects, page 384	
	Managing assigned facets, page 389 (view assignments, unassign, move, change order)	
Generation stage, page 396	Generating for the first time, page 401	
	Enabling automatic generation, page 272	Generating equipment names, page 437
	Generating after changes, page 429 (auto-assign and generate, regenerate)	
	Managing generated facets, page 442 (unassign, reassign, unlink, move)	
	Managing versions of types, page 409	–
Refinement stage, page 445	Refining projects, page 446 (viewing the application structure, viewing code generated by facets, managing device DDTs, creating LL984 segments)	Refining projects, page 508
	Checking consistency, page 460	Editing Supervision pages, page 509
	Managing Peer to Peer communication, page 475	Editing advanced project settings, adding included projects, attaching user files, page 509
	Creating network variables, page 463 (legacy)	
Mapping stage, page 719	Creating Control executables, page 721	Creating executables, page 764
	Mapping services, page 723	Mapping services, page 767
	Mapping communication, page 725	Mapping communication, page 769 (and connecting Modbus devices to Supervision)
	Mapping network variables, page 738 (legacy)	
	Mapping hardware, page 747	
	Exporting/Importing hardware mappings, page 754	
Build stage, page 773	Building projects, page 775 (also, generate and build)	Building projects, page 788 (also, generate and build)
	Viewing the built Participant project, page 780	Downloading project files, page 791
Deployment stage, page 793	Refer to the Topology Explorer task reference.	Deploying built projects, page 842 and deploying changes, page 851 (to the workstations that have a service mapping with the project executable)

Topology Explorer Tasks

System Engineering Life Cycle Stage	Control Projects	Supervision Projects
Configuration stage, page 512	Configuring controllers, page 591 and M580 safety controllers, page 908	
	Configuring workstations, page 597 (for Supervision/simulation)	
	Configuring Modbus TCP and EtherNet/IP devices, page 603	
	Configuring PROFIBUS PRM primary and Decentralized Peripherals, page 601	
	Configuring STB Islands, page 599	
	Configuring logical networks, page 576	
	Adding entities to logical networks, page 681	
	Managing the Topology, page 622 (export, import, update templates)	
	Exporting and importing I/O devices to/from CSV file, page 660	
	Managing communication, page 935 (Modbus TCP Ethernet implicit (I/O scanning, peer to peer communication)/explicit, EtherNet/IP explicit, Modbus serial)	
Deployment stage, page 793	Using the controller simulator, page 799	Deploying built projects, page 842 and deploying changes, page 851 (to specific workstations)
	Managing controller passwords for deployment and execution operations, page 804	
	Deploying built projects, page 794	
	Re-deploying projects, page 819	
	Deploying project changes, page 830 (and undoing online changes)	
	Deploying controller data, page 824	
Execution stage, page 854	Starting/stopping controllers, page 861	Refer to Executing the Deployed Supervision Project (see <i>EcoStruxure Process Expert, Supervision Participant Services, User Guide</i>)
	Synchronizing redundant controllers, page 866	
	Refining online, page 874 (and importing generated changes, viewing the application structure)	
	Updating the Control project and the configuration, page 897	
	Backing up controller data, page 880	

Global Templates Explorer Tasks

System Engineering Life Cycle Stage	Tasks
Global Templates, page 990	Managing Global Templates, page 1035 (save, copy, export/import, purge)
	Updating, duplicating, and replacing Global Templates, page 1051
	Creating Global Templates (see <i>EcoStruxure Process Expert, Global Templates Reference Manual</i>)
	Templatizing Supervision animated graphics (see <i>EcoStruxure Process Expert, Global Templates Reference Manual</i>)
	Enabling and managing the Control facet template password, page 1002

Content Repository Tasks

Tasks
Creating content containers (see <i>EcoStruxure Process Expert</i> , <i>Runtime Navigation Services</i> , <i>User Guide</i>) (and editing, copying, moving, exporting and importing them to set up runtime navigation services (RTNS))

Software Participant Tasks

Tasks
Configuring the number of Control Participant instances (see <i>EcoStruxure Process Expert</i> , <i>Installation and Configuration Guide</i>) that you can open per computer .
Installing extensions in the Control Participant (see <i>EcoStruxure Process Expert</i> , <i>Installation and Configuration Guide</i>) (DTMs, GSD and EDS files, hotfixes)
Opening the Help of a Participant, page 95, OPC Factory Server, and OPC UA Server Expert

Managing Systems

What's in This Chapter

Organizing Systems	135
Creating Systems	137
Managing the System Access Password	141
Creating System Backups	145
Scheduling System Backups	150
Restoring Systems	152

Overview

This chapter describes how to manage systems by using the **Systems Explorer**. Creating a system is prerequisite to starting the system engineering life cycle.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Organizing Systems

Overview

Systems that you create are stored in the database of the system server.

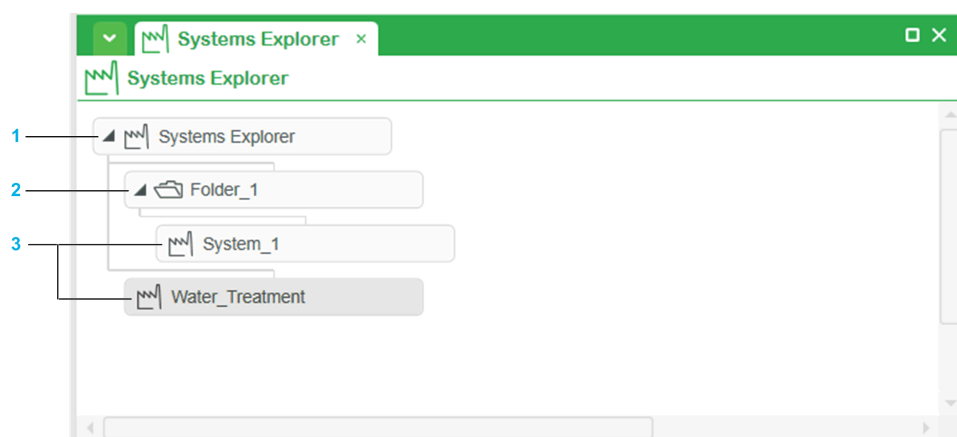
You can access a system from any client that is connected to the server of this infrastructure.

Accessing the Systems Explorer

To access the **Systems Explorer**, click the **Systems** icon in the toolbar, page 104.

Systems Explorer Window

The following figure shows the **Systems Explorer** window.



Item	Description
1	Systems Explorer root folder
2	System folder
3	Systems

Creating System Folders

To organize systems, you can create system folders that can contain one or more systems.

To create a system folder, right-click the **Systems Explorer** root folder and select **Create Folder**.

System Folder Actions

Right-click the root folder or any system folder to open a context menu with the following commands.

Command	Description
Create Folder	Creates a system folder or subfolder.
Create System	Creates a system.
Delete	Deletes the folder. The command is not available from the root folder. NOTE: The command is not available if the folder is not empty.
Rename	Allows you to enter a new folder name. The command is not available from the root folder. NOTE: You can also click the folder twice to edit its name.
Restore	Opens the Restore window, which allows you to select a system backup file (.sbf) to restore a system, page 152 in the database.

Creating Systems

Creating a System

To create a system, right-click the **Systems Explorer** root folder or a system folder and select **Create System**.

System Actions

Right-click a system to open a context menu with the following commands.

Command	Description	
Open Application ⁽¹⁾	Opens the Application Explorer , which allows you to start the Instantiation stage, page 157 or access the applications of an existing system.	
Open Topology ⁽¹⁾	Opens the Topological Explorer , which allows you to start the Configuration stage, page 512 or access the topology of an existing system.	
Open Projects ⁽¹⁾	Opens the Projects Explorer , which allows you to start the Project Definition stage, page 262 or access the Control and Supervision projects of an existing system.	
System Access Password	Opens a submenu containing the following commands. The menu is only available if the System Access Password property, page 139 of the system is enabled.	
	Manage Password	Opens the Manage Password dialog box, which lets you set and change the system password, page 142.
	Clear Password	Opens the Clear Password dialog box, which lets you remove the system password, page 142.
	Forgot Password	Opens the Forgot Password dialog box, which contains an authentication code that you need to reset the forgotten password, page 143.
Delete	Deletes the system including all of its contents. The command is inactive if either an application, topology, or project window of the system is still open.	
Rename	Allows you to enter a new system identifier. NOTE: You can also click the system slowly twice to edit its identifier	
Back Up	Opens the Back Up window, page 145, which allows you to create a backup file of the system.	
Generate Report	Opens the Generate Report window, which lets you create and save to disk a report, page 1080 that contains engineering information about the selected system or one or more of its domains (such as topology or Control projects), including Participant project documentation. All windows in the engineering client, except the Systems Explorer , need to be closed to use the command.	
Settings	Opens the Settings window, page 138, which lets you view and configure system-related settings.	
Properties	Opens the Properties, page 139 window of the system.	
(1) You can also use the navigation buttons, page 104 located in the toolbar of the engineering client.		

System Settings Window

The **Settings** window opens when you select the **Settings** command in the context menu of a system.

Section	Description
Document Viewers	Lets you enter the command line that EcoStruxure Process Expert executes for a given document type. The command is executed when you open, in the operation client, the document that is associated to an instance by using the HyperLink service, page 191.

System Properties

To open the **Properties** window of the system, right-click the system and select **Properties**.

You can view and/or edit the following items.

Item	Description
Identifier	The identifier must be unique on the system server. Default value: <i>System_n</i> where <i>n</i> is an incremental number starting at 1.
Description	Optional. You can enter a description of the system with free form text. The description you enter appears in a tooltip when you move the pointer over the system. Default value: Blank
Market	Optional. You can enter a description of the market that the system pertains to with free-form text. Default value: Blank
Customer	Optional. You can enter a description of the customer that the system pertains to with free-form text. Default value: Blank
Site	Optional. You can enter a description of the site that the system pertains to with free-form text. Default value: Blank
Created By	Logged-in user who has created the system.
Creation Date Time	Date and time of the system creation.
Modified By	Logged-in user who has modified the system properties.
Modification Date Time	Date and time of the last modification to the system properties.
Change Description	Optional. You can enter a description of the changes to the system with free form text. Default value: Blank

Item	Description		
Security	Opens a submenu containing the following properties. The properties are enabled and read-only when the Control application and facet template password protection setting (see EcoStruxure Process Expert, Installation and Configuration Guide) is enabled. The setting can be configured in the Control Passwords section of the System Server Configuration Wizard .		
	<table><tr><td>Controller Access Password at Creation (formerly Optional Security Services By Default)</td><td><p>The setting has an impact on the default value of the Controller and Simulator password property of controllers, page 592 and workstations acting as simulator, page 598 that you create in the system.</p><p>Possible values:</p><ul style="list-style-type: none">• Enabled (default): Sets their property to Enabled by default.• Disabled: Sets their property to Disabled by default.<p>NOTE: Modifying the setting does not change the value of the password property of existing workstations and controllers of the system.</p><p>NOTE: This setting does not impact the Safety password property of M580 safety controllers, page 925.</p></td></tr></table>	Controller Access Password at Creation (formerly Optional Security Services By Default)	<p>The setting has an impact on the default value of the Controller and Simulator password property of controllers, page 592 and workstations acting as simulator, page 598 that you create in the system.</p> <p>Possible values:</p> <ul style="list-style-type: none">• Enabled (default): Sets their property to Enabled by default.• Disabled: Sets their property to Disabled by default. <p>NOTE: Modifying the setting does not change the value of the password property of existing workstations and controllers of the system.</p> <p>NOTE: This setting does not impact the Safety password property of M580 safety controllers, page 925.</p>
	Controller Access Password at Creation (formerly Optional Security Services By Default)	<p>The setting has an impact on the default value of the Controller and Simulator password property of controllers, page 592 and workstations acting as simulator, page 598 that you create in the system.</p> <p>Possible values:</p> <ul style="list-style-type: none">• Enabled (default): Sets their property to Enabled by default.• Disabled: Sets their property to Disabled by default. <p>NOTE: Modifying the setting does not change the value of the password property of existing workstations and controllers of the system.</p> <p>NOTE: This setting does not impact the Safety password property of M580 safety controllers, page 925.</p>	
<table><tr><td>System Access Password</td><td><p>Lets you enable the System Access Password context menu of the system, page 137, which lets you set and manage a password, page 141 at the system level.</p><p>Possible values:</p><ul style="list-style-type: none">• Enabled: After enabling the setting, you are required to set a password, page 141 for the system in the Manage Password dialog box to open one of its explorers or generate a report.• Disabled (default): You cannot set a password for the system.<p>To disable the property when a password has been set, you must enter the password, which also clears it, page 142. To perform this operation, all other clients connecting to the system server must be closed and no other tab can be open in this engineering client.</p></td></tr></table>	System Access Password	<p>Lets you enable the System Access Password context menu of the system, page 137, which lets you set and manage a password, page 141 at the system level.</p> <p>Possible values:</p> <ul style="list-style-type: none">• Enabled: After enabling the setting, you are required to set a password, page 141 for the system in the Manage Password dialog box to open one of its explorers or generate a report.• Disabled (default): You cannot set a password for the system. <p>To disable the property when a password has been set, you must enter the password, which also clears it, page 142. To perform this operation, all other clients connecting to the system server must be closed and no other tab can be open in this engineering client.</p>	
System Access Password	<p>Lets you enable the System Access Password context menu of the system, page 137, which lets you set and manage a password, page 141 at the system level.</p> <p>Possible values:</p> <ul style="list-style-type: none">• Enabled: After enabling the setting, you are required to set a password, page 141 for the system in the Manage Password dialog box to open one of its explorers or generate a report.• Disabled (default): You cannot set a password for the system. <p>To disable the property when a password has been set, you must enter the password, which also clears it, page 142. To perform this operation, all other clients connecting to the system server must be closed and no other tab can be open in this engineering client.</p>		

Managing the System Access Password

Overview

The system access password that you manage for each system by using its context menu has the following purpose:

- It is used as application password in the Control Participant projects of the system (projects and topology domains).
- It enables file encryption for these Control projects.

NOTE: For an overview of the entire Control project password management functionality, refer to the topic describing the system and Control facet template passwords (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

Prerequisites

The following are prerequisites to setting and managing the system access password:

- The **System Access Password** property, page 139 of the system is enabled.
- No other tab than the **Systems Explorer** is open in the engineering client.
- No other client is connected to the system server.

Scope of the System Access Password Protection

Once you have set the password, you are required to enter it to perform the following actions:

- Open an explorer of the system.
- Generate a system documentation report.
- Create a system backup that is forward compatible.

NOTE: If you have already entered the password once, you are not required to enter it again until you close the engineering client or restart the system server.

System Access Password Requirements

To be valid, the system access password must contain the following:

- Between 8 and 16 characters.
- At least one uppercase letter from the classical Latin alphabet (A...Z).
- At least one lowercase letter from the classical Latin alphabet (a...z).
- At least one base-10 digit (0...9).
- At least one special character (~, !, @, \$, %, ^, &, *, _ , +, -, =, ` , |, \, (,), [,], ;, :, " , ' , < , > , { , } , , , #).

Setting the System Access Password

You can set the system access password either:

- By using the **Manage Password** command of the **System Access Password** context menu of the system.
- By opening an explorer of the system (for example, the **Application Explorer**).

Changing the System Access Password

To change the system access password, proceed as follows.

Step	Action
1	In the Systems Explorer , right-click the system and click System Access Password > Manage Password . Result: The Manage Password dialog box opens.
2	Enter the existing password in the Current Password text box.
3	Enter the new password and confirm the new password in the corresponding text boxes.
4	Click OK . Result: The new password is applied to the Control Participant projects of the system.

Clearing the System Access Password

Clearing the system access password has the following impact:

- Clears the application password of Control Participant projects of the system.
- Disables file encryption for these Control Participant projects.

NOTE: You cannot clear the password when the **Control application and facet template password protection** setting (see EcoStruxure Process Expert, Installation and Configuration Guide) is enabled at the system server level.

To clear the system access password, proceed as follows.

Step	Action
1	In the Systems Explorer , right-click the system and click System Access Password > Clear Password . Result: The Clear Password dialog box opens.
2	Enter the existing password in the Current Password text box.
3	Click OK . Result: The Confirm Clear Password dialog box opens.
4	Click OK . Result: The password is cleared.

Resetting a Forgotten System Access Password

To reset the system access password, proceed as follows.

Step	Action
1	In the Systems Explorer , right-click the system and click System Access Password > Forgot Password . Result: The Forgot Password dialog box opens and an authentication code is displayed.
2	Copy the authentication code, contact Schneider Electric support, page 106, and provide the code. Result: You receive a temporary password.
3	Right-click the system and click System Access Password > Manage Password.. . Result: The Manage Password dialog box opens.
4	Follow the procedure to change the password , page 142 and enter the temporary password in the Current Password text box. NOTE: If you attempt to enter the temporary password when prompted for the existing password (for example, when opening an explorer), you are asked to change the password first. You cannot clear the forgotten password by using the temporary password.

Password Management When Exporting and Importing Control Projects

When you export a Control project, its application password is removed and file encryption is disabled.

When you import a Control project, an application password is set and file encryption is enabled if the **System Access Password** property, page 139 of the system in which you import is enabled.

Password Management When Importing Control Expert Projects

When you import a Control Expert project, an application password is set and file encryption is enabled if the **System Access Password** property, page 139 of the system in which you import is enabled.

Password Management When Importing and Exporting Application and Topology Objects

When you export instances or topological entities, if the export file contains Control constituents (.stu), their application password is removed and file encryption is disabled.

When you import instances or topological entities, an application password is set and file encryption is enabled for any Control constituents (.stu) if the **System Access Password** property, page 139 of the system in which you import is enabled.

File Encryption Management for Already Deployed Control Projects

If you enable the **System Access Password** property for the first time and the **Controller** password has already been set for a controller, the already deployed Control project is not impacted. That is, file encryption is not enabled.

This can be the case, for example, after restoring the database of a previous version.

To apply file encryption to the deployed project, you need to deploy the Control project again after setting the system access password. This requires stopping the controller.

Password Management When Saving Control Participant Project Files from the Content Repository

When you [save to disk, page 1123](#) a Control Participant project file (.stu) from the content repository (of a Control project or controller) and you have set a system access password for the corresponding system, the application password protection and file encryption remain in effect for this project file.

You can change its application password but you cannot remove it. Also, you cannot disable file encryption.

Creating System Backups

Overview

You can create a backup file (.sbf) of systems that exist in the **Systems Explorer**. This allows you to do the following:

- Back up your work on demand to help preserve your data in case of a computer malfunction (refer to the topic describing the [content of a system backup](#), page 147).
- Restore your work to an earlier point in time.
- Deliver a system to a site.
- Make a system forward compatible to reuse it with a subsequent supporting version (manual backup only).
- Create an export file (.sbk) that contains the templates that are used by a system (manual backup only).

You can create system backups in the following ways:

- Manually, by using the **Back Up** command from the context menu of the system in the **Systems Explorer**. You can create these backups to be forward compatible.
- Automatically, by using the **System Backup Scheduler**, page 150 command from the **Settings** menu of the system server (requires system server access rights (see *EcoStruxure Process Expert, Installation and Configuration Guide*)).

A backup of a system cannot be created if a component of the system is open in an engineering or operation client (for example, if the **Application Explorer** is open or if system data is accessed by using runtime navigation services or the **Global Templates Explorer**). During a system backup, no user can open the content repository.

Back Up Window Description

The following figure shows an example of the **Back Up** window.

System_1: Back Up

Select destination path then save

1 — File name: C:\Users\User1\Documents\System_1 (20230522131404).sbf

2 — ☐ Back up templates used by the system

3 — ☐ Make backup forward compatible

Change Description

4 —

! Global templates are not included in the backup file by default. They must be in the database to restore the system.

Save Cancel

Item	Description
1	System backup file name and path, page 148.
2	<p>When selected, a template export file (.sbk) is created at the same location and with the same name as the system backup file (.sbf). It contains the templates that are used by topological and application instances of the system (for example, \$MotorGP (4.2.0)). It also contains the Included projects that are used by the existing Supervision projects.</p> <p>If required, these templates are imported when you restore, page 152 the system.</p> <p>NOTE: If the Control constituent password, page 1002 is set, selecting this option increases the time required to create and restore the backup file significantly.</p>
3	<p>When selected, the system backup file is forward compatible.</p> <p>Selecting this option increases the time required to create and restore the backup file significantly because system data undergoes a pre-migration process.</p> <p>NOTE: If the check box is disabled, you can enable it by installing a premigration patch that is included with the next supporting release of EcoStruxure Process Expert.</p>
4	<p>System description. Free-form text (optional).</p> <p>The description is shown when you restore the system.</p>

Version and Language Compatibility of System Backups

The following table describes on which version/language you can restore a system by using a system backup that is created with the installed version.

System backup is forward compatible	Restore on the same version	Restore on subsequent supporting versions	Restore on a different language
No	Yes	No. It can also be restored on subsequent releases of the same version (<i>R•</i> versions, for example, 2020 R2) and service packs of the same version only if the Participant version is the same.	Yes
Yes	Yes	Yes. NOTE: Forward compatible system backups created with version 2020 R2 cannot be restored on version 2021.	No ⁽¹⁾
(1) The system can be restored on an infrastructure of a different language only if it is restored on the same version.			

When you back up a system, the version of EcoStruxure Process Expert is recorded and displayed when you restore the system.

System Backup File Contents

By default, a system backup file contains the following data:

- The definition of the topological and application instances used in a system.
- Physical views, page 536.
- Participant project and executable files with the related mapping information.
- Controller data backup files, page 296.
- Supervision included projects, page 791 added by users.
- User files, page 511 added to Supervision projects.
- OPC Factory Server and OPC UA Server Expert configuration files, page 791.
- The setting of the **System Access Password** property, page 139 and the password (if set).
- Password protection properties, page 592 for topological entities and their value.
- Content containers, page 1121 that you created in the **User Contents** folder structure of the system.

NOTE: By default, a system backup does not contain templates that are used by the instances of the applications and the topological entities of the system. You can only restore the system, page 152 if the templates are in the EcoStruxure Process Expert database. If required, select to back up templates or export templates, page 1039 manually from the root folder of the **Global Templates Explorer**.

NOTE: Extensions of the Control Participant that you installed locally, such as DTMs, are not included in the system backup file. Ensure that the same extensions are installed on the computers on which the system is restored or opened to perform operations on Control projects (for example, generating projects).

System Backup File Name

A system backup consists of a file with the default name `<system identifier> <(time stamp)>.sbk` where *time stamp* is the date and time when the backup file was created, displayed by using the format `YYYYMMDDHHMMSS`. Time is shown in the 24-hour format.

For example, `System_1 (20160625144852)`.

You can edit the file name when you create the backup.

NOTE: For scheduled backups, page 150, a *time stamp* suffix is automatically added to the file name that appears in the **Backup Path** text box. The time stamp information corresponds to the point in time when the backup file is created.

Managing Passwords When Creating Forward Compatible System Backups

If you make the system backup forward compatible and either password is set, you must enter it to proceed with the backup:

- The system access password, page 141.
- The controller access password, page 139 for controllers of the system. For workstations where a Control Expert service is created, entering the password is optional.

If you back up the templates and the Control constituent password, page 1002 is set, you must enter it to proceed with the backup.

NOTE: If either password was already entered once since the engineering client was started, you are not prompted to enter it.

Canceling System Back Up Tasks

After clicking **Save** in the **Back Up** window, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The backup file is not created.

Only the user who selected the command is allowed to cancel it.

NOTE: The creation of a forward compatible backup cannot be canceled once the premigration of data has started.

NOTE: Scheduled backups cannot be canceled once started.

Creating a System Backup Manually

To create a system backup manually, proceed as follows.

Step	Action
1	Right-click the system in the Systems Explorer and select Back Up . Result: The Back up window opens.
2	To create the back up in a different location and/or with a different file name, click the browser button; otherwise, proceed to step 4. Result: A Save As dialog box opens.
3	Select a location where to create the back up file and edit the file name if needed.
4	Select the check box to make the system backup forward compatible and/or back up templates, as needed. NOTE: Forward compatible systems can also be restored on the same version.
5	Enter a description (optional).
6	Click Save . Result: The system backup file is created and information is displayed in the notification panel. If you chose to back up templates and/or make the backup forward compatible and if a password is set, page 148, you must enter it; otherwise, the backup is not created. NOTE: To cancel the backup task, click the Abort icon in the notification panel.

NOTE: Do not run several forward compatible system backups in parallel on one or more clients connected to the same system server. This may cause system backups to not complete successfully.

Creating System Backups Automatically

To schedule the creation of system backups, proceed as follows.

Step	Action
1	On the system server computer, open the System Backup Scheduler (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>), select a system, and enable backups, page 150.
2	Configure the backup settings and select a backup file location.
3	Click Save and close the scheduler. Result: A backup file of the system is created at the configured dates and time. Relevant information is displayed in the notification panel of engineering clients that are connected to the system server.

Scheduling System Backups

Overview

The **System Backup Scheduler** lets you schedule the automatic backup of systems, page 145.

Each system can have its own backup schedule.

The system server must be running to configure schedules and perform backups.

Version Compatibility

System backup files created by using the **System Backup Scheduler** are not forward compatible, page 147.

Backup Notifications

Ten minutes before the scheduled backup time, a notification about the scheduled backup operation is displayed in the notification panel of engineering clients that are connected to the system server and in the server console.

After the backup is completed, the system displays a notification about the backup creation status.

Managing Backup File Storage Location

Depending on the selected schedule and the size of the systems, backup files can use a lot of disk space. Manage storage folders regularly to make sure sufficient disk space remains available for new backup files.

System Backup Scheduler Description

The following figure shows an example of the **System Backup Scheduler** window.

The screenshot shows the 'System Backup Scheduler' window. It has a title bar with a question mark and close button. The window is divided into two main sections: 'System' and 'Settings'.

1. In the 'System' section, there is a 'Select System' dropdown menu currently showing 'System_1'.

2. In the 'Settings' section, there is a checked 'Enabled' checkbox.

3. Below the checkbox is a 'Frequency' dropdown menu set to 'Weekly'.

4. Below the frequency is a 'Day Of Week' section with checkboxes for Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday. Monday and Wednesday are checked.

5. Below the days is a 'Backup Time (hh:mm)' section with two dropdown menus showing '19' and '20'.

6. Below the time is a 'Backup Path' text field containing 'C:\backup\System_1.sbf' and a browse button (three dots).

7. At the bottom right are 'Save' and 'Cancel' buttons.

8. At the bottom left is a 'View Logs' link.

Numbered callouts 1 through 9 point to these specific elements: 1 points to 'Select System', 2 to 'Enabled', 3 to 'Frequency', 4 to 'Day Of Week', 5 to 'Backup Time', 6 to the browse button, 7 to 'Save', 8 to 'Cancel', and 9 to 'View Logs'.

Item	Description
1	Lets you select an existing system to configure its backup settings and file location.
2	<p>Enabled (check box selected): A backup schedule can be configured and backups are performed accordingly once you save changes.</p> <p>Disabled (check box cleared, default value): No backup is performed. If backup settings were configured, disabling backups and saving the change discards the settings.</p> <p>The check box applies to the system that is selected.</p>
3	<p>Backup frequency.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Daily • Weekly • Monthly
4	<p>Backup days.</p> <ul style="list-style-type: none"> • For daily backups: No setting is available. • For weekly backups: You must select at least one day of the week and a time. • For monthly backups: Select a date. For months with less than 31 days, the backup is performed the last day of the month.
5	<p>Time (hour and minutes in 24-hour format) when the backup is performed based on the time setting of the system server computer.</p> <p>NOTE: If the backup cannot be performed, for example, because an editor of the system is open, EcoStruxure Process Expert retries two times at 10-minute intervals.</p>
6	<p>Opens the Save As dialog box, which lets you select a folder where backup files are created and a backup file name, page 148.</p> <p>Default value: <System identifier>.dbf in the <i>Documents</i> folder of the logged-in user on the system server computer.</p> <p>If you select a network folder, ensure that the system server computer can access the folder at the scheduled times.</p>
7	Saves the backup settings for the selected system.
8	Discards unsaved changes.
9	Opens a window that lists the last 20 automatic backup attempts, which did not complete successfully. The list is for all systems and contains the description of the detected issues.

Restoring Systems

Overview

In the **Systems Explorer**, you can restore a system from a system backup file that you created by using the **Back Up** command or the **System Backup Scheduler**.

Restoring a system requires that templates that are used by the instances of applications and the topological entities are present in the EcoStruxure Process Expert database. If you also backed up the templates, page 145, they are imported as needed.

You cannot merge system data. If a system with the same identifier already exists, the system is restored with a new identifier.

You cannot restore several systems in parallel on one or more clients connected to the same system server.

NOTE: Restoring a system and the templates that it uses when a **Control constituent password**, page 1002 is set or restoring a system created with an earlier version of EcoStruxure Process Expert takes time.

Version and Language Compatibility

Backup files are version-dependent unless you made them forward compatible, page 145 at the time of creation.

Forward compatible backup files are language-dependent.

For details, refer to the topic describing the version and language compatibility of system backups, page 147.

NOTE: Forward compatible system backups created with version 2020 R2 cannot be restored on version 2021.

Detection of Missing Data Before System Restoration

When you restore a system, if EcoStruxure Process Expert detects that an object (for example, a template) is missing in the database, page 147, it opens the **Restore** dialog box. Missing objects are reported in the **Status** column of this dialog box.

To restore the system, you must add the missing object on the database where you want to restore it.

If an **Include** project is missing, you can add it to the content repository by following the steps describing how to add a project file (see *EcoStruxure Process Expert, Global Templates Reference Manual*). Start from an existing Supervision genie facet template or create a new one from the **Toolbox** of the **Global Templates Explorer**.

NOTE: If you selected to back up the templates, page 146 during system backup, the templates contained in the template export file (.sbk) that are not present in the database are automatically imported before restoring the system.

Password Management When Restoring a System

The following table describes the impact on system access and Control constituent application password protection when you restore a system.

Restored system	Infrastructure on which system is restored	Result for system after restore
The system access password, page 141 is set.	The Control application and facet template password protection setting is enabled or disabled.	The System Access Password setting remains enabled and the password set. NOTE: You must enter the password during restoration.
The system access password is not set (setting disabled).	The Control application and facet template password protection setting is enabled.	The System Access Password setting is enabled and you must set a password.
The system is restored with the template export file (.sbk), page 145 present.	The Control constituent application password protection setting is enabled and a password is set.	The Control constituent application password is applied to imported Control facet templates. NOTE: You must enter the password during restoration.
Is restored without template export file.		No impact.

Password Management After Restoring a System

After you restore a system that contains a controller and/or a workstation acting as simulator for which password protection, page 592 is enabled, you must do the following to deploy a Control Participant project:

- If no password was set before creating the backup file, you must set one.
- If a password was set before creating the backup file, you must enter this password when prompted.

For more information, refer to the topic describing how to manage passwords for Control Participant projects, page 804.

Canceling System Restoration

After clicking **Restore** in the **Restore** window, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task.

Only the user who selected the command is allowed to cancel it.

Restoring a System

When you restore a system, its name in the database may be different from the name it had when the backup was created. If this occurs and you had already deployed Supervision projects contained in this system when you created the backup file, the system name that was used as reference by runtime navigation services now refers to a different system. As a result, runtime navigation services may not function or display data of another system.

NOTICE

NAVIGATION TO DIFFERENT SYSTEM

Redeploy Supervision Participant projects after restoring a system.

Failure to follow these instructions can result in equipment damage.

NOTE: For details, refer to the topic describing the impact of restoration on system names in the [database](#), [page 155](#).

To restore a system from a backup file, proceed as follows.

Step	Action
1	Right-click the system root folder or a system folder in the Systems Explorer and select Restore . Result: The Restore window opens.
2	Browse to the system backup file (.sbf) and click Open . Result: A confirmation dialog box opens.
3	Review the information that is displayed and click Restore . Result: If you selected to back up the templates and a Control constituent password , page 1002 is set on the system server of the infrastructure and/or the system access password is set on the system, you are prompted for the corresponding passwords; otherwise, the templates contained in the template export file (.sbk) are imported and the system is restored. NOTE: If a system with the same identifier already exists, the new identifier that will be used to restore the system is indicated in the confirmation dialog box.
4	In the Verify Password dialog boxes that open, enter the Control constituent and system access passwords that are set and click OK . Result: The templates contained in the template export file (.sbk) are imported and the system is restored. NOTE: Click Cancel to close the Verify Password dialog box without restoring the system.
5	If the system that you restored contains Supervision Participant projects that were deployed before creating the backup, redeploy each one of them.

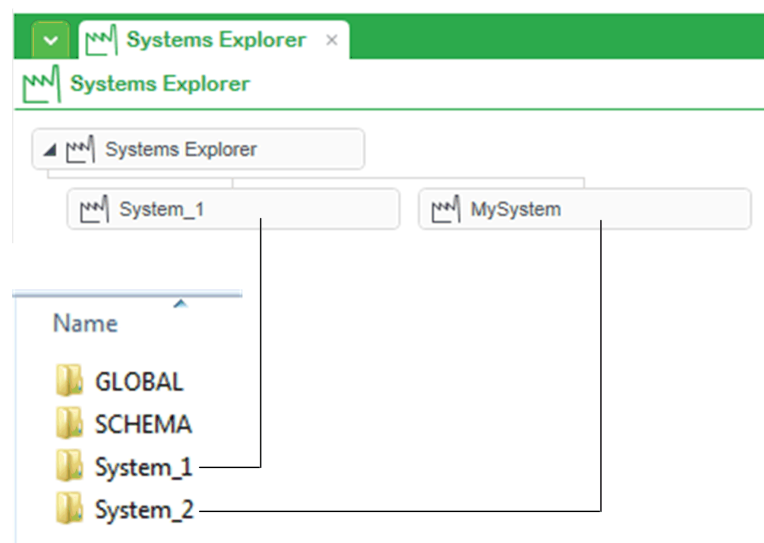
Impact of the Restoration on System Names in the Database

Each system that you create has an identifier in the **Systems Explorer** and a name in the database, which may be different. Names of systems in the database are assigned incrementally by EcoStruxure Process Expert and used internally for system identification purposes, for example, by runtime navigation services provided by the operation client. The name in the database does not change when you rename a system in the **Systems Explorer**.

When you restore a system in the same or another EcoStruxure Process Expert infrastructure, if the name that was originally assigned to it is already used, the system is restored with same identifier in the **Systems Explorer** but with the next available name in the database.

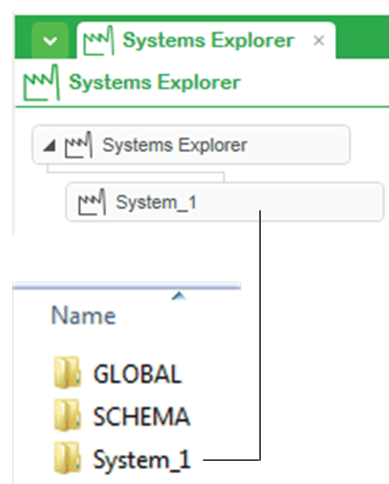
The following example illustrates the scenario where the name of a system in the database changes after restoring it.

The figure shows two systems that exist in the **Systems Explorer** and their respective name in the database folder.

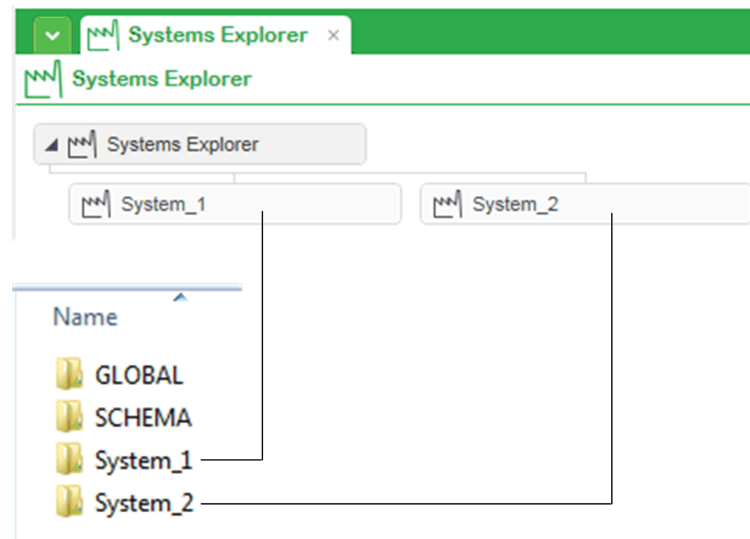


NOTE: The path to the database folder is *C:\ProgramData\Schneider Electric\Process Expert x\Db*, where *x* corresponds to the installed major version (the *R•* version suffix does not appear).

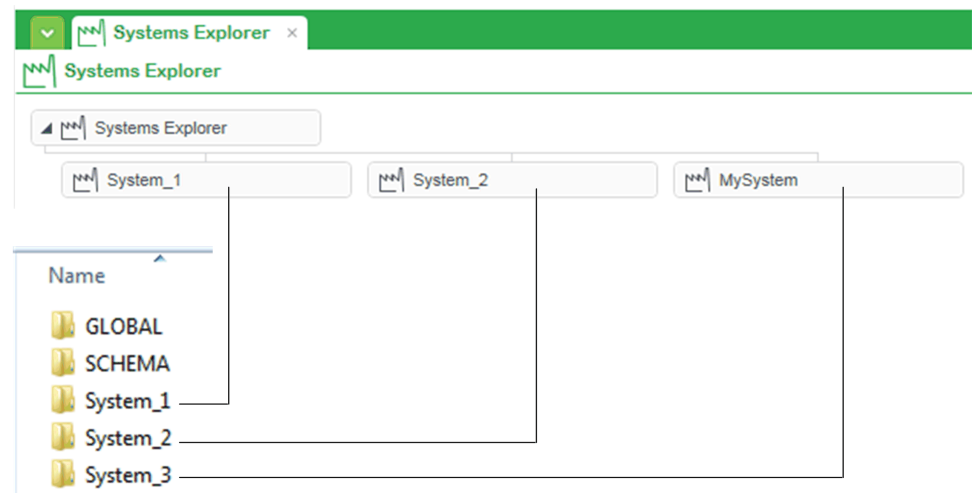
MySystem is backed up and deleted from the **Systems Explorer**. Its name in the database is also removed and becomes available for assignment to a new system.



Then, a new system (*System_2*) is created. EcoStruxure Process Expert assigns it the next available name in the database. In this case, it corresponds to the name that was originally assigned to *MySystem*.



Later, *MySystem* is restored. EcoStruxure Process Expert assigns it the next available name in the database, which is different from its original name.



As a result, in this example, the name of *MySystem* in the database:

- Was *System_2* before restoration.
- Is *System_3* after restoration.

Instantiation Stage

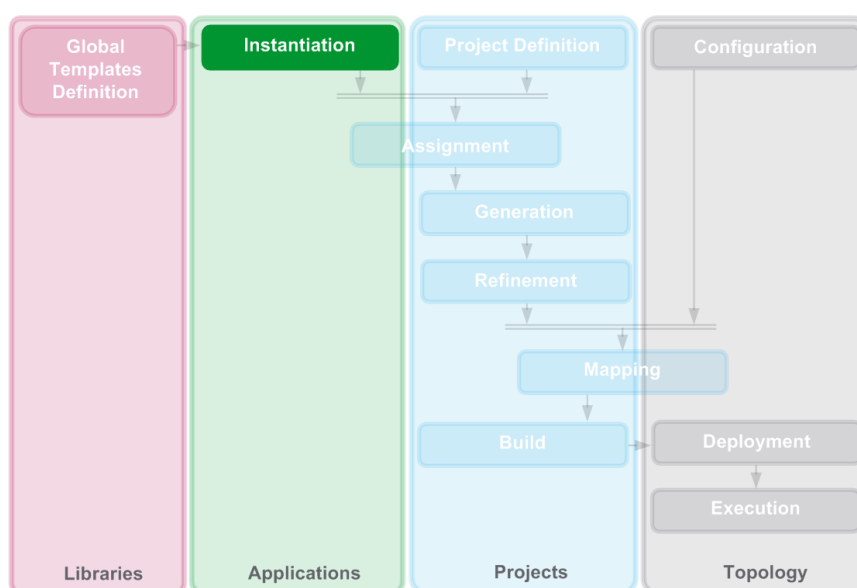
What's in This Chapter

Instantiating Templates	158
Instance Indicators	195
Linking Instances by Using the Asset Workspace Editor	200
Managing Application Folders and Instances	210
Exporting and Importing Application Objects to/from CSV and XML Formats	227

Overview

This chapter contains information to help you create the applications of the system according to the Process and Instrumentation Diagram (PI Diagram).

The following figure shows the position of the Instantiation stage within the system engineering life cycle.



Refer to the Instantiation stage, page 49 for a description of the purpose of this stage.

Instantiating Templates

Overview

This section describes how to use the **Application Explorer** to:

- Create a folder structure representing the process hierarchy of applications.
- Create and configure instances by instantiating application templates to model the functional components of an application.
- Link instances to each other to propagate data.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Organizing the System Application

Before Starting

Before creating the folder structure for your system application, define the appropriate segmentation, hierarchy, and naming convention for your process.

The **Application Explorer** lets you use templates of the Global Templates Library to create instances that model typical components and functions of the applications of a system. For example:

- **Process objects:** Represent the functions of process hardware, for example, motors and valves. This includes signals from the field and/or variables.
For example, the *\$Motor* template can model the functions of the electrical motor of a pump.
- **Device objects:** Provide functions required to communicate with, control, and/or diagnose a range of electrical devices and instrumentation that are connected to field buses.
For example, the *\$TeSysTEM* template can model the functions of a TeSys T variable speed drive communicating through Modbus TCP explicit messaging.
- **Communication objects:** Provide functions to manage communication with device objects connected to field buses.
For example, the *\$EMPortQ* template can model the management of communication through the port of a Quantum controller through Modbus TCP explicit messaging.

Refer to *Browsing Application Templates*, page 163 for additional information on templates that you can instantiate.

Creating an Equipment Hierarchy

When you create instances and folders in the **Application Explorer**, you create a structure, which is defined by:

- The identifier and location of the application folders.
- The identifier and location of the instances.

By default, this structure is used in the Supervision project to create an equipment hierarchy, page 437, which allows you to view alarms in the Supervision page in a structure identical to the folder structure of the application.

Therefore, changes to the structure, page 213 of the application are reflected on the facets, page 379 of the impacted instances once these facets are assigned to a Supervision project and generated.

Viewing the Application Structure in Control Projects

You can view the folder structure of the application also in relation to Control Participant project sections, page 343 once you have instantiated templates and assigned facets to the Control project.

The functionality is available also when you refine a Control project, page 448 and make changes online, page 875.

Application Explorer

Overview

The work area, page 107 of the **Application Explorer** features:

- The **Templates Browser** pane, which lets you search installed Libraries and find the application templates that you need to create the instances of your application.
- The **Application Browser** pane, which lets you view, search, sort, and filter folders and instances of your application. You can also perform other tasks on folders and instances by using this pane.
- The **Asset Workspace** pane, which lets you create asset workspaces. These are structural elements that let you group instances in order to create links between them.
- In the form of tabs:
 - The properties of folders.
 - The **Instance Editor**, which lets you view and edit properties of instances.
 - The **Asset Workspace Editor**, which lets you view, create, and manage links between instances.

Opening the Application Explorer

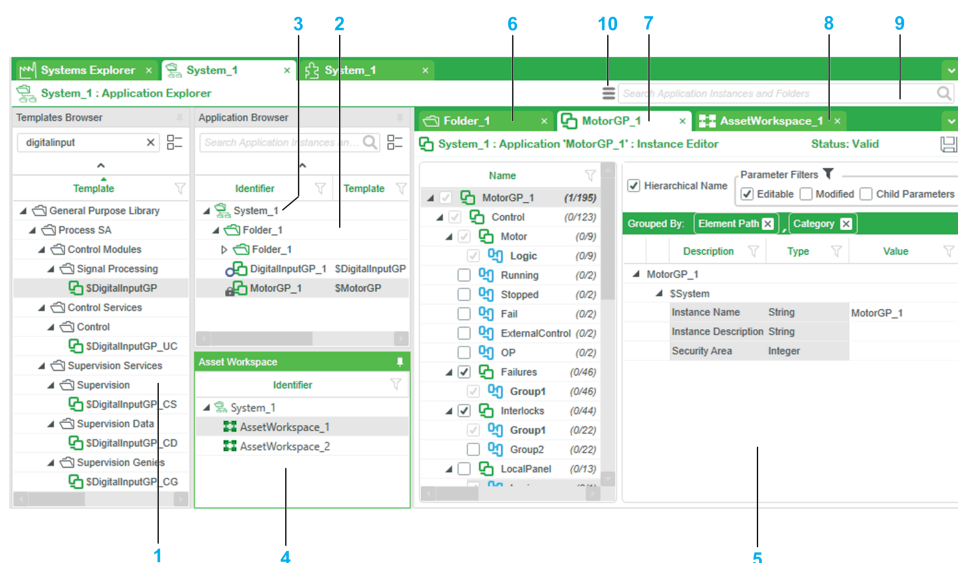
To open the **Application Explorer**, double-click the system in the **Systems Explorer**.

NOTE: You can also open the **Application Explorer** in the following ways:

- Drag the system from the **Systems Explorer** tree view to the tabs bar, page 103 or to a docking tab, page 110.
- Right-click the system in the **Systems Explorer** and select **Open Application**.
- Select the system and click the **Open Application Explorer** button in the toolbar, page 104.

Application Explorer Window

The following figure shows an example of the **Application Explorer** window.



Item	Description
1	Templates Browser pane, page 163
2	Application Browser pane, page 166
3	System root folder
4	Asset Workspace pane, page 193
5	Work area, page 107 of the Application Explorer . It is empty by default.
6	Properties of a folder
7	Instance Editor tab
8	Asset Workspace Editor tab
9	Search field, page 161 for folders and instances
10	Button to open the folder Settings pane, page 176

Searching Instances and Folders by Using the Application Explorer Search Field

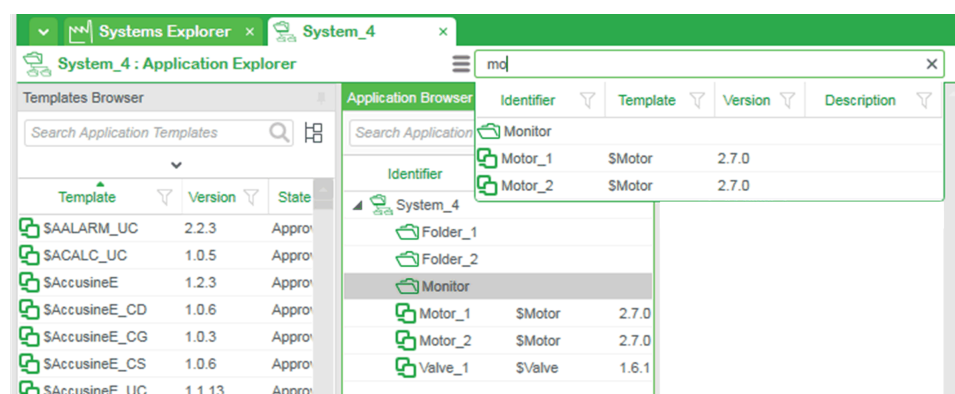
The search field of the **Application Explorer** lets you find instances and folders in the **Application Browser** by inputting all or part of their identifier. You do not need to use wildcards. The search is not case-sensitive. Results are displayed as you type in a pane with columns. Your last three matching search keywords are stored and appear in a keyword menu. The stored keyword entries are system-independent and persist even after the system server is restarted.

For example, `to` returns instances and folders that contain `to` in their identifier, such as instance *Motor_1*, and folder *Reactor*.

By default, results are displayed in ascending alphabetical order with folders shown first, instances next.

The results pane provides basic information on instances, page 177 and remains open until you close it by clicking the close button. A tooltip provides additional information.

The following figure shows an example of the search results that are displayed when you enter `mo` in the search field.



Right-click a search result in the results pane to open a context menu with the following commands.

Command	Description
Navigate	<ul style="list-style-type: none">For an instance: Expands the containing folder and highlights the instance.For a folder: Expands the folder structure and selects the folder. <p>You can also double-click the item.</p> <p>If the item is open in a window in a work area, the corresponding tab is also selected.</p> <p>NOTE: If the instance is hidden because a filter is applied in the Application Browser, the instance is not shown and a dialog box opens to inform you.</p>
Edit	<ul style="list-style-type: none">For an instance: Opens the Instance Editor tab, which lets you view and edit parameters of the instance, page 182.For folder: Opens the Properties tab, which lets you view, edit, and create parameters of the folder, page 172.

You can sort and filter, page 120 search results.

NOTE: Search results are not updated automatically if instances or folders were created, modified, or deleted in the meantime. You need to restart the search by clicking the search field and pressing **Enter**.

Browsing Application Templates

Overview

The **Templates Browser** pane of the **Application Explorer** lets you find the application templates of the Foundation Library and other Libraries that are installed in the database of the system server. These templates let you model the application of the system.

The **Templates Browser** pane also displays templates that you have created and/or imported.

To be displayed, a template needs to meet the following criteria:

- Template type: **Application**
- **Usability State** (see EcoStruxure Process Expert, Global Templates, Reference Manual) of either:
 - **Approved**
 - **Deprecated**

By default, quick filters are applied to display only templates that are:

- Composite templates, page 42.
- The latest version of the template.
- In the approved state.

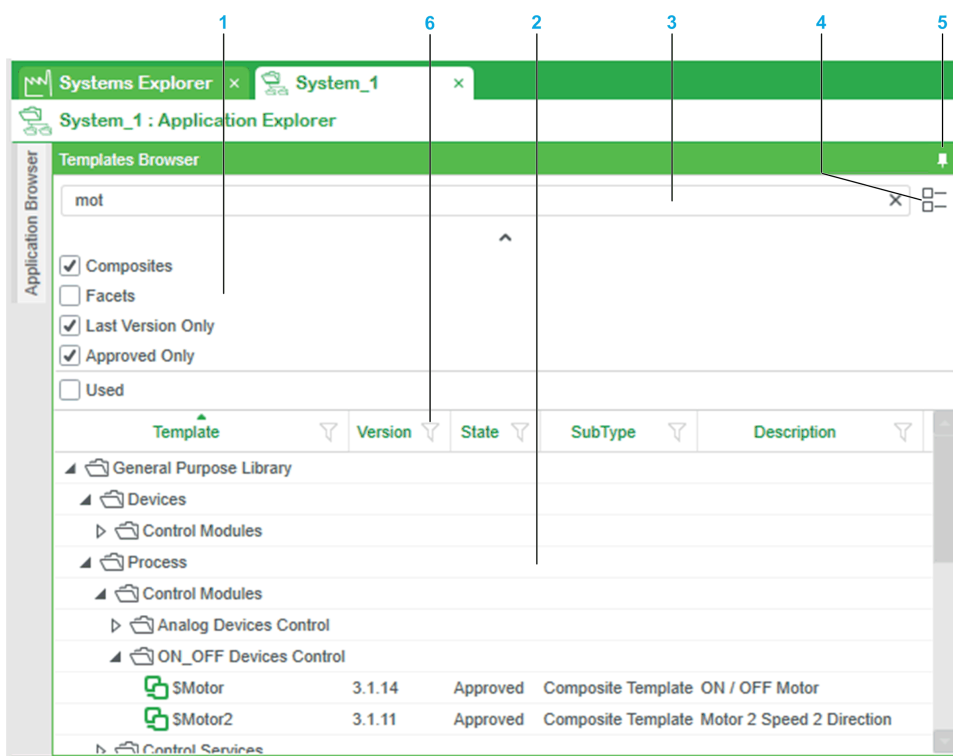
You can display additional templates or refine your search by:

- Using the search field.
- Modifying quick filters.
- Using column filters.

NOTE:

- For details on available Schneider Electric application templates, refer to the EcoStruxure Process Expert help, page 104.
- For information on the naming conventions followed by Schneider Electric templates, refer to the topic describing *Global Templates Libraries*, page 993.

Templates Browser Pane



Item	Description
1	Quick filters, page 165 (the check mark indicates that the filter is applied).
2	Search results. By default, in tree view, folders are shown first followed by instances. In grid view, instances are shown in a flat list. In both view modes, items are listed in ascending alphanumeric order.
3	Search field.
4	Button to toggle between grid and tree view for the display of search results.
5	Pin button, page 113. Lets you minimize the Templates Browser pane to the left edge of the screen. Move the pointer over the Templates Browser tab to display the pane. Click the pin button again to restore the pane.
6	Filters and sorting, page 120.

Searching Templates by Using the Search Field

The search field of the **Templates Browser** pane lets you search templates by inputting all or part of the template identifier. You do not need to use wildcards or type the \$ prefix. The search is not case-sensitive. Your last three matching search keywords are stored and appear in a keyword menu. The stored keyword entries are system-independent and persist even after the system server is restarted.

The search field is a filter for the active view mode as soon as you enter a string. Results are displayed as you type. To show the entire content, clear the field by clicking the close button or deleting your entry.

For example, `val` returns templates that contain `val` in their identifier, such as `$HandValve_UC`, `$MValve`, and `$ControlValve`.

The table describes the scope of the search and how results are displayed depending on the view mode that is selected.

View mode	What you can find	What is displayed
Tree view	<ul style="list-style-type: none">• Templates• Folders	Templates and folders matching the search criteria with their entire parent folder structure.
Grid view	Templates	Templates matching the search criteria in a flat list.

NOTE: Click the toggle button to switch between display modes. Search keywords and search results are retained.

Searching Templates by Using Quick Filters

The table describes the filters that are available in the **Templates Browser** pane to refine the template search.

Filter	Description	Selected by default
Composites	Displays only templates that are composites, which also include control module templates.	Yes
Facets	Displays only templates that are facets.	No
Last Version Only	Displays only the latest version of templates.	Yes
Approved Only	Displays only templates that have the approved (see EcoStruxure Process Expert, Global Templates, Reference Manual) usability state.	Yes
Used	Displays only the templates that are already used by instances of the application. Using this filter disables the other filters.	No

NOTE:

- When you select multiple filters, only templates that match the selected filters are displayed.
- When you select a filter, the results refresh automatically. When the filtered data is modified, the filters are not applied to the modifications.
- Filters are reset when you close the **Application Explorer**.

Additional Filters and Sorting

You can apply additional filters and sort the results, page 120 to refine your search.

Using the Application Browser Pane

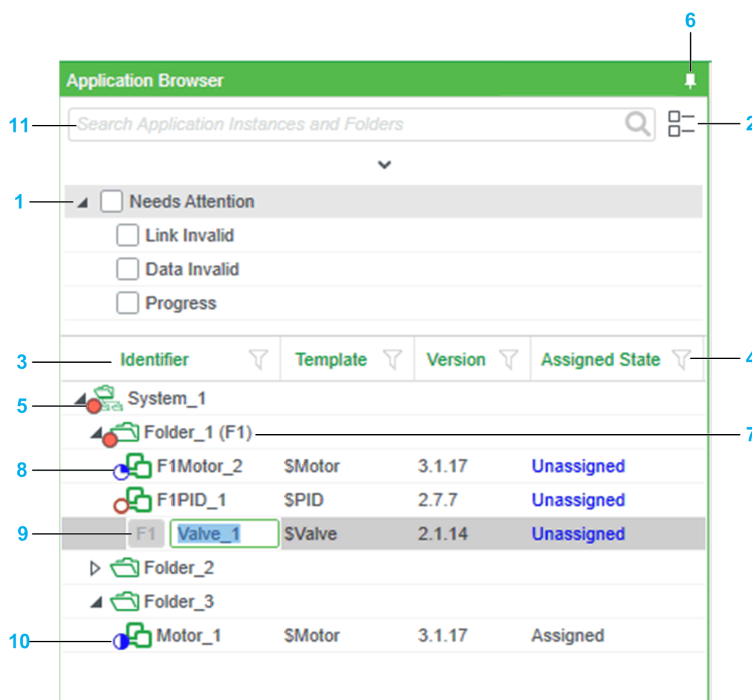
Overview

The **Application Browser** pane lets you:

- Create, delete, move, copy-paste, and rename folders and instances.
- Search, sort, and filter folders and instances
- Toggle between tree and grid view mode, the latter showing only instances.
- Customize the display and the order of columns.
- View and edit the properties of folders and instances.
- Add instances to an asset workspace that is open in the **Asset Workspace Editor** to create interface links.
- View assignments.
- Export and import application objects.

Application Browser Pane

The following figure shows an example of the **Application Browser** pane.



Item	Description
1	Status filters, page 167 (a check mark indicates that the filter is applied).
2	Button to toggle between tree and grid view mode.
3	You can customize, page 168 which columns appear and in which position.
4	Filters and sorting, page 120.
5	System root folder. By default, the system root folder is expanded to display its immediate children. Folders are shown first in the order you have arranged them, followed by instances in ascending alphanumeric order.
6	Pin button, page 113. Lets you hide the pane and restore it when it is docked.
7	If the folder has an alias, page 172, it is shown in brackets after the folder identifier.
8	For instances for which the hierarchical naming function is enabled, the alias is shown in front of the instance name, page 177 (if applicable).

Item	Description
9	When you edit the identifier of an instance for which the hierarchical naming function, page 189 is enabled, the alias appears to the left of the identifier in read-only mode (if applicable).
10	Indicator icons, page 195.
11	Search field, page 168.

Using Instance Status Filters

The table describes the impact on the search result when you select instance status filters in the **Application Browser** pane.

Filter	Description
Needs Attention	Shows only instances that satisfy the three subfilter criteria.
Link Invalid	Shows only instances whose Link status, page 204 is invalid.
Data Invalid	Shows only instances whose Data status, page 177 is invalid.
Progress	Shows only instance for which a progress icon, page 196 appears.

NOTE:

- When you select multiple filters, only templates that match the selected filters are displayed. You can combine column filters, page 120 and use the search field to refine your search further.
- When you select a filter, the results refresh automatically. The other way around, when the filtered data is modified, the filters are not applied to the modifications.
- Filters are reset when you close the **Application Explorer**.

Toggling Between Tree and Grid View

Tree view is the default view mode. You can switch to grid view by clicking the toggle button. The item selection in either view mode is retained when you switch and revert to this mode.

In grid view, when no sorting and filter is applied, all the instances of the application are shown in a flat list sorted in ascending alphanumeric order.

When switching between view modes, the search criteria are carried over, but the filtering and sorting criteria are not.

NOTE: In grid view mode:

- Commands of context menus related to folders are not available.
- You cannot create instances by dragging a template to the **Application Browser** pane.

Navigating Inside the Browser Pane

The table describes the various methods that you can use to navigate inside the browser pane.

Key or action	Result
Up/down arrow key	Moves the selection to the previous/next item.
Left/right arrow key ⁽¹⁾	Expands/collapses the next level of the selected node.
Enter ⁽¹⁾	
Double-click ⁽¹⁾	On a node, expands/collapses the next level of the node.
Plus (+)/minus (-) key	Expands/collapses the entire hierarchy from/up to the node.

(1) The next time you expand the node by using one of these methods, the display reverts to the last expanded view.

Customizing Column Display and Order

You can choose which columns you want to see by right-clicking a column header and selecting **Customize**. The **Identifier** and **Template** columns are always shown.

The **Path** column is available only in grid view.

When you hide a column for which a filter is applied, the filter is cleared.

When the **Customize** dialog box is closed, you can change the order of columns by dragging a column to its new position (except for **Identifier**).

You can configure the column display and order independently in tree and grid view.

For a description of each column, refer to the topic describing the creation of instances, page 177.

NOTE: The order of columns is reset each time you open the **Customize** dialog box and click **Apply**.

Searching Instances and Folders

The search field of the **Application Browser** pane lets you search folders and instances by inputting all or part of their identifier. You do not need to use wildcards. The search is not case-sensitive. Your last three matching search keywords are stored and appear in a keyword menu. The stored keyword entries are system-independent and persist even after the system server is restarted.

For example, inputting `t○` returns instances that contain `t○` in their identifier, such as folder *Torrefier* and instance *Motor_1*.

The search field is a filter for the active view mode as soon as you input a string. Results are displayed as you type. To show the entire content, clear the field by clicking the close button or deleting your entry.

The table describes the scope of the search and how results are displayed depending on the view mode that is selected.

View mode	What you can find	What is displayed
Tree view	<ul style="list-style-type: none">• Instances• Folders	Instances and folders matching the search criteria with their entire parent folder structure. NOTE: When searching for folders, if several folders within the same hierarchical structure match the search criteria, only the highest level parent folder is shown in a collapsed state. You can see whether any child folders match the search criteria by entering the search criteria in the search field of the Application Explorer . A tooltip indicates the path to folders in the search results.
Grid view	Instances	Templates matching the search criteria in a flat list

Search keywords and search results are retained when you switch between view modes.

By default, results are displayed with folders shown in the order you have arranged them, followed by instances in ascending alphanumeric order.

If [sorting and/or filters](#), [page 120](#) are applied, they are also applied to the search results.

NOTE: Search results are not updated automatically if instances or folders are created, modified, or deleted. For example, if you have entered `va` in the search field and create a folder, which has `Folder_n` as default identifier, it will be shown.

Creating Application Folders

Overview

You can create application folders to structure your application by using the **Application Browser**.

You can structure the application according to the ISA-S88.01-1995 standard for batch control. You can create and rename the application folders according to the physical assets that you want to model.

Folders that you create may inherit properties of their parent folder, such as the **Area**.

You can also do the following:

- Create folders by copy and paste or import, page 227.
- Move folders, page 172 with or without changing the folder hierarchy.
- Assign icons to application folders, page 176 to define a hierarchy of equipment.

Creating Application Folders

To create an application folder, right-click the system root folder and select **Create Folder**.

Application Folder Actions

Right-click the root folder, an application folder, or multiple folders to open a context menu with the following commands.

Command	Description
Create Folder	Creates an application folder or subfolder. The command is not available when more than one folder is selected.
Open Topology	Opens the Topology Explorer , page 270, which allows you to start the configuration stage or access the topology of the system. The command is available only from the root folder.
Open Project	Opens the Projects Explorer , which allows you to start the project definition stage, page 262 or access the Control and Supervision projects of an existing system. The command is available only from the root folder.
Update Template , page 220	Opens the Update Instance Template window. It lets you update the template that is used by instances inside the folder and subfolders with the latest version of that template that is available in the Global Templates Library.
Replace Template , page 220	Opens the Replace Instance Template window. It lets you select another template to be used by instances inside the folder and subfolders. The template must be available in the Global Templates Library.
Copy , page 215	Copies folders, their subfolders, and the instances they contain. The command is not available from the root folder.
Paste , page 215	Pastes the copied items to the selected locations in the system or in another system of the system server. The command is available only if the copied content is valid for the selected target locations.
Export , page 227	Opens an Export dialog box, which allows you to save the parameters and configuration data of application objects to a file in CSV or XML format.
Import , page 254	Opens an Open dialog box, which allows you to select an export file in CSV or XML format, display its compatible contents, and select instances to be imported into the application. The command is not available if more than one folder is selected.
Delete	Deletes the selected folders and their contents. The command is not available from the root folder. NOTE: After confirming the command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. Already deleted instances and/or folders are restored. Only the user who selected the command is allowed to cancel it. If you are deleting only a few items, the task may complete before you are able to click the icon.
Rename	Lets you enter a new folder identifier. The command is not available from the root folder and if more than one folder is selected. NOTE: Renaming a folder may affect the equipment hierarchy, page 159. If the folder has an alias, it is not shown while the identifier is in edit mode.
Properties	Opens the Properties , page 172 tab, which allows you to edit the properties of the application folder. The command is not available from the root folder and if more than one folder is selected.

Moving Folders

You can move one or more folders by dragging them to a new position. A tooltip indicates the result of the action, which can have an impact on instances if the folder hierarchy is changed.

The table describes the tooltips that can appear and the corresponding result.

Tooltip	Result
Move X before Y	The folder and its content is repositioned before or after the other folder.
Move X before Y	The folder hierarchy is not changed.
X	<p>The folder that you are moving and its content becomes a child of the other folder.</p> <p>The equipment hierarchy, page 437 may be impacted.</p> <p>The instance identifier, page 189 and assignment status, page 425 of generated facets may be impacted.</p>
<p>Where:</p> <p>X Is the identifier of the moved folder.</p> <p>Y Is the identifier of another folder of the application.</p>	

NOTE: If you move several folders, you cannot reposition them and maintain the same hierarchy.

Application Folder Properties

To open the properties of the application folder, right-click the folder and select **Properties**. A tab opens inside the work area, page 107 of the **Application Explorer**.

You can also open the properties by dragging the folder to the work area, the tabs bar of a work area, or to a docking tab, page 111 within the **Application Explorer**.

You can edit the following items.

Tab	Item		Description	
General	Identifier		<p>Designation of the folder.</p> <p>The identifier must be unique for each subfolder of the same parent folder and satisfy naming rules, page 31.</p> <p>Default value: <i>Folder_n</i> where <i>n</i> is an incremental number starting at 1 for each folder level.</p> <p>NOTE: Changing the identifier of an application folder may affect the equipment hierarchy, page 159.</p>	
	Alias		<p>Optional.</p> <p>If you assign an abbreviated name to the folder, it must comply with applicable naming rules, page 31.</p> <p>The alias is shown in brackets next to the folder identifier.</p> <p>The alias is used for the hierarchical naming function, page 189.</p> <p>You cannot create or modify the alias value if it causes an instance identifier to become non-unique.</p> <p>Default value: Blank</p>	
	Description		<p>Optional.</p> <p>You can enter a description of the folder with free-form text. The description you enter appears in a tooltip when you hover with the pointer over the folder.</p> <p>Default value: Blank</p>	
	Area		<p>Optional.</p> <p>You can assign a folder to an area.</p> <p>Range: 1...255 (integer)</p> <p>Default value: Blank</p> <p>EcoStruxure Process Expert attempts to propagate the area value to subfolders and instances therein if no area value was manually assigned to them.</p> <p>For details, refer to the topic describing area value propagation, page 173.</p> <p>NOTE: You can also enter a string instead of an integer value. In this case, you must refine the Supervision project after generating it and use the same string as label to name the area; otherwise, the Supervision project does not build successfully. Refer to the help of the Supervision Participant for details.</p> <p>NOTE: The area value is also propagated to Supervision elements of instances such as genies.</p>	
	Manage Parameters	Parameter_1	Identifier	You can use as many folder parameter sets as you require.
...		Description	Default value: No parameter	
Parameter_n		Value		
NOTE: Click Add Parameter to add a parameter.				

Area Parameter Value Propagation

EcoStruxure Process Expert automatically propagates the value (including blank (NULL value)) of the **Area** parameter of a folder to the following:

- Existing subfolders and instances that the folder and its subfolders contain unless you manually assigned them an area value.
- Subfolders and instances that you create afterwards.
- Subfolders and instances that you add afterwards either by copying or moving them, unless an area value was automatically or manually assigned to them.

If you change an existing area value of a folder, only the area value that was propagated automatically to the next level of subfolders and their instances is updated. The change is propagated through the folder structure, level after level, until a manual folder area entry is detected.

- If EcoStruxure Process Expert is unable to propagate the value of the **Area** parameter to an item because the item is locked, it proceeds as follows:
- It displays a notification to inform you which item is locked.
 - It undoes the area value propagation.
 - It reverts the **Area** parameter value that you entered in the folder properties to its previous value.

NOTICE

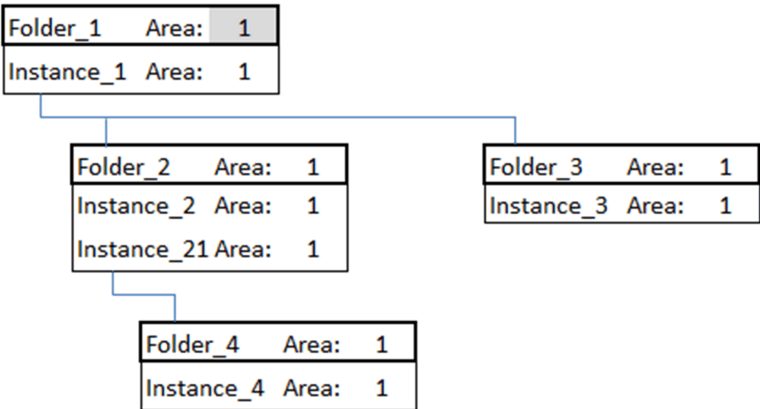
NO CHANGE PROPAGATION
Ensure that the **Area** parameter value of a folder is propagated to the entire folder structure by referring to the notification panel. Otherwise, manually enter the **Area** parameter value in the folders where it was not propagated.
Failure to follow these instructions can result in equipment damage.

NOTE: If the **Area** parameter value of a folder is not propagated to the entire folder structure, alarms may be signaled in the wrong area in runtime.

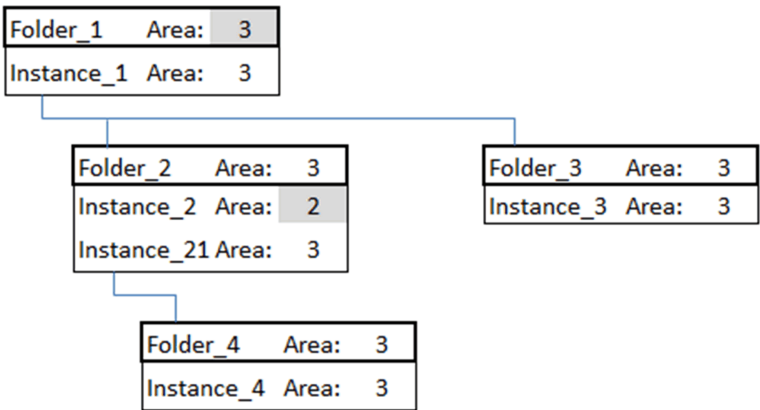
Examples

The following three examples illustrate the area value propagation to existing subfolders and instances in the **Application Explorer**. The changes described in these examples are performed in steps inside the same folder structure.

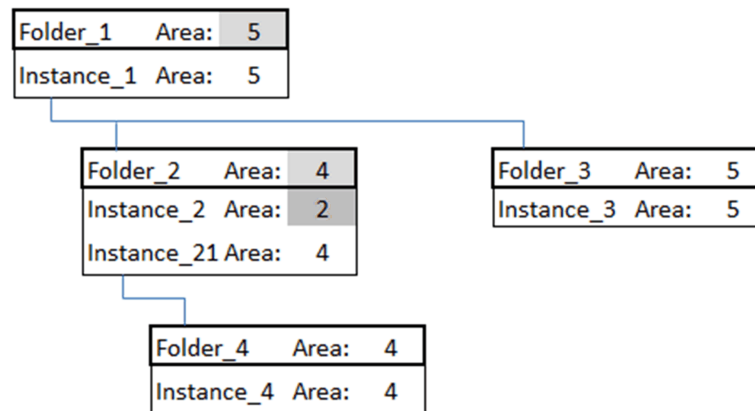
Starting from a folder structure and instances without area value, changing **Area** of *Folder_1* to 1 propagates it to *Instance_1*, subfolders, and their instances.



Then, if you change **Area** of *Instance_2* to 2 and **Area** of *Folder_1* to 3, value 3 is propagated to *Instance_1*, subfolders, and their instances except for *Instance_2* because the area was manually entered.



Last, if you change **Area** of *Folder_2* to 4, value 4 is propagated to *Instance_21*, the *Folder_4* subfolder and its instance. *Instance_2* remains unchanged because the area was manually entered. If you then change **Area** of *Folder_1* to 5, value 5 is propagated only to *Instance_1*, subfolder *Folder_3*, and its instance because the area of other subfolders and instances have either been manually entered (for example, *Folder_2*) or are manual entries that were propagated (for example, *Folder_4*).



Assigning Icons to Application Folders

Overview

By using the **Settings** pane, you can assign icons to each level of application folders. The icons appear in the tree view of the **Application Browser** pane, page 160.







You can use this feature, for example, to represent graphically the physical hierarchy of equipment used in a batch process according to the ISA-S88.01-1995 standard.

The settings apply to the system for which you configure them, to any user who opens the system in an engineering client, and persist even if you restart the system server.

NOTE: You cannot assign an icon to the system root folder.

Available Icons

You can choose from the following selection of icons.

Icon	Description
	Default folder icon
	Site
	Area
	Process cell
	Unit
	Equipment module

NOTE: Control modules are represented by the instances of the application.

Assigning Icons to Folder Levels

To assign an icon to a folder level, proceed as follows.

Step	Action
1	Click the Explorer Layout Settings button, page 104. Result: The Settings pane opens.
2	In the Appearance section, select the folder level to which you want to assign an icon.
3	In the Style menu, click the icon that you want to use. Result: The icon is assigned to each existing folder of the selected level and to each new folder that you create in the selected level.
4	Repeat the procedure for other folder levels.

Creating Instances

Overview

Instances that you create in the [folder structure](#), [page 170](#) of the application are created with the default configuration of their template. Basic information on each instance is available at the folder level.

Once you have created an instance by using the **Application Browser**, you can perform various actions by using the pointer, accessing the context menu of the instance, or using keyboard shortcuts.

Order of Instantiation

The order in which you create instances inside an application folder may affect the order of the referenced Control facets in a section during [assignment](#), [page 375](#).

Instantiating Application Templates

To instantiate a template in the root folder or in any application folder, proceed as follows.

Step	Action
1	Verify that the folder in which you want to create the instance is displayed in the Application Browser .
2	In the Templates Browser , select a template.
3	Drag the template to the folder. Result: The folder opens or expands and an instance of the template is created in last position with the default identifier in edit mode. NOTE: If the folder has an alias, it is shown in a separate cell on the left of the identifier.
4	To change the identifier, type it; otherwise, press Enter to accept the default name. NOTE: To keep the default name and create a new instance, you do not need to press Enter ; you can drag another template while the identifier of the instance is still in edit mode.

NOTE: Refer to *Managing Application Folders and Instances*, [page 210](#) for information on features that are available when working with many application folders and instances.

Instance Description

The following information of an instance is displayed in the **Application Browser** when the corresponding column is shown, [page 168](#).

Header	Description
Icon	Displays the icon corresponding to the type of template that the instance uses.
Identifier	<p>For instances, displays the instance identifier (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>), which must be unique for each instance of the same template. The identifier is calculated by using the \$Name parameter of the instance and the applicable naming conventions.</p> <p>For folders, displays the folder identifier, which you can edit in the Properties tab or by using the Rename command.</p> <p>NOTE:</p> <ul style="list-style-type: none"> You can edit the \$Name parameter of an instance in the Instance Editor, page 182 or by using the Rename command. If the resulting identifier is not unique, the \$Name text box is outlined in red. In this case, clicking outside of the field reverts the name to its previous value. The hierarchical naming function, page 189 can modify the identifier of an instance.

Header	Description
	<ul style="list-style-type: none"> Two instances using two different templates cannot have the same identifier if both instances have an element in their composition in common (for example, the <i>Logic</i> element, which is created by the <code>\$DEVCTL_UL</code> facet). In this case, the identifier of the element that both instances have in common would be the same; this is not allowed.
Template	<p>Indicates the identifier of the template that the instance uses.</p> <p>The field is not used for folders.</p>
Version	<p>Indicates the version of the template that the instance uses.</p> <p>The field is not used for folders.</p>
Data	<p>Indicates the status of the instance. Possible values:</p> <ul style="list-style-type: none"> Valid Invalid <p>The value is updated when you save changes in the Instance Editor.</p> <p>The field is not used for folders.</p> <p>NOTE: You can obtain information about the invalid status, page 186 of an instance in the Validity Diagnosis window:</p> <ul style="list-style-type: none"> By clicking Invalid. When viewing its properties, page 182.
Link	<p>Indicates the status of the application interface link of the instance. Possible values:</p> <ul style="list-style-type: none"> Valid Invalid <p>The value is updated when you make a link by using the Asset Workspace Editor, page 200 or save changes in the Instance Editor that impact the link requirements of the instance.</p> <p>The field is not used for folders.</p> <p>NOTE: You can obtain information about the invalid status, page 204 of an instance in the Validity Diagnosis window:</p> <ul style="list-style-type: none"> By clicking Invalid. In the Asset Workspace Editor.
Assigned State	<p>Indicates the assignment status, page 379 of the facets of the instance. Values:</p> <p>Assigned: At least 1 facet of the instance is assigned to any project of the system.</p> <p>Unassigned: No facet of the instance is assigned to any project of the system.</p> <p>The field is not used for folders.</p>
Description	<p>Displays the content of the \$Description parameter the folder or the instance. You can edit the field in the Properties or the Instance Editor, page 182 tab respectively.</p>
Area	<p>Displays the value of the Area parameter for the folder or the instance. You can edit the field in the Properties tab or the Instance Editor, page 182 tab respectively.</p> <p>NOTE: If the value was assigned automatically, page 173, it appears between angle brackets. For example, <5>.</p>
Path	<p>Indicates the path to the instance in the application starting from the system root folder.</p> <p>NOTE: The column is shown only in grid view mode.</p>

Instance Actions

Right-click an instance to open the instance context menu with the following commands.

Command	Description
Properties⁽¹⁾	Opens the Instance Editor , page 182 in the work area, which allows you to view and configure parameters, and select services of the instance.
Navigate to Workspace⁽¹⁾	<p>Opens the Show Workspace window, which lists the asset workspaces that the instance belongs to.</p> <p>Double-clicking an asset workspace opens it in the Asset Workspace Editor and highlights the instance. This lets you create and manage links between instances.</p>
Edit Links⁽¹⁾	Opens the Links Editor , page 1127, which allows you to view the interface links that exist for the instance.
View Assignments , page 392 ⁽¹⁾	<p>Opens the View Assignments window, which lets you verify if the facets that are referenced by the instance are assigned to Participant projects:</p> <ul style="list-style-type: none"> If a facet is not assigned, you can select an existing project and open the Assignment Editor to assign it. If a facet is assigned, you can access the container to which the facet is assigned. <p>For Supervision genie facets, you access the Pages node in either case.</p> <p>NOTE: The command is not available if the Data or Link status of the instance is Invalid.</p>
Update Template , page 220	Opens the Update Instance Template window. It lets you update the template that is used by the instance with the latest version of that template, which is available in the Global Templates Library on the system server.
Replace Template , page 220	Opens the Replace Instance Template window. It lets you replace the template that the instance is using by another template, which is available in the Global Templates Library on the system server.
Inspect Instance⁽¹⁾ , page 211	Opens the Inspect Instance window, which provides information on the entire instance configuration, including interface links.
Copy , page 214	Copies the selected instances for pasting within the application of the system or any other system on the server.
Export , page 228	Opens an Export dialog box, which allows you to save the instance data to a file in CSV or XML format.
Rename⁽¹⁾	<p>Sets the identifier of a single instance to edit mode. Type the new identifier and press Enter.</p> <p>NOTE: The instance identifier is split in two and only the instance name (\$Name) part is editable if the following conditions are satisfied:</p> <ul style="list-style-type: none"> You have defined an alias for the folder containing the instance or any of its parent folders. The hierarchical naming function, page 189 for the instance is enabled. <p>The concatenated, non-editable aliases are shown on the left of the instance identifier.</p>
Delete	<p>Deletes the instance from the application.</p> <p>You can select multiple instances within the same folder.</p> <p>Not available while you edit the instance.</p> <p>If you have assigned facets of the instance to projects, the assignment state, page 379 of these facets changes to Deleted.</p> <p>NOTE: After confirming the command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. Already deleted instances are restored.</p> <p>Only the user who has selected the command is allowed to abort it.</p> <p>If you are deleting only a few instances, the task may complete before you are able to click the icon.</p>
(1) The command is not available when multiple instances are selected.	

Counting Instances

Overview

EcoStruxure Process Expert counts the instances that exist in the application of any system on the system server and compares their number to the maximum number of instances that the application size license that is activated allows.

At any time during the system engineering life cycle, you can view the instance count for each system, the total count, and the maximum number of instances that is allowed.

EcoStruxure Process Expert does not allow actions that cause the maximum number of instances of the application to be exceeded.

Viewing Instance Count Information

Instance count and software license information is displayed in the **About EcoStruxure Process Expert** dialog box, page 106.

The table describes the information that is displayed in the **License Details** section of the **General** tab.

Label	Description
Maximum Instances Allowed	Total number of instances that you can create with the current license.
Current Instance Count	Total number of instances in the systems that exist on the system server.

The table describes the information that is displayed in the **Systems Details** tab.

Header	Description
System Name	Identifiers of the systems that exist on the system server.
Instance Count	Total number of instances of the system.

Actions Affecting Instance Count

The following actions that you can perform in the **Application Explorer** increase and/or reduce the instance count in the application:

- Instantiate
- Paste
- Import
- Delete

The following actions that you can perform in the **Systems Explorer** increase and/or reduce the instance count in the application:

- Restore
- Delete

Exceeding the Allowed Number of Instances

If one of your actions causes the maximum number of instances that is allowed by the activated license to be exceeded, the entire action is canceled. It also displays a message to advise you that you have reached the limit of instances allowed by the license.

In this case you have the choice to:

- Delete instances (you can export the instances, page 228 before deleting them to reuse them later.)
- Delete systems (you can back up the system, page 145 before deleting it to reuse it later.)
- Upgrade the application size license (see *EcoStruxure Process Expert, Licensing Guide*).

Accessing Instance Properties

Overview

The **Instance Editor** tab opens in the inside the work area, page 107 of the **Application Explorer** and provides access to:

- Control, Supervision, and documentation services, page 42, which are provided by the template that the instance is using.
- Parameter values of the elements, page 39 of the instance.
- The number of parameters that have been modified.
- The status of the instance.

NOTE: More information on the instance configuration is available in the **Inspect Instance** window, page 211 in read-only mode.

Read-Only Mode

You can open the **Instance Editor** for an instance in read-only mode if either of the following is true:

- The instance is being edited, for example, from another client connected to the same system server.
- The instance is displayed in the **Links Editor**, either as active or inactive., page 1127

The read-only mode allows you to view instance parameters but not to modify them.

Opening the Instance Editor

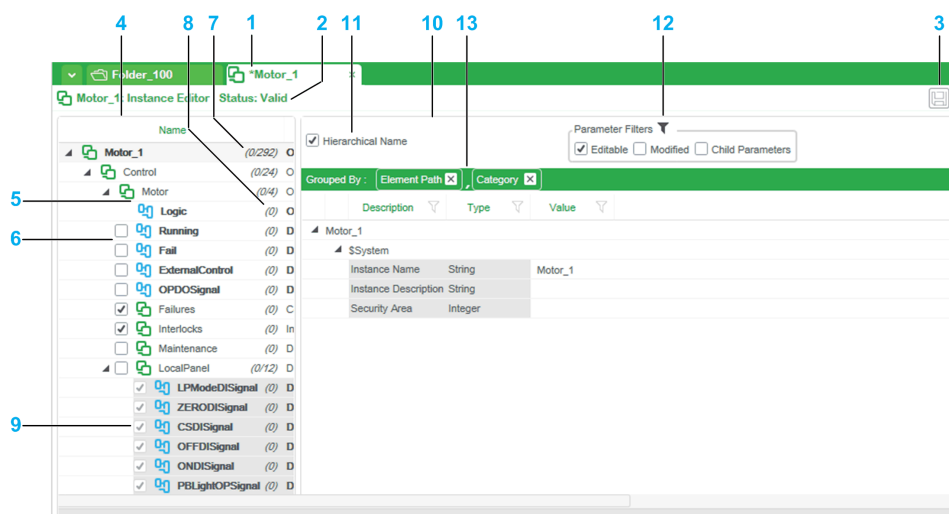
To open the **Instance Editor**, double-click the instance that you want to configure in the **Application Explorer**.

You can also open the **Instance Editor** in the following ways:

- Right-click the instance and select **Properties**.
- Drag the instance to either area within the **Application Explorer**:
 - The work area, page 107.
 - The tabs bar of a work area.
 - A docking tab, page 111.

Instance Editor

The following figure shows an example of instance *Motor_1*, which uses template *\$Motor* open in the **Instance Editor**.



Item	Description
1	Instance identifier. It begins with an asterisk (*) if you made a change without saving it.
2	Instance validity status indicator. The status is set to Invalid and details are available by clicking the information button if either the Data or Link status, page 177 is invalid.
3	Save button. Click to save parameter values and propagate changes, page 379.]
4	Instance pane, which allows you to view the services that the instance provides and to enable/disable optional ones. By default, nodes are shown expanded.
5	Element displayed in bold indicating that the element contains editable parameters.
6	Check box for element selection.
7	Counter indicating the number of modified parameters for a node out of the total number of parameters that exist for that node.
8	Counter indicating the number of modified parameters for an element.
9	Elements of optional nodes of the instance that are not selected and that have configurable parameters are displayed on a light gray background. This allows you to identify which are the optional services of the instance that provide additional customization capabilities. Selecting the node removes the gray background.
10	Parameters pane, which allows you to configure the parameters of the elements of the instance that you have selected.
11	Optional Hierarchical Name function, page 189 selection box. The function is enabled by default.
12	Parameter Filters section, page 185.
13	Grouped by sorting function for parameters.

The following table describes the columns of the instance pane.

Column	Description	
Name	Shows the composition of the instance by displaying the identifier of the instance and: <ul style="list-style-type: none">• Its core and optional composite elements (nodes)• Core and optional child elements (facets) Elements (facets) provide the services of the application template, which depend on the template that the instance uses. For details on the services that are provided by Schneider Electric application templates, refer to the EcoStruxure Process Expert <i>help</i> , page 104. An item in bold indicates that you can configure parameters of the element. Move the pointer over an element to see a description of the element in a tooltip.	
	Icon	Indicates the type of template that is used by the instance and its elements.
	Check box	Indicates that the element is optional and not mandatory for the control module to function. Typically, optional elements correspond to additional functionality that is built into the template and that you can select to customize the instance according to your needs. An optional element may contain optional subelements, which you can select individually. Selecting an element enables it and creates the corresponding facets. If applicable, it allows you to configure parameters of the service that the element provides. Selecting an optional Control element automatically selects the corresponding Supervision element if it is available. For example, when you select the <i>Interlocks</i> element of the <i>\$Motor</i> template, the <i>InterlockTags</i> Supervision element is selected. Conversely, clearing a selected Control element automatically clears the corresponding Supervision element. The following Supervision elements are selected by default: <ul style="list-style-type: none">• Elements corresponding to core Control elements, which are not optional.• Elements corresponding to optional Control elements, which are selected by default.
	(x / y)	Node counter (7). For the instance and for its composite elements, the first number indicates the number of parameters that you have changed in the node (x) compared to the default value and the second, the total number of parameters (y) that you can change for this node. Changes to the Instance Name are not counted. Counters are updated as soon as you confirm your entry by pressing Enter . For boolean types, when you select or clear them. NOTE: The counter turns bold to indicate that you have changed a parameter of the element itself. When you change a parameter of any of its child elements, the counter font style remains regular.
	(z)	Element counter (8). For child elements, indicates the number of parameters that you have changed compared to the default values. The counter appears even if the element has no editable parameters. Counters are updated as soon as you confirm your entry by pressing Enter .
Description	Description of the instance or element displayed in the Name column.	

Column	Description
Template	For instances and composite elements, indicates the identifier of the template that the instance or composite element is using. For facet elements, indicates the identifier of the facet template that encapsulates the constituent provided by the Participant to implement the service.
Version	Version of the template that the instance or element uses.

The table describes the columns of the parameters pane.

Column	Description
Name	Identifier of the parameter. The column is hidden by default. Right-click any column header and select Show Name to display the column. Right-click any column header and select Hide Name to hide the column.
Description	Description of the parameter.
Type	Indicates the data type of the parameter.
Value	Default value of the parameter or the value that you have entered. NOTE: For enumerated data type parameters, each enumerator displays both the name and the corresponding integer (for example, <i>Operator (2)</i>)
Resolved Value	Displays the value after formatting according to the data type. For example, when the check box of a boolean parameter is selected, the value that is displayed is <i>True</i> . The column is hidden by default. Right-click any column header and select Show Resolved Value to display the column. Right-click any column header and select Hide Resolved Value to hide the column. NOTE: Open the Inspect Instance window, page 211 to view the resolved value of additional template parameters, which is calculated by using the value property and any applicable logic defined in the template definition.

Navigating Inside the Instance Pane

The table describes the various methods that you can use to navigate inside the instance pane.

Key or action	Result
Up/down arrow key	Moves the selection to the previous/next item.
Left/right arrow key ⁽¹⁾	Expands/collapses the next level of the selected node.
Enter ⁽¹⁾	
Double-click ⁽¹⁾	On a node, expands/collapses the next level of the node.
Plus (+)/minus (-) key	Expands/collapses the entire hierarchy from/up to the node.

(1) The next time you expand the node by using one of these methods, the display reverts to the last expanded view.

Displaying Instance Parameters Using Filters

The table describes the filters that are available in the **Parameter Filters** section to customize the display of parameters of the selected instance or element.

Filter	Description
Editable	Displays only parameters of the selected instance or element that you can edit. A parameter may not be editable because it is optional and you have not selected it. The filter is applied by default.
Modified	Displays only parameters of the selected instance or element whose value you have modified compared to their default value.
Show Child Parameters	Displays parameters of the selected instance or element including those of child elements.
None selected	No filter is applied. Displays the parameters of the selected instance or element.

NOTE:

- When you select multiple filters, only parameters that match the selected filters are displayed.
- When you select or clear a filter, the results refresh automatically.

To apply default filters, click the filter icon, then click **Default**.

To clear any filters, click the filter icon, then click **Clear Filter**.

Grouping Instance Parameters

The **Group By** bar allows you to group parameters that are displayed in the parameters pane by the following criteria:

- **Element Path**: Path to the element that provides the parameters, as shown in the instance pane. The path is not available for instances based only on a facet template.
- **Category**: Category of the parameter.
- **Type**: Data type of the parameter.

By default, parameters are grouped as follows:

- For instances based on a composite template: By **Element Path**, then **Category**.
- For instances based on a facet template: By **Category**.

Click the close button of a criteria to remove it from the grouping. The corresponding column is displayed.

Drag a column header to the **Group By** bar to group parameters by the corresponding criteria. You can group by multiple criteria simultaneously.

When multiple criteria are selected, you can swap their order by dragging a criteria left or right.

When you have chosen to group by a criteria, the corresponding column is not displayed anymore.

Instance Data Status


When any of the following conditions apply, the **Data** status of the instance in the **Instance Editor** changes to **Invalid**:

- Entering a character that is not allowed, page 31 in the **\$Name** field.
- Entering a number of characters in the **\$Name** field causing the instance identifier to exceed the length limit that may be defined, page 31.
- Selecting an invalid combination of optional elements in the instance pane of the **Instance Editor**.

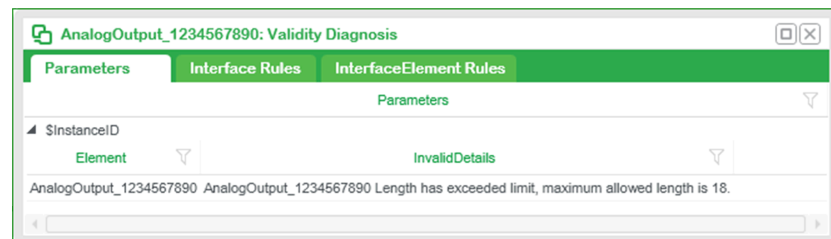
- Creating an instance that references an element that has the same identifier as an existing element in the application. For example, if you create two instances with the same identifier from two different templates, which have an element in common, such as *Interlocks*, which uses the *\$CONDSUM1_UC* facet template.

Instances with an **Invalid Data** status cannot be assigned to projects.

If facets of an instance are assigned to a Participant project, and later either the **Data** or **Link** status of the instance changes to **Invalid**, EcoStruxure Process Expert propagates the invalid status to facets of Participant projects.

While the **Data** status is displayed as **Invalid**, click the validity status information button  to display the **Validity Diagnosis** window, which provides information about the cause of the status.

The following figure shows an example of the **Parameter** tab of the **Validity Diagnosis** window that is displayed for an invalid identifier.



NOTE: For information on the other tabs that are related to the **Link** status of the instance, refer to the topic describing instance links, page 204.

NOTE: If the status of an instance is **Invalid** because of a duplicate element identifier, deleting or modifying the other instance that contains the same element does not change the status back to **Valid**. After you have solved the conflict, edit the invalid instance and save any change (for example, a modification of the description) to set its status to **Valid**.

Configuring Instances

Overview

The **Instance Editor** lets you do the following:

- Enable or disable optional Control, Supervision, and documentation services that are provided by the template that the instance is using.
- View and configure parameters of the instance.

For more information to help you select and configure instances of templates to meet the requirements of your application, refer to the EcoStruxure Process Expert help.

For information on how to link documents, files, and associate web links to instances for use with runtime navigation services, refer to the topic describing how to add user contents (see *EcoStruxure Process Expert, Runtime Navigation Services, User Guide*).

NOTE: If an instance is being edited by another user, It can only be opened with the **Instance Editor** in read-only mode.

Modifying Instance Parameters After Assignment

If you modify and save the properties of an instance whose facets are assigned to a Participant project, a message may be displayed to inform you that your modifications affect the assignment status of these facets.

The message does not take into consideration whether the assigned facets are generated or not.

Only the assignment status of generated facets can be changed to **Out Of Date** or **Deleted**, page 425.

If facets are assigned without being generated, when you save changes, the following applies:

- Modifications are propagated without affecting the assignment status. For example, if you change the instance name, the identifier of assigned facets is updated in the Participant project.
- Facets may be removed from the container to which they are assigned. For example, if you clear the element of an instance, the corresponding facet that was assigned is deleted.

Changing the Element Selection of Instances With Assigned Facets

If you change the element selection of an instance that has at least one facet already assigned and the change results in the creation of new facets, EcoStruxure Process Expert can assign candidate facets automatically to the same container.

Refer to the topic describing automatic facet assignment, page 381 for more information.

Validating Parameter Values

If you edit parameters of the instance, the **Instance Editor** displays and validates values according to the supported data types, page 28, helping you configure the values in accordance with the data type of the parameter.

You cannot save changes if EcoStruxure Process Expert detects an invalid parameter value.

Certain supported data types, page 28 use language-specific formatting according to the **Regional Options** that are set for the operating system of the computer. For example, if your regional settings define . (dot) as the decimal delimiter and you enter a decimal value by using comma instead of the dot, for example, 100,5, this value is processed as 1005.0.

NOTE: Except for changes to the **\$Name** parameter, the following applies when edit instance parameters:

- Changes appear in italic until you save them.
- Edited values that are different from the default value are shown in bold.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Verify which **Regional Options** are configured in the operating system of your computer and follow the applicable formatting rules to enter data in the **Instance Editor**.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

In addition, the following formatting rules apply in the **Instance Editor** when you edit the **\$Description** parameter and other element parameters of type String of an instance.

Rule	Example or comment
"" returns an empty string.	"example or example" is not allowed.
Conversion of " into ^".	"example" is converted into ^"example^" to return "example". ^example or example^ is not allowed.
To return ^ use ^.	^^example^^ returns ^example^.

Naming Rules for Instance Parameters

If you change instance parameter values, ensure that they satisfy the Participant-specific naming rules, page 31 so that the subsequent generation process can complete successfully.

In addition, entering incorrect values may change the **Data** status, page 186 of the instance to **Invalid** when you save the instance configuration.

NOTE: Naming rules may also apply if you use the hierarchical naming function.

Hierarchical Name Function

You can use the **Hierarchical Name** function to let EcoStruxure Process Expert manage the identifier, page 177 of an instance. The function is enabled by default.

If you select the hierarchical naming function, EcoStruxure Process Expert creates the identifier of an instance automatically through the concatenation of the following:

- The **Alias** of each application folder, page 172 in the path of the instance, starting from the system root folder.
- The **\$Name**, page 191 parameter of the instance.

For example, if the **\$Name** of an instance is *Motor_1* and the path to its location is: *System Root Folder\Folder_1\Folder_2*.

If you assigned the aliases *F1* and *F2* to *Folder_1* and *Folder_2* respectively.

Then, the identifier of the instance becomes *F1F2Motor_1*.

NOTE: The length of the instance identifier cannot exceed the maximum length, page 31; otherwise, the status of the instance is shown as **Invalid**., page 186

NOTE: EcoStruxure Process Expert excludes folders without alias from the hierarchical naming (for example, if you assign no alias to *Folder_1*, the identifier of the instance becomes *F2Motor_1*). You cannot assign an alias to the system root folder.

To use the hierarchical naming function, proceed as follows.

Step	Action
1	Right-click the instance and select Properties . Result: The Instance Editor opens.
2	Verify that Hierarchical Name is selected, enabling the hierarchical naming function; otherwise select it.
3	Close the Instance Editor .
4	Right-click the application folder that is to be considered for the hierarchical naming and select Properties . Result: The Properties tab of the folder opens.
5	Assign an alias to the folder by using the Alias parameter. NOTE: Keep the alias as short as possible to help avoid reaching the instance identifier length limit.
6	Repeat steps 4 and 5 for the other folders that you want to include in the hierarchical naming.
7	If required, modify \$Name of the instance in the Instance Editor to help avoid reaching the instance identifier length limit.

NOTE: EcoStruxure Process Expert applies the hierarchical naming when you move, copy, page 210, or import, page 254 instances in the **Application Explorer**.

Instance Parameters

You can configure the following instance parameters in the parameter pane of the **Instance Editor**.

Selection in the instance pane	Parameter	Data type	Description
Instance	\$Name	String	<p>Default value: Template identifier <i>_n</i> where <i>n</i> is an incremental number starting at 1. For Schneider Electric templates, the \$ suffix and the <i>_xy</i> prefix are omitted.</p> <p>Within the application of a system, the resulting instance identifier, page 177 must be unique for each instance using the same template; otherwise, the \$Name text box is outlined in red.</p> <p>Entering a character that is not allowed, page 31 changes the instance Status to Invalid, page 177.</p> <p>NOTE: If you enter the same value for two instances that use a different template and the resulting instance identifiers are identical, the \$Name text box is outlined in red if the identifiers of Control facets of the two instances are not unique anymore (for example, if both instances contain the <i>Interlocks</i> element, which is created by the same facet template).</p> <p>NOTE: EcoStruxure Process Expert changes the \$Name parameter of the instance if the value is not unique within the application following a paste, page 214 or import, page 254 operation.</p>
	\$Description	String	<p>Description of the instance.</p> <p>Default value: Blank.</p>
Instance	\$Area	Integer ⁽¹⁾	<p>Area that the instance is assigned to.</p> <p>Range: 1...255</p> <p>Default value: Blank.</p> <p>NOTE: If you assigned an area value to the folder, page 173 containing the instance, EcoStruxure Process Expert automatically assigns the same value to the instance unless you manually assign a value.</p> <p>If EcoStruxure Process Expert is unable to propagate the value of the Area parameter to an item because the item is locked, it proceeds as follows:</p> <ul style="list-style-type: none"> • It informs you about the locked item. • It rolls back propagated area values. • It reverts the Area parameter value that you entered in the folder properties to its previous value. <p>NOTE: The area value is also propagated to Supervision elements of instances such as genies. However, the change to blank (NULL) from a non-blank value is not propagated to <i>Genies</i>, page 508 that are assigned to a Supervision page (for example, changing the area value from 3 to 4 is propagated but not changing it from 3 to blank).</p>
Control node	–	–	<p>You can configure the values of the parameters of the core and optional Control services that the instance provides through the selected elements.</p> <p>The default values are defined in the respective elements of the template that the instance uses.</p> <p>For details on parameters of Schneider Electric application templates, refer to the EcoStruxure Process Expert help, page 104.</p>
Supervision node	–	–	<p>You can configure the values of the parameters of the optional Supervision services that the instance provides through the selected elements.</p> <p>The default values are defined in the respective elements of the template that the instance uses.</p> <p>For details on parameters of Schneider Electric application templates, refer to the EcoStruxure Process Expert help, page 104.</p>

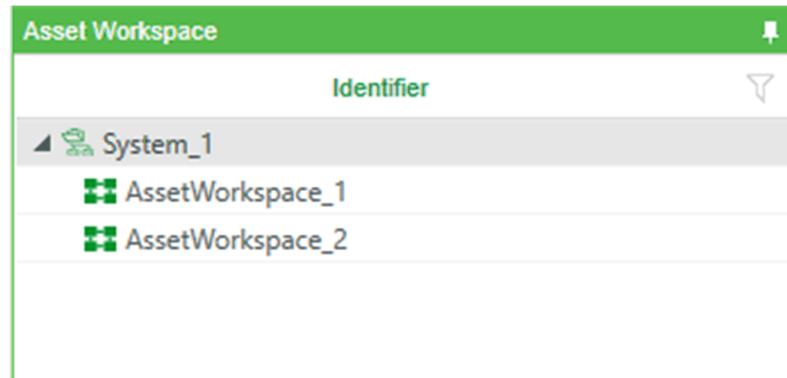
Selection in the instance pane	Parameter	Data type	Description
HyperLink node NOTE: Requires using templates of a Library that supports this functionality.	—	—	<p>You can configure the parameters of the optional Documents service that let you link documents to the instance. You can open these documents on computers running the operation client when using runtime navigation services (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>).</p> <p>You can link up to five documents.</p> <p>You must add documents to the content repository first.</p> <p>For details, refer to the topic describing how to link documents to instances (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>).</p> <p>You can configure the parameters of the optional URL service that let you associate web links and files to the instance. You can open these links on computers running the operation client when using runtime navigation services (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>).</p> <p>You can link up to five files or URLs.</p> <p>For details, refer to the topic describing how to associate web links and files to instances (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>).</p>
(1) You can also enter a string instead of an integer value. In this case, you must refine the Supervision project after generating it and use the same string as label to name the area; otherwise, the Supervision project does not build successfully. Refer to the help of the Supervision Participant for details.			

Using the Asset Workspace Pane

This topic describes the **Asset Workspace** pane. It lets you manage asset workspaces and open them in the **Asset Workspace Editor** to manage links between instances, page 200.

Asset Workspace Pane

The following figure shows an example of the **Asset Workspace** pane, which contains two user-created asset workspaces. *System_1* is the system root node.



Root Node Actions

The table describes the commands of the context menu of the system root node in the **Asset Workspace** pane.

Command	Description
Create Workspace	Creates a new empty asset workspace. This is a structural element that lets you group assets, which are existing instances of the application in order to create links between them.
Paste	Lets you create a copy of the asset workspaces and their content that you have added to the Clipboard. You can paste within the same system only.
Export	Lets you export, page 227 the data of existing asset workspaces and instances that belong to them to an editable application export file in comma-separated value (CSV) format.
Import	Lets you import, page 227 the data of workspaces and their instances, which are contained in an application export file (.csv).

Asset Workspace Actions

The table describes the commands of the context menu of asset workspaces.

Command	Description
Edit Workspace	Opens the asset workspace, page 200 in the Asset Workspace Editor . Lets you add and remove instances and edit links between them.
Show Instances	Opens the Show Instances window, which lets you see information about the instances that have been added to this asset workspace. Double-clicking an instance opens the asset workspace and highlights the instance.
Copy	Copies the data of the asset workspace and instances that belong to it to the Clipboard.
Export	Lets you export, page 227 the data of this asset workspace and instances that belong to it to an editable file in comma-separated value (CSV) format.
Delete	Deletes the asset workspace but does not delete the instances from the application nor the links between them.
Rename	Lets you rename the asset workspace.

Instance Indicators

Overview

This topic describes the various indicators that appear on folders and instances of the application.

Instance Progress Indicators

Overview

Various icons give an indication of the steps that remain before the instantiation and assignment, page 340 stages for each instance are completed.





The color of progress icons is used to indicate instance validity, page 198.

A progress icon is not visible while the checked-out icon, page 338 is shown.

NOTE: The icons are visible to any user who opens the application.

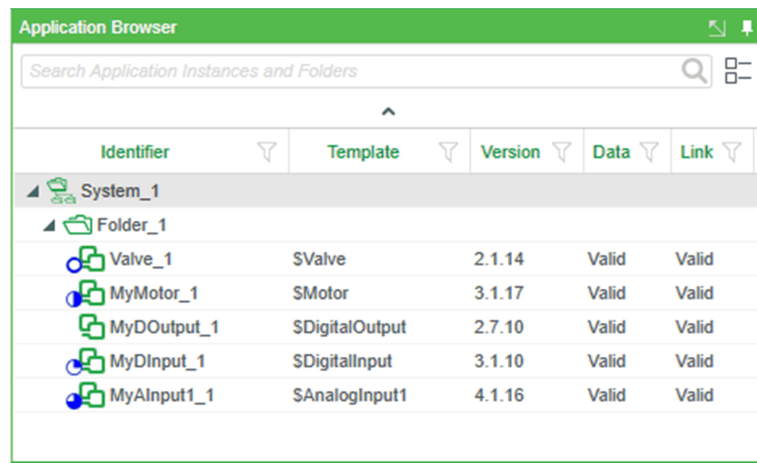
Meaning of Icons

The following progress information is valid only for instances whose facets are not yet generated.

Icon	Condition description
 (0% progress)	The instance is created or has been pasted without further action.
	Either one applies: <ul style="list-style-type: none"> The instance properties, page 188 have been modified at least once but no assignments have been performed. The instance progress is 50% or 75% and existing assignments, page 340 have been removed. The instance has been imported without impact on Participant projects.
	Either one applies: <ul style="list-style-type: none"> Facets of the instance have been assigned but there is still one type of Participant project to which no facet has been assigned (for example, to a Supervision project). At least one facet that is assigned has a status, page 379 other than Assigned. The instance has been imported and has changed the assignment status of at least one facet to Out Of Date or Deleted.
	At least one facet of the instance has been assigned to a Control and Supervision Participant project respectively and each one has the Assigned status.
No icon (100% progress)	All the facets of the instance have the Assigned status.

How Icons Appear

The following figure shows an example of the **Application Browser** displaying several instances with various completion states, which are indicated by progress icons.



The screenshot shows the 'Application Browser' window. It has a search bar at the top with the placeholder text 'Search Application Instances and Folders'. Below the search bar is a tree view showing a hierarchy: 'System_1' (with a folder icon) contains 'Folder_1' (with a folder icon). 'Folder_1' contains five items, each with a progress icon (a square with a circle inside) and a text label. The items are: 'Valve_1', 'MyMotor_1', 'MyDOutput_1', 'MyDInput_1', and 'MyAInput1_1'. To the right of the tree view is a table with five columns: 'Identifier', 'Template', 'Version', 'Data', and 'Link'. Each column has a filter icon (a funnel). The table contains five rows of data, corresponding to the items in the tree view.

Identifier	Template	Version	Data	Link
Valve_1	SValve	2.1.14	Valid	Valid
MyMotor_1	SMotor	3.1.17	Valid	Valid
MyDOutput_1	SDigitalOutput	2.7.10	Valid	Valid
MyDInput_1	SDigitalInput	3.1.10	Valid	Valid
MyAInput1_1	SAnalogInput1	4.1.16	Valid	Valid

Instance Validity Indicator

Overview

A dot or progress icon, page 196 that appears in red color indicates that the configuration of an instance is invalid.

When the icon appears on at least one instance, it appears automatically on folders in the hierarchy of the instance.

NOTE: The indicator icon is visible to any user who opens the application.

Meaning of the Icon

Level	Icon color	Condition description
Instance	Red (progress icon or dot for objects with 100% progress)	The instance Data status, page 177 and/or Link status, page 204 is invalid.
Folder	Red (dot)	The icon appears on at least one instance in the child structure.

Instance Write-Lock Indicator

Overview

A lock icon lets you see that an instance or folder is being used or modified by another process or user and that you cannot work on it.



Certain objects that are locked can be opened in read-only mode.

NOTE: The indicator icon is visible to any user who opens the application.

Meaning of the Icon

The lock icon is shown, for example, when either operation is performed on an instance or folder:

- Properties are open (instances only)
- Paste
- Rename
- Export/import
- Update/replace template

Linking Instances by Using the Asset Workspace Editor

This topic describes how to add instances to asset workspaces and link them by using their exposed interfaces. It also describes how to manage interface links.

Asset Workspace Editor

Overview

The **Asset Workspace Editor** lets you link the exposed application interfaces of an instance to the interfaces of other instances that are compatible and respect the cardinality of the roles (for example, 1 to 1, or 1 to n). The editor lets you create, modify, or delete links between instances, page 206 that belong to a same asset workspace.

You can use the editor, for example, to map a signal to an application object such as a limit switch signal to the appropriate input of a pump motor.

Some instances require that you link them to other instances based on the interface rules of their templates; otherwise the status of the **Link**, page 177 of the instance remains **Invalid** and you cannot proceed further in the system engineering life cycle.

NOTE:

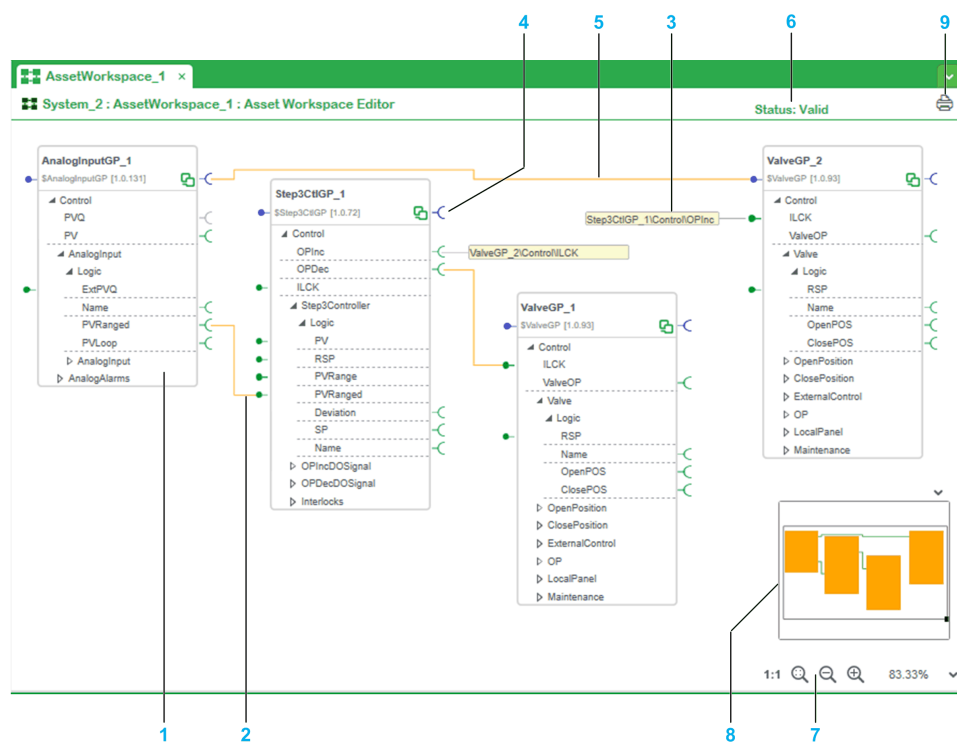
- Linking instances by using the **Asset Workspace Editor** helps avoid having to create such links later by refining the Control Participant project and provides a level of consistency in terms of Control resource names.
- Linking an instance to itself is not allowed. If this is required, edit the template used by the instance or refine the Control project to which facets of the instance are assigned.

Opening the Asset Workspace Editor

To open the **Asset Workspace Editor**, right-click an asset workspace and select **Edit Workspace**.

Asset Workspace Editor

The following figure shows an example of the **Asset Workspace Editor**.



Item	Description
1	Instances that have been added to the asset workspace by dragging them from the Application Browser or importing them. Linked instances that do not belong to the asset workspace are not shown. An instance can belong to an asset workspace even if it is not linked.
2	Existing application interface links and new ones. Only links between instances that belong to the same asset workspace are shown.
3	Interface link that was converted to a connector by using the link context menu command.
4	RTNS interfaces, page 208 (runtime navigation services). Appear in blue and are located in the header section of the instance.
5	RTNS interface links.
6	Indication of the link status of the instance that is selected and button to open the Validity Diagnosis window if the status is Invalid .
7	Zoom tools, page 1023.
8	Radar view, page 1031.
9	Lets you print the visible portion of the asset workspace.

NOTE: You can also move the editor background, page 108.

Instance Actions

The table describes the commands of the context menu of instances that appear in the **Asset Workspace Editor**.

Command	Description
Show/Hide Disabled	Lets you show or hide interfaces that are disabled (gray) and not available for connection.
Show/Hide Unbound	Lets you show or hide interfaces that are not connected to another instance in the same asset workspace. RTNS interfaces are always shown.
Remove Instance	Removes the instance from this asset workspace but does not delete it from the application. In addition, if the instance is linked to another instance in the same asset workspace, you can perform either action: <ul style="list-style-type: none"> Delete the links that this instance has with any other instances in the same asset workspace. (Links to instances that are not in this asset workspace are maintained.) Remove the instance but maintain its links.
View Assignments	Opens the View Assignments window, page 392.
Properties	Opens the instance, page 182 in the Instance Editor .

Link Actions

The table describes the commands of the context menu of links that appear in the **Asset Workspace Editor**.

Command	Description
Convert to Connector	Lets you toggle between binding styles (see EcoStruxure Process Expert, Global Templates Reference Manual).
Convert to Line	
Go to	Opens a submenu that indicates the following: <ul style="list-style-type: none"> The identifiers of the two instances that are linked. For each instance, the name of the element that exposes the linked interface and its path.
Delete	Deletes the link.

Impact of Modification of Instance Properties

When you save changes to an instance in the **Instance Editor** applicable changes are automatically reflected on the instance in the **Asset Workspace Editor**. If a change disables an interface that was linked, the link is automatically removed.

This may set the **Link** status, page 204 of either linked instances to **Invalid**.

Viewing Interfaces and Interface Description

Not all interfaces of instances that are displayed in the **Asset Workspace Editor** are visible by default. You may need to expand nodes of the interface to show enabled (green) or disabled (gray) interfaces.

To enable an interface, edit the instance and select the element that contains the interface, and/or configure the element parameter that enables it.

Move the pointer over any interface in the editor to display a description of the interface in a tooltip.

NOTE: Verify that disabled and unbound interfaces are shown by using the corresponding commands in the context menu of instances.

Using the Simplified View

Overview

Zooming out to the lowest level in the **Asset Workspace Editor** switches the display to the simplified view.

This view mode lets you see at a glance the instances and the relation between them without showing the details of links, interfaces, and elements of instances.

Some engineering functionality is not available.

To exit the simplified view, zoom in.

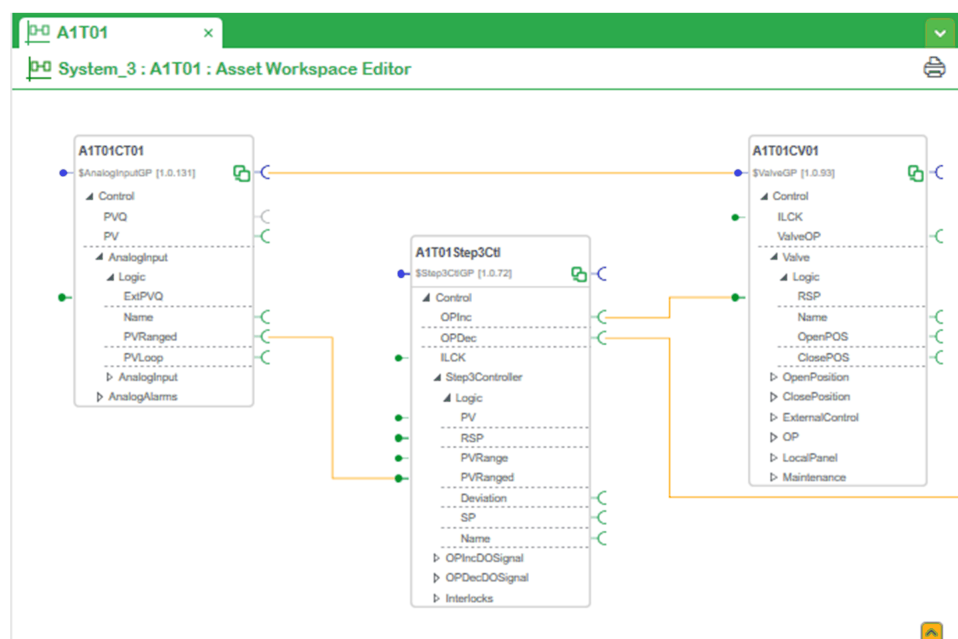
Restrictions of the Simplified View

The table describes the restrictions that are in effect while the simplified view is active.

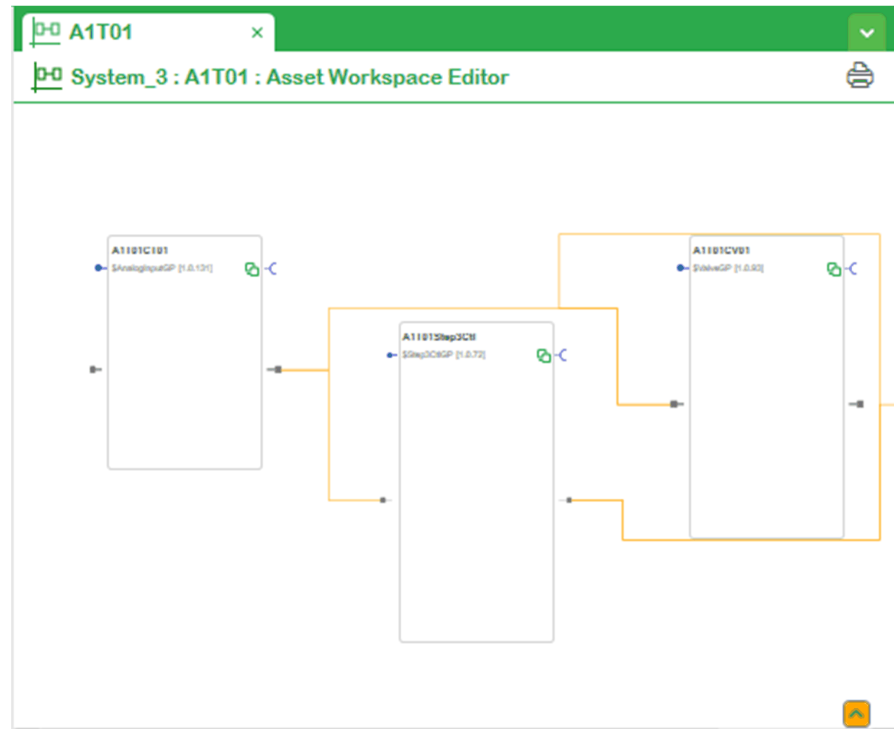
Actions or object	Restrictions
Actions related to links such as creating and moving	Actions are not allowed
Elements and interfaces	Are not shown
Multiple links to and from an instance	Are shown as a single line

Description

The following figure shows an example of instances as they normally appear in the **Asset Workspace Editor**.



The following figure shows the same view when simplified view is active (the view is enlarged for better visibility).



- Instances maintain their relative size and position to preserve the aspect of the layout.
- Links between instances are grouped into a single line.

Instance Link Status

Overview

The **Link** status of an instance can be one of the following:

- **Valid:** The interface rules and interface element rules that are defined at the template level are satisfied.
- **Invalid:** At least one of the defined interface rules or interface element rules are not satisfied.

You cannot assign to projects instances that have a **Link** status that is **Invalid**.

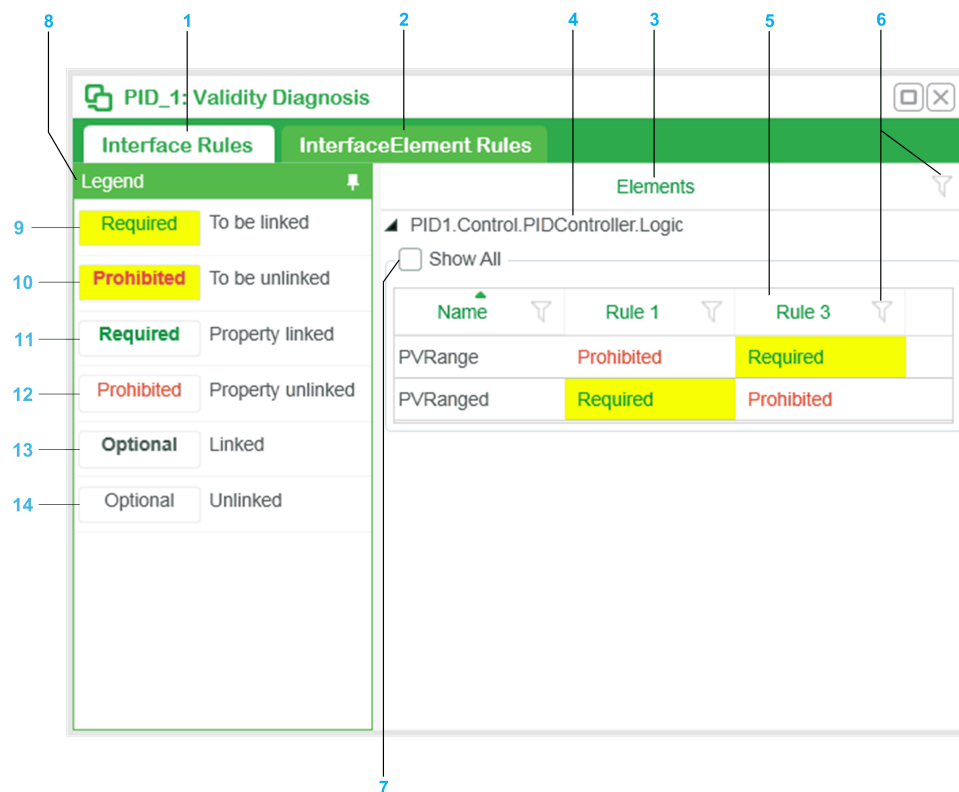
When the status is displayed as **Invalid**, click the validity status information button



or **Invalid** in the **Application Browser** to open the **Validity Diagnosis** window, which contains information about the cause of the invalid status.

Validity Diagnosis Window - Interface Rules

The following figure shows an example of the **Interface Rules** tab of the **Validity Diagnosis** window.



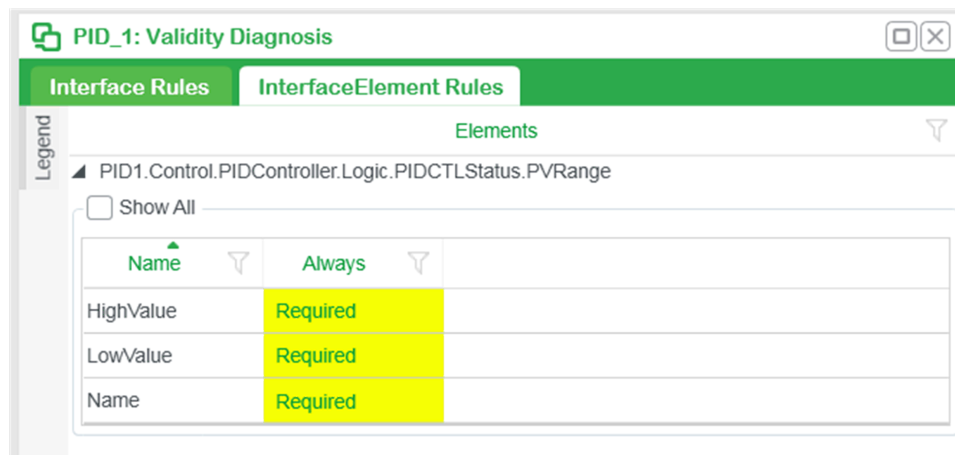
Item	Description
1	Tab displaying information on interface rules of the instance with invalid Link status.
2	Tab displaying information on rules of interface elements, page 206 of the instance with invalid Link status.
3	By default, this section displays only the elements of the instance for which at least one interface rule is not satisfied.
4	Path to the element whose interface rules are shown. You can view the element by editing the instance, page 182.
5	Shows for each element, the rules that are defined and their status.
6	Filter, page 120.
7	Select the check box to view elements of the instance for which an interface rule is defined independently if the rule is satisfied or not.
8	Hover with the pointer over Legend to view an explanation of the statuses displayed in the rule columns.

The table describes the content of the **Legend** tab of the **Validity Diagnosis** window.

Item	Description
9	The link is required to satisfy the rule.
10	Remove the link; otherwise the Status remains Invalid .
11	The link is made, which satisfies the rule.
12	No link is made in accordance with the rule.
13	The link is optional and a link is made.
14	The link is optional and no link is made.

Validity Diagnosis Window - Interface Element Rules

The figure shows the **Interface Element Rules** tab, which is available if rules exist for elements of an interface. In this example, interface *PVRange*, which is one of the interfaces shown in the **Interface Rules** tab has three elements for which a rule is defined but not satisfied.



Linking Instances

Linking Instances That Are Open for Editing

While an instance is open in the **Instance Editor**, you can add it to and remove it from, page 200 the **Asset Workspace Editor** but you cannot create, modify, nor delete links for it.

Naming Rules for Variable Names

The name of the variable that is created by linking two instances needs to satisfy the naming rules, page 32 for the subsequent generation process to complete successfully.

Creating Interface Links

To create an interface link between two or more instances by using the **Asset Workspace Editor**, proceed as follows.

Step	Action
1	Right-click an asset workspace and select Edit Workspace . Result: The asset workspace opens in the editor.
2	From the Application Browser pane, drag the instances that you want to link to the editor and position them as needed. Result: The instances are shown in the editor with their interfaces exposed. If links already exist between these instances, they are shown as well. NOTE: You can drag a maximum of 20 instances at once either by selecting them individually and/or selecting folders.
3	From either instance, position the pointer over an interface and when the pointer turns into a hand, click and move the pointer away from it while holding down the left mouse button. Result: A line appears and follows the pointer. You can release the mouse button. NOTE: Press ESC to return to the normal pointer function.
4	Move the pointer over the other instance. Result: Compatible and enabled interfaces of the instance are automatically highlighted and the line snaps to the nearest matching connector.
5	Click to confirm the connection or move the pointer to another highlighted interface until the line snaps to it. Result: A click performs the following actions: <ul style="list-style-type: none"> Links the interfaces of both instances with a line. Changes the status of the active instance to Valid if the interface rules are satisfied (if applicable). NOTE: To delete the link, right-click the line and select Delete .
6	Verify that the link status of the instances is Valid ; otherwise click Invalid in the Link column of the Application Browser or in the Asset Workspace Editor to open the Validity Diagnosis window and verify which links are required. Create these links for the instance status to become Valid .

Changing the Destination of a Link

A destination interface is represented graphically by a plug .

It is typically the interface with the *REF* role.

To change the destination of an interface link, proceed as follows.

Step	Action
1	Select the link whose destination you want to change. Result: A black dot appears on the destination connector.
2	Drag the black dot to a new compatible destination connector of the same or another instance and release the mouse button. Result: The link is moved to the new destination.

Enabled Interfaces That Cannot Be Connected

The interface of an instance may appear as enabled, unbound, and be compatible but the link does not snap to it. This can be the case if this interface is linked to another instance that is not in the same asset workspace and the interface accepts only one connection at a time.

To view the links that exist for an instance, open it in the **Links Editor** so that it is the active instance or open the **Inspect Instance** window, page 211.

Using the RTNS Interface

Purpose of the RTNS Interface

When no application link is required between instances, you can use the RTNS interface to link them so that they appear in the **Linked Production Assets** section of the operation client (see *EcoStruxure™ Process Expert, Runtime Navigation Services, User Guide*).

The cardinality of the interface is *n* to *n* and it can be connected only to another RTNS interface.



Using the RTNS Interface

You can create a link between any instances that exist in the application by using this interface. The procedure to create an RTNS link is the same as the one to create a regular interface link.

NOTE:

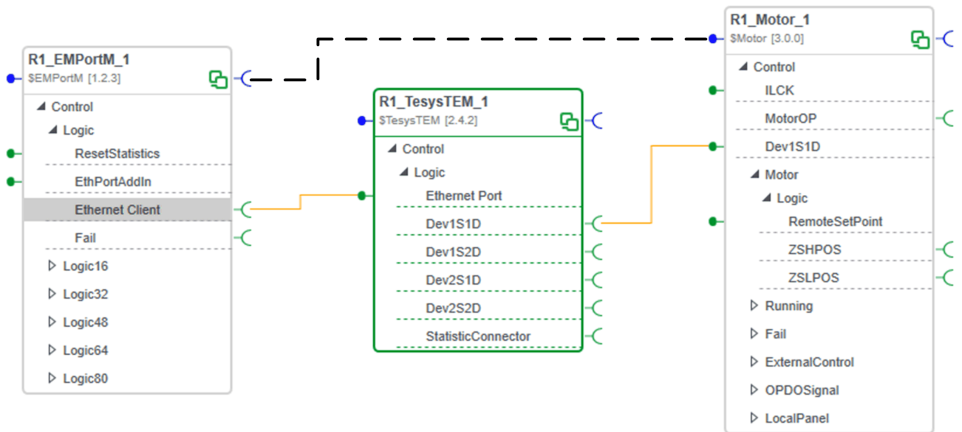
- Creating an RTNS link between two instances that are already linked by an application link has no effect.
- Adding or removing an RTNS link between two instances does not change the assignment status, page 425 of facets of these instances.

RTNS Interface Identifiers and Roles

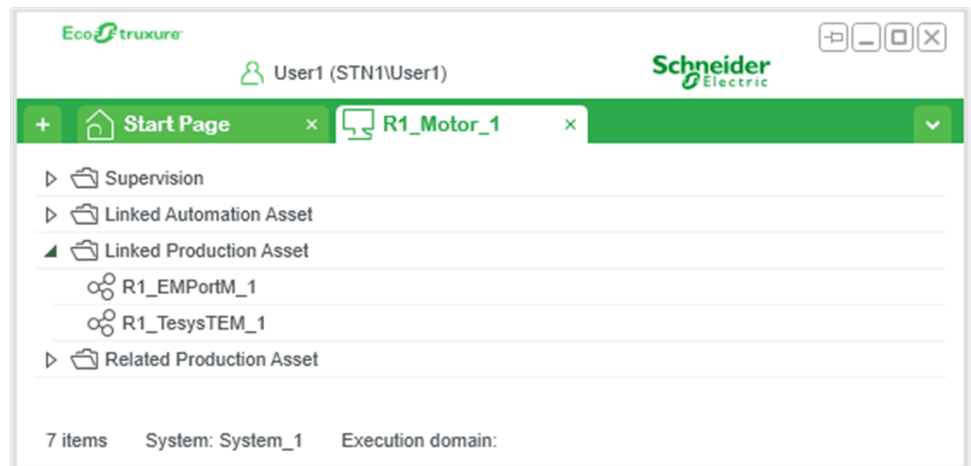
Representation	Identifier	Role
	<i>NullInterfaceRoleA</i>	A
	<i>NullInterfaceRoleB</i>	B

Example

In the following example, three instances are linked but no link exists between *R1_EMPortM_1* and *R1_Motor_1*. To be able to view also information of *R1_EMPortM_1* in the operation client when accessing runtime navigation services from *Motor_1*, you can create an RTNS link with *R1_EMPortM_1* (shown here as a dotted line for illustration purposes).



The following figure shows the result in the EcoStruxure Process Expert operation client after creating the RTNS link when you access runtime navigation services from *R1_Motor_1*. Instance *R1_EMPortM_1* also appears in the **Linked Production Assets** section because of its RTNS link with *R1_Motor_1*.



Managing Application Folders and Instances

Overview

This section describes the **Inspect Instance** command and the functionality of EcoStruxure Process Expert that lets you manage many application folders and instances.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Viewing Instance Configuration

Overview

The **Inspect Instance** command lets you view the entire composition of any instance of the application along with current values of its parameters, elements, functions, and interfaces.

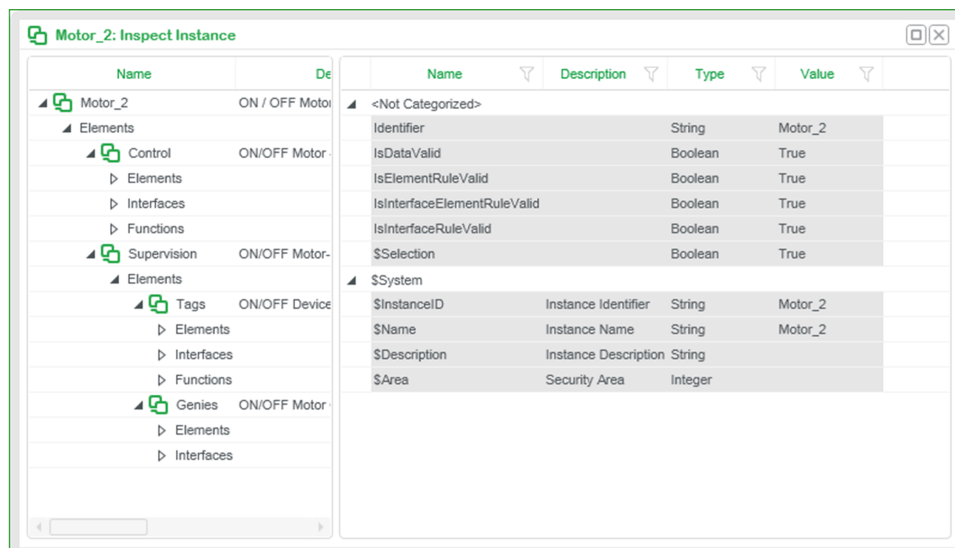
The functionality does not allow you to change values.

Opening the Inspect Instance Window

In the **Application Browser**, right-click an instance and select **Inspect Instance**.

Inspect Instance Window

The following figure shows an example of the **Inspect Instance** window for an instance of the application.



The table describes the various components of an instance that you can view in the left-hand pane of the **Inspect Instance** window.

Component	Description
Instance identifier	Identifier of the instance from which you have opened the Inspect Instance window.
Elements	Elements of the instance grouped by Participant (for example, Control) Inside each category, you can view the hierarchy of elements as they are visible in the Instance Editor down to the constituent level.
Functions	Binding functions (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) that are used by the parent element. For example, it lets you view information related to the location of an instance in the application, which can be provided by the LocationGet function.
Interfaces	Interfaces that are referenced by the parent element. For each interface, you can view its identifier, role used by the element, its template identifier, parameters, and their current value.

The table describes the parameter information that you can view in the right-hand pane of the **Inspect Instance** window. The data pertains to the component that is selected in the left-hand pane of the window.

Header	Description
Name	Name of the parameters of the element grouped by category.
Description	Description of the parameter.
Type	Data type of the parameter.
Value	Current resolved value of the parameter, page 183 or property.

Changing the Application Hierarchy

Changes Impacting Assigned and Generated Supervision Facets

When you change the hierarchy of the application, facets of instances that are assigned to a Supervision project and generated are affected when they generate equipment based on this hierarchy, page 437. The table details these changes and their impact.

Change	Impact
Moving an application folder to another parent folder.	The assignment status of any facets of any instances that the folder and its subfolders contain is set to Out Of Date .
Modifying the identifier of a folder.	
Moving an instance to another folder.	The assignment status of any facets of the instance is set to Out Of Date .

For information on steps that you need to take to integrate the changes into the system, refer to *Generating Projects After Changes*, page 430.

Copying and Pasting Application Folders and Instances

Description of the Functionality

You can copy and paste instances and application folders:

- Within the applications of a system.
- Between applications of different systems.

The **Notification Panel** provides detailed information about the result of the paste process.

NOTE: Pasting instances affects the [instance count](#), [page 180](#).

The table describes the effects of the copy and paste functionality when applied to application folders and instances.

Item	Effect
Folders and instances	<p>The copied folder and any instances that it contains, including any subfolders and their contents are pasted.</p> <p>NOTE:</p> <ul style="list-style-type: none"> • When you select the Copy command for a folder, this folder, any subfolders, and any instances they contain are copied. This is the case even if you initially selected, in addition to the folder, only one of the instances or subfolders it contains with the intention to copy only the selected items. • If you delete an instance or a folder after copying it, you cannot paste it. Also, if you copy a folder containing several instances and you delete an instance in this folder, when you paste the folder, the instance that you deleted is not pasted.
Folder identifier and instance name	<p>If the folder identifier already exists at the same hierarchical level, or the instance identifier already exists in the target location, the folder identifier or the instance name is changed by adding an incremental numerical value as suffix; otherwise the identifier or instance name is not changed.</p> <p>For example, <i>Motor_1</i> becomes <i>Motor_1_1</i>, and <i>Folder_1</i> becomes <i>Folder_1_1</i>.</p> <p>NOTE: If the hierarchical naming function, page 189 is activated for an instance, the instance identifier changes according to the new location of the instance and the alias of the folders in its path.</p>
Interface links	<p>Interface links are maintained between pasted instances like they were defined between the original instances if you copy and paste the instances that are linked at the same time.</p> <p>Interface links can be copied across systems under the same conditions.</p>
Other user-defined configuration data	<p>EcoStruxure Process Expert does not modify the data during the paste process. Therefore, relative expressions created at the instantiation stage could produce different results when applied to the new location.</p> <p>Also, when you paste a folder and/or instance in a folder to which you have assigned an area value, page 172, this value is propagated to the pasted items if they do not have an area value yet and if they are not locked. For details refer to the topic describing the propagation of the area parameter value, page 173.</p>

Aborting Paste Tasks

After selecting the **Paste** command, the Abort icon is displayed in the [notification panel](#), [page 85](#). Click the icon to cancel the task. No instances and folders are pasted.

Only the user who has selected the command is allowed to abort it.

NOTE: If you are pasting only a few items, the task may complete before you are able to click the icon.

Copying and Pasting Instances

To copy and paste instances, proceed as follows.

Step	Action
1	Expand the folder that contains the instances that you want to copy.
2	Select the instances. You can make multiple selections within one folder by holding: <ul style="list-style-type: none">• The Shift key for contiguous instances.• The Ctrl key if instances are non-contiguous.
3	Right-click a selected instance and select Copy .
4	Right-click the destination folder and select Paste . You can select multiple folders as targets. NOTE: If the target is not valid for pasting, the command is not available. Also, you cannot paste in the Application Browser in grid view mode.

Copying and Pasting Application Folders

To copy and paste application folders, proceed as follows.

Step	Action
1	Right-click the folder that you want to copy and select Copy . You can make multiple selections within the Application Browser by holding the Ctrl key.
2	Right-click the destination folder and select Paste . You can select multiple folders as targets. NOTE: If the target is not valid for pasting, the command is not available. Also, you cannot paste in the Application Browser in grid view mode.

Moving Application Folders and Instances

Overview

You can move instances and application folders:

- Within the applications of a system.
- Between the applications of different systems.

Both actions change the structure of the application, page 159. A notification is displayed informing you of the possible impact on the assignment status of already assigned facets before executing the move operation.

Once the move operation is completed, if changes to folder and/or instance identifiers are required to satisfy naming rules, the **Move Summary** window, page 218 opens to inform you of the changes.

NOTE:

- You cannot move instances within a folder.
- You can change the position of folders without modifying the hierarchy of the application, page 219.

Moving Folders and Instances

By using the **Application Browser** pane, you can select a combination of folders and instances to move them at once. This is possible in tree view mode only.

When you move a folder, any subfolders and any instances they contain, are moved.

This is the case even if you initially selected, in addition to the folder, only one of the instances or subfolders it contains with the intention to move only the selected items.

NOTE:

- You can make multiple selections by holding **Shift** for contiguous or **Ctrl** for non-contiguous items.
- If sorting and/or filters are applied, they are also applied to instances and folders that you have moved.

Effects of the Move Operation

The table describes the effects of the move operation when applied to application folders and instances.

Item	Effect
Folders and instances	Moves the folder and any instances that it contains, including any subfolders and their contents.
Folder identifier and instance name	<p>If the folder identifier already exists at the same hierarchical level, or the instance identifier already exists in the target application, the folder identifier or the instance name is changed by adding an incremental numerical value as suffix; otherwise the identifier or instance name is not changed.</p> <p>For example, <i>Motor_1</i> becomes <i>Motor_1_1</i>, and <i>Folder_1</i> becomes <i>Folder_1_1</i>).</p> <p>NOTE: If the Hierarchical Name function, page 189 is activated for an instance, the instance identifier changes according to the new location of the instance and the alias of the folders in its path.</p>
Interface links	<p>Interface links are maintained between instances if you move the instances that are linked at the same time.</p> <p>NOTE: Moving interface links across systems is supported.</p>
Other user-configured properties	<p>EcoStruxure Process Expert does not modify the data during the move process. Therefore, relative expressions that was created at the instantiation stage could produce different results when applied to the new location.</p> <p>Also, when you move a folder and/or instance to a folder to which you have assigned an area value, page 172, this value is propagated to the moved items if they do not have an area value yet and if they are not locked. For details refer to the topic describing the propagation of the area parameter value, page 173.</p>

Moving Instances

To move instances, proceed as follows.

Step	Action
1	Expand the folder that contains the instances that you want to move.
2	<p>Select the instances.</p> <p>You can make multiple selections by holding Shift for contiguous or Ctrl for non-contiguous instances.</p> <p>Result: The selected instances are highlighted.</p> <p>NOTE: To move instances across systems, use two workspaces, page 107; one for each application.</p>
3	<p>Drag one of the highlighted instances to the target application folder.</p> <p>Result: The identifiers of the instances you are moving appear in a tooltip.</p>
4	<p>Release the mouse button.</p> <p>Result:</p> <ul style="list-style-type: none"> The selected instances are moved to the new location. When no sorting is applied, instances are inserted in ascending alphabetical order among existing instances. The Move Summary window opens if a change was performed to comply with applicable naming rules.
5	Close the Move Summary window.

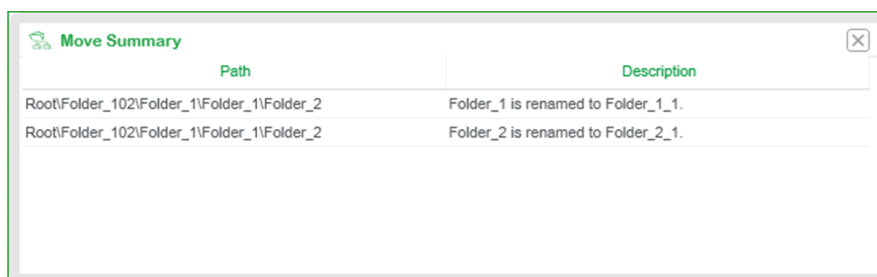
Moving Application Folders

To move application folders, proceed as follows.

Step	Action
1	<p>Select the folders that you want to move.</p> <p>You can make multiple selections by holding Shift for contiguous or Ctrl for non-contiguous folders.</p> <p>Result: The selected folders are highlighted.</p> <p>NOTE: To move folders across systems, use <i>two workspaces</i>, page 107; one for each application.</p>
2	<p>Drag one of the highlighted folders that you want to move to the target.</p> <p>Result: The identifiers of the folders you are moving appear in a tooltip.</p>
3	<p>Release the mouse button.</p> <p>Result:</p> <ul style="list-style-type: none"> The folders and their contents are moved to the new location. When no sorting is applied, folders are added at the end of existing folders. The Move Summary window opens if a change was performed to comply with applicable naming rules.
4	Close the Move Summary window.

Move Summary Window

The following figure shows an example of the **Move Summary** window.



Header	Description
Path	Indicates the path to the folder or instance location after the move.
Description	Indicates the name change was performed on the folder and/or instance during the move.

Changing the Position of Folders Without Changing the Hierarchy

You can change the position of folders inside the structure without modifying the hierarchy of the application. You can do this one folder at a time.

Step	Action
1	Verify that no sorting is applied in the Application Browser pane of the Application Explorer and that tree view mode is selected.
2	<p>In the Identifier column, start dragging a folder along the folder structure until one of the following messages appears in a tooltip:</p> <ul style="list-style-type: none">• Move X before Y.• Move X after Y <p>Where:</p> <ul style="list-style-type: none">• X is the identifier of the folder that you are moving.• Y is the identifier of the target folder.
3	<p>Release the mouse button.</p> <p>Result: The folder is relocated to the selected position.</p> <p>NOTE: If you release the mouse button while only the identifier of the folder that you are moving appears in the tooltip, you are changing the hierarchy of the folder structure, page 216.</p>

Updating and Replacing Templates of Instances

Overview

In the **Application Browser**, you can perform the following actions on the template that is used by an instance of the application. The working principle of both commands is the same. The difference resides in the choice of templates provided by each command.

Update Template	Lets you update the template that is used by the instance with the latest available version. When you select the command from a folder, the command applies to any instance that is contained in the folder and any subfolders, and for which a template of a newer version is available.
Replace Template	Lets you replace the template that is used by one or more instances with another version of this template or a different template.

NOTE: The **Update Template** command is also available in the **Topology Explorer** for certain topological entities, page 622.

Updating Versus Replacing Templates

The following examples are given to illustrate the difference between the two commands.

Update Template command:

- The following templates, with usability state **Approved**, are available in the Global Templates Library: *\$Motor (2.0.0)*, *\$Motor (2.0.1)*, and *\$Motor (2.0.2)*
- Instance *Motor_1* uses template *\$Motor (2.0.0)*
- The command lets you update the template that is used by *Motor_1* with *\$Motor (2.0.2)* only (latest version).

Replace Template command:

- The following templates are available in the Global Templates Library: *\$Motor (2.0.1)*, *\$Motor (2.0.x)*, and *\$Valve (3.0.0)* with usability state **Approved**.
- Instance *Motor_1* uses template *\$Motor (2.0.1)*
- The command lets you replace the template that is used by *Motor_1* with either:
 - *\$Motor (2.0.x)*: Same template, any available version with the appropriate usability state.
 - *\$Valve (3.0.0)*: Different template.

Using the Commands on a Folder

When you select the **Update Template** or **Replace Template** command at the folder level, the command is performed on this folder, any subfolders, and any instances they contain.

This is the case even if you initially selected, in addition to the folder, only one of the instances or subfolders it contains with the intention to perform the command only the selected items.

Accessing the Commands

Update Template	Right-click one instance, a selection of several instances in the same or in various folders, the root folder, or any application folder, and select the command. The command executes only if a newer version of the template that is used by the instance is available in the Global Templates Library.
Replace Template	Right-click one instance or a selection of several instances in the same folder and select the command.

Description of the Functionality

EcoStruxure Process Expert proceeds with the update or replacement of a template in the following order:

- The template that is associated to the instance is updated or replaced.
- Interface links are processed.

The following conditions need to be satisfied:

- The template or template version that you want to use must be available in the Global Templates Library.
- It needs to have the usability state **Approved** (see EcoStruxure Process Expert, Global Templates, Reference Manual) or **Deprecated**.

NOTE: Refer to *Importing Templates*, page 1042.

The table describes the effects on the various template components when you update or replace the template that is used by an instance.

Template component	Identifier in new template ⁽¹⁾	Effect
Element	The identifier of the element and its path in the composition of the template is the same as in current template.	<ul style="list-style-type: none"> • The element is maintained or updated. If updated, the status of the corresponding facet is set to Out Of Date if the facet was generated. Refinements that you have made are preserved if the element that creates the facet encapsulates the same constituents. • The element selection that you have made in the current template is applied to the new template. For example, in the current and the new template, the <i>Interlocks</i> element is at the path <i>Control/Interlocks</i>. If it was selected in the current template, it is selected in the new template. • If the identifier of the template used by the corresponding facet (Facet Template column) is different, the status of the facet is set to Out Of Date if the facet was generated. <p>NOTE: If you replace a control module template (highest level template) by one that has a different identifier but an identical composition, no changes are made to the instance, except for the template identifier.</p>
	The identifier of the element does not exist in the current template or the path to the element is different.	<p>The element is added and the corresponding facet is created.</p> <p>An optional element that is not part of the current template keeps its default value (selected or cleared).</p> <p>NOTE: Any element of the current template whose identifier does not exist in the new template or whose path is different is removed.</p>
Parameter value	Is same as in current template	Explicitly defined parameters of the current template keep their respective value in the new template.
	Is not in current template	New parameters that are not part of the current template keep their default value.
Interface link	—	<p>EcoStruxure Process Expert re-establishes interface links that are valid after the template update or replacement in their original configuration.</p> <p>Interface links that are not valid after the template update or replacement are discarded because, for example:</p> <ul style="list-style-type: none"> • The cardinality of the interface role is not respected. • A linked element does not exist anymore. • An incompatibility of the type and/or role of the interface.
(1) Refers to either a different version of the same template or a different template		

This example illustrates the impact of a template update operation on generated Control facets of an instance. In this scenario, instance *Valve_1* uses template *\$Valve* version 1.3.9. The optional *Maintenance* service is selected at the instance level.

The following figure shows the composition of the template as it appears in the **Instance Editor**. Elements appear in the **Name** column and the identifier of the template that is used by the corresponding facet in the **Template** column.

Valve_1: Instance Editor Status: Valid

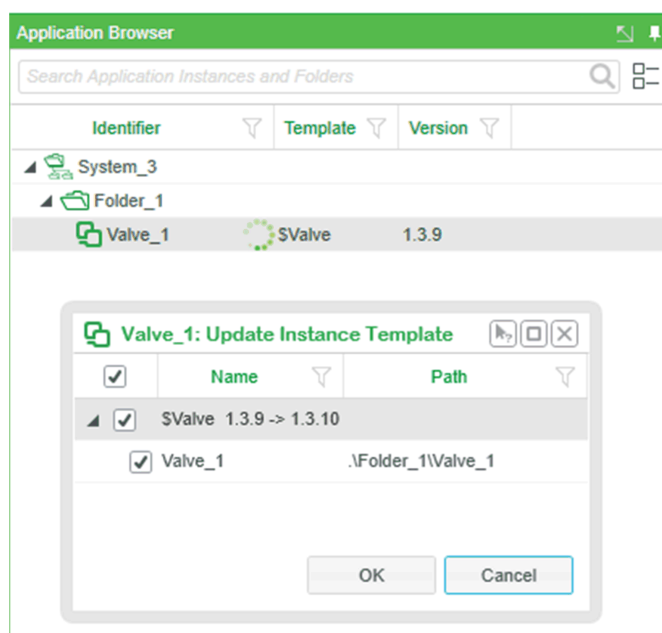
Name	Description	Template	Version
Valve_1 (0/262)	ON/OFF Valve	\$Valve	1.5.0
Control (0/23)	ON/OFF Valve	\$VALVE_UC	1.2.12
Valve (0/3)	ON/OFF Device Control - Unity Control	\$DEVCTL_UC	1.0.8
Logic (0)	ON/OFF Device Control Unity Logic	\$DEVCTL_UL	1.0.8
<input checked="" type="checkbox"/> LowLimitSwitch (0)	Digital Input Signal	\$DISignal_UL	5.6.9
<input checked="" type="checkbox"/> HighLimitSwitch (0)	Digital Input Signal	\$DISignal_UL	5.6.9
<input type="checkbox"/> EXTCTLCond (0)	Digital Input Signal	\$DISignal_UL	5.6.9
<input checked="" type="checkbox"/> OPCControl (0)	Digital Output Signal	\$DOSignal_UL	5.4.5
<input checked="" type="checkbox"/> Interlocks (0)	Interlock Condition summary with Unity HMI	\$CONDSUM1_UC	1.0.8
<input checked="" type="checkbox"/> Maintenance (0)	Device Maintenance Unity Logic with Unity HMI	\$DEVMNT_UC	1.0.3
LocalPanel (0/12)	Device Local Panel	\$DEVL_P_UC	2.1.9

The following figure shows the Control facets (**Facet** column) of instance *Valve_1*, which are assigned to a section of a Control project. By relating this information to the one shown in the **Instance Editor**, you can see, for example, that element *LowLimitSwitch* creates facet *Valve_1_ZSL* by using template *\$DISignal_UL*. This information is also visible at a glance in the **View Assignments** window, page 392.

FBDSection_1 - Assignments

Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
\$Valve	Valid	Valve_1_DEVCTL	\$DEVCTL_UL	Control\Valve	0	Assigned	Generated
\$Valve	Valid	Valve_1_ZSL	\$DISignal_UL	Control	1	Assigned	Generated
\$Valve	Valid	Valve_1_ZSH	\$DISignal_UL	Control	2	Assigned	Generated
\$Valve	Valid	Valve_1_VALVE_OP	\$DOSignal_UL	Control	3	Assigned	Generated
\$Valve	Valid	Valve_1_CONDSUM1	\$CONDSUM1_UL	Control\Interlocks	4	Assigned	Generated
\$Valve	Valid	Valve_1_DEVMNT	\$DEVMNT_UL	Control\Maintenance	5	Assigned	Generated

\$Valve 1.3.9 is updated to version 1.3.10 by using the **Update Template** command. *\$Valve* 1.3.10 has a modified *Interlocks* service, which is provided by the modified *CONDSUM1_UL* template. All other elements are identical.



The result of the template update operation shows that one facet has a modified assignment status. All facets remain generated.

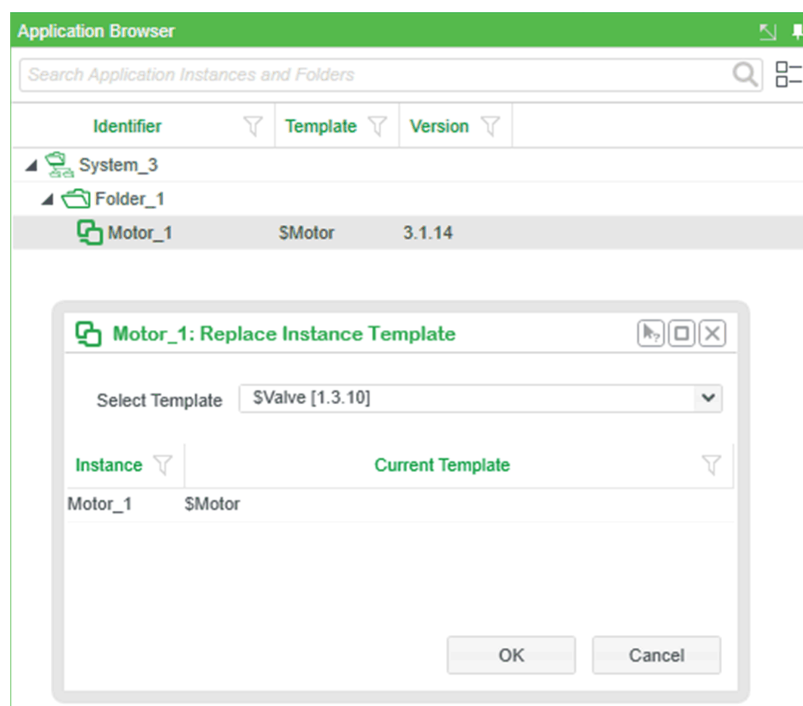
Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
\$Valve	Valid	Valve_1_DEVCTL	\$DEVCTL_UL	ControlValve	0	Assigned	Generated
\$Valve	Valid	Valve_1_ZSL	\$DISignal_UL	Control	1	Assigned	Generated
\$Valve	Valid	Valve_1_ZSH	\$DISignal_UL	Control	2	Assigned	Generated
\$Valve	Valid	Valve_1_VALVE_OP	\$DOSignal_UL	Control	3	Assigned	Generated
\$Valve	Valid	Valve_1_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	4	Out Of Date	Generated
\$Valve	Valid	Valve_1_DEVMNT	\$DEVMNT_UL	ControlMaintenance	5	Assigned	Generated

Item	Description
1	The facet created by the <i>CONDSUM1_UL</i> template (Interlocks service) has a status of Out Of Date because the already generated constituents (from \$Valve 1.3.9) are not consistent with the constituents of the modified <i>CONDSUM1_UL</i> template anymore. Generating the section injects the constituents coming from the modified <i>CONDSUM1_UL</i> template into the logical Control Participant project.
2	The optional <i>Maintenance</i> service remains selected after the template update because it is also available in the latest version of \$Valve.

This example illustrates the impact of a template replacement operation on generated Control facets of an instance. In this scenario, instance *Motor_1* uses template *\$Motor*. *\$Motor* has an *Interlocks* service provided by the *CONDSUM1_UL* template. The optional *Maintenance* service is selected at the instance level.

Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
\$Motor	Valid	Motor_1_DEVCTL	\$DEVCTL_UL	ControlMotor	0	Assigned	Generated
\$Motor	Valid	Motor_1_CONDSUM	\$CONDSUM_UL	ControlFailures	1	Assigned	Generated
\$Motor	Valid	Motor_1_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	2	Assigned	Generated
\$Motor	Valid	Motor_1_DEVMNT	\$DEVMNT_UL	ControlMaintenance	3	Assigned	Generated

\$Motor is replaced with *\$Valve* by using the **Replace Template** command. Compared to *\$Motor*, *\$Valve* has a modified *Interlocks* service, which is provided by the modified *CONDSUM1_UL* template (located at the same path). The *Maintenance* service is identical in both and provided by the same template at the same path. Their other elements are either different (for example, *\$CONDSUM_UL*) or located at a different path (for example, *\$DEVCTL_UL*) in the composition of the control module template.



The result of the template replacement operation shows that several facets have a modified assignment status.

FBDSection_1 - Assignments							
Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
\$Motor	Deleted	Motor_1_DEVCTL	\$DEVCTL_UL	Control\Motor	0	Deleted	Generated
\$Motor	Deleted	Motor_1_CONDSUM	\$CONDSUM_UL	Control\Failures	1	Deleted	Generated
\$Valve	Valid	Motor_1_CONDSUM1	\$CONDSUM1_UL	Control\Interlocks	2	Out Of Date	Generated
\$Valve	Valid	Motor_1_DEVMNT	\$DEVMNT_UL	Control\Maintenance	3	Assigned	Generated

1 2 3

Item	Description
1	These two facets are deleted because in <i>\$Valve</i> , <i>\$DEVCTL_UL</i> is at a different path and <i>\$CONDSUM_UL</i> does not exist. Generating the section removes the corresponding constituents from the logical Control Participant project.
2	The facet providing the <i>Interlocks</i> service is set to Out Of Date because <i>Valve</i> contains this service (provided also by <i>\$CONDSUM1_UL</i>) but a modified version of it. Already generated constituents (from <i>\$Motor</i>) are not consistent with the constituents of the modified <i>CONDSUM1_UL</i> template anymore. Generating the section injects the constituents coming from the modified <i>CONDSUM1_UL</i> template into the logical Control Participant project.
3	The optional <i>Interlocks</i> service is not affected because it is identical for both templates. The Instance Template column indicates that the facet is now created by the <i>Valve</i> template.

In the **Assignment Editor**, you can see the facets that are specific to *Valve* and that have been created for the *Motor_1* instance. You need to assign these to the section as needed. These facets have the **Candidate** status. By default, they are assigned automatically, page 381.

Motor_1 - Facets						
Instance	Instance Template	Facet	Facet Template	Path	System	Project
Motor_1	\$Valve	Motor_1_CONDSUM1	\$CONDSUM1_UL	Control\Interlocks	Assigned	Done
Motor_1	\$Valve	Motor_1_DEVCTL	\$DEVCTL_UL	Control\Valve	Not Assigned	Candidate
Motor_1	\$Valve	Motor_1_DEVMNT	\$DEVMNT_UL	Control\Maintenance	Assigned	Done
Motor_1	\$Valve	Motor_1_VALVE_OP	\$DOSignal_UL	Control	Not Assigned	Candidate
Motor_1	\$Valve	Motor_1_ZSH	\$DISignal_UL	Control	Not Assigned	Candidate
Motor_1	\$Valve	Motor_1_ZSL	\$DISignal_UL	Control	Not Assigned	Candidate

NOTE: The same that is described in the above examples for Control facets also applies to Supervision facets of the instance.

Updating or Replacing Templates of Linked Instances

When instances are linked by interface links (**Asset Workspace Editor**), update the templates of these instances at once, page 225 to maintain the links.

Updating or Replacing Templates Containing a New Version of a Type

When you update or replace the template used by an instance and the Control facets that are referenced by the instance are already generated, if the new template that you selected contains a version of the type that is different from the one that was generated, you must select the version to be used when you generate the Control Participant project. In certain cases, generation may not be able to complete successfully. For details, refer to the topic describing the selection of type versions during generation, page 409.

NOTE: To view the version of a type that is used by a facet, edit the facet encapsulating the type by using the **Global Templates Explorer**, select the element that represents the type, and click **Templatizer**. The version of the type is indicated in the **Global Types** section of the **Select Variables** window (for example, DEVCTL 0.01).

Validating Instance Identifiers

When you perform a template replacement operation, EcoStruxure Process Expert verifies if the result of the operation satisfies the uniqueness requirement for instance identifiers. If it does not, it displays the **Replace Template Summary** dialog box, which indicates which instance creates a duplicate identifier, and cancels the replace operation.

This can be the case when either of the following applies:

- Two instances use a different template but have the same identifier.
- You replace one of the two templates with the same template that the other instance uses.

For example:

- The application contains two instances:
 - *Valve_1*, which uses template *\$HandValve*
 - *Valve_1*, which uses template *\$Valve*
- If you replace template *\$HandValve* by template *\$Valve* (any version), the two instances use the same template and have the same identifier. This is not allowed.

NOTE: When replacing a template creates an element with the same identifier as the element of an existing instance, you can still proceed with the template replacement but the **Data** status of the instance is set to **Invalid**.

For example:

- The application contains two instances:
 - *Pump_1*, which uses template *\$Motor*
 - *Pump_1*, which uses template *\$HandValve*
- If you replace template *\$HandValve* by template *\$Valve* (any version), although the two instances do not use the same template, they have the element *Logic* in common, which has the same identifier (*Pump_1_DEVCTL*) in both instances because both instances have the same identifier. This is not allowed.

Updating the Template of One Instance

To update the template that is used by one instance, proceed as follows.

Step	Action
1	<p>Right-click the instance that you want to update and select Update Template.</p> <p>Result: The Update Template dialog box opens. It shows the latest version of the template that is available for the update.</p>
2	<p>Click OK.</p> <p>Result:</p> <ul style="list-style-type: none"> • The template that is used by the selected instance is updated with the latest version of the same template. • The version of the template that is used by the instance after the update appears in the Version column. • The result of the update process appears in the notification panel. <p>NOTE: Click Cancel to close the dialog box without updating the template.</p>

NOTE: To revert to an earlier version of a template, use the **Replace Template** command.

Updating the Templates of Several Instances

To update the templates of several instances of the application at once, proceed as follows.

Step	Action
1	<p>Right-click the root folder of the application and select Update Template.</p> <p>Result: The Update Template dialog box opens, which:</p> <ul style="list-style-type: none"> • Lists the templates, which are used by instances of the application for which a template of a later version is available. It indicates the current version of the template and the latest version that is available for the update. • Indicates, for each template, the instances that are using the template with an indication of the location of the instances (Path).
2	<p>Select the instances that you want to update:</p> <ul style="list-style-type: none"> • Instance by instance. • By template. • By using a combination of both.
3	<p>Click OK.</p> <p>Result:</p> <ul style="list-style-type: none"> • The update of the template that is used by the selected instances is performed. • The version of the template that is used by each instance after the update appears in the Version column. • The result of the update process appears in the notification panel. <p>NOTE: Click Cancel to close the dialog without updating the templates.</p>

NOTE: If you use a filter in the **Name** column of the **Update Template** dialog box and as a result, not all the instances that you had selected are displayed, the update operation is still performed on the entire selection of instances.

Replacing Templates of Instances

To replace the template that is used by one or more instances, proceed as follows.

Step	Action
1	<p>Select one or more instances and/or one or more folders.</p> <p>Result: The selected items are highlighted.</p>
2	<p>Right-click a highlighted item and select Replace Template.</p> <p>Result: The Replace Template dialog box opens.</p>
3	<p>Click the Select Template list.</p> <p>Result: The list of templates that are available for the replace operation is displayed.</p>
4	<p>Select the template that you want to use by scrolling or entering a keyword in the field of the menu.</p> <p>To search by entering the first letter of the template identifier, enter also the \$ prefix for Schneider Electric templates (for example, \$m).</p>
5	<p>Click OK.</p> <p>Result: If the result of the replacement operation does not satisfy the identifier uniqueness requirement, the Replace Template Summary dialog box opens and the replace operation is canceled.</p> <p>If no conflict is detected:</p> <ul style="list-style-type: none"> • The dialog box closes and the template that is used by the instances is replaced with the template that you selected. • The identifier and the version of the template used by the instances appears in the Template and Version columns respectively. • The result of the replace process appears in the notification panel. <p>NOTE: Click Cancel to close the dialog without replacing the template.</p>

NOTE: If you use a filter in the **Instance** column of the **Replace Template** dialog box and as a result, not all the instances that you had selected are displayed, the replace operation is still performed on the entire selection of instances.

Exporting and Importing Application Objects to/from CSV and XML Formats

Overview

The export and import functionality lets you create comma-separated value (CSV) or XML files and import them into the application of the same or another system.

This allows you to do the following:

- Move or copy the entire application or a selection of folders, *asset workspaces* (CSV only), *page 230*, and instances to another system.
- Perform bulk processes such as creating, modifying and/or deleting instances, folders, *asset workspaces* (CSV only), and interface links by editing the exported file.

This topic describes how to export objects, edit the export file to perform the required changes, and import it.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Exporting Folders and Instances to CSV and XML Formats

Overview

You can export application folders, instances, and interface links by using the **Export** command of the **Application Explorer**.

The application objects and their respective configuration parameters are saved to an export file in comma-separated, page 236 value (CSV) or XML format.

For example, you can use the export file to import, page 254 the application into another system while modifying some parameters, which helps avoid having to create it from scratch. Also, you can use it to restore instances after deleting them or to restore them to an earlier state.

NOTE: You also need to export templates, page 1039 that are used by the instances that you are exporting if the same version of template is not present in the Global Templates Library of the target system at the time of import. Instances whose template is not present cannot be imported.

Good Practices

- If you intend to edit the export file, try to limit your export selection to the objects that you want to modify or delete, for example, by using filters, page 120 in the **Application Browser**. It reduces the steps to manage unwanted objects during import and speeds up the process.
- If you want to create instances, export, at least, one instance that uses the same template. If you want to create interface links, export, at least, two instances that are linked. This lets you use the data as model. For example, you can copy and paste it and modify only necessary location, element, and/or parameter values.

Export Format Selection

The CSV, page 231 and XML, page 244 export files contain similar information and let you perform the same actions.

Data in CSV format can appear more comprehensible and easier to manage by using a spreadsheet editor because of its user-friendly, tabular presentation. Also, it facilitates the integration of existing data, such as an asset list, depending on their format.

Exporting to XML may be preferred by users who are familiar with the use of structured data and more powerful XML editors.

NOTE: You cannot update existing interface links by using the XML format.

Invariant Culture

EcoStruxure Process Expert uses the invariant culture, page 236 to format parameter values independently of the language setting that is used on the computer on which you create the export file.

For example, a decimal value that appears as 5,5 (comma separator) in the **Instance Editor** because of your regional settings appears as 5.5 (dot separator) in the export file.

Software Version Compatibility

You can import into a system instances that were exported from an earlier version of EcoStruxure Process Expert.

Some restrictions apply, page 254.

Object Selection

You can select the following objects to export them:

- One or several instances: Exports the data pertaining to the selected instances and their location in the folder structure of the application. You can select individual instances located in various folders and export them at once.
- One or several application folders: Exports the data pertaining to the selected folders, the instances contained in the folders, and in any subfolders.
- The root folder of the application: Exports any application folders and instances of the system.

An interface link is exported only if both linked instances are exported.

NOTE: When you select the **Export** command for a folder, this folder, any subfolders, and any instances they contain are exported. This is the case even if you initially selected, in addition to the folder, only one of the instances or subfolders it contains with the intention to export only the selected objects.

Aborting Export Tasks

After confirming the export by clicking **Save** in the **Save As** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The export file is not created.

Only the user who selected the command is allowed to cancel it.

NOTE: If you are exporting only a few objects, the task may complete before you are able to click the icon.

Creating the Export File in CSV or XML Format

To export application objects, proceed as follows.

Step	Action
1	In the Application Browser , select the objects that you want to export. Use filters and sorting, page 120 to facilitate the selection.
2	Right-click your selection and select Export . Result: The Export dialog box opens.
3	Select the location, file name, and format for the export file.
4	Click Save . Result: <ul style="list-style-type: none"> • The export file containing the selected objects is created. • A summary of the exported objects is displayed in the notification pane.

NOTE: If any object that you have selected for export is open in an editor (for example, the **Instance Editor**), accessed for an assignment, update, or replace operation, or if its **Properties** tab is open, you cannot create the export file.

Exporting Asset Workspaces to CSV Format

Overview

You can export asset workspaces and the instances that belong to them by using the **Export** command.

The application objects and their respective configuration parameters are saved to an export file in comma-separated value (CSV), page 236 format.

NOTE: You also need to export templates, page 1039 that are used by the instances that you are exporting if the same version of template is not present in the Global Templates Library of the target system at the time of import. Instances whose template is not present cannot be imported.

Object Selection

You can select the following objects to export asset workspaces:

- The root node of the **Asset Workspace** pane: Exports the existing asset workspaces and the instances that belong to them.
- One or several asset workspaces: Exports the workspaces and the instances that belong to them.
- The root folder of the application: Exports any application folders, asset workspaces, and instances of the system.

Creating the Export File Containing Asset Workspace Data

For information on how to create the export file, refer to the topic describing the export of folders and instances, page 228.

CSV Application Export File

Export File Description

When you export application objects to comma-separated, page 236 format (CSV), two files are created.

File	Default file name	Content and purpose
Application export file	<System identifier>	<p>Data of the folders, instances, interface links, and asset workspaces that you have exported.</p> <p>The data is organized in five types of groups.</p> <p>You need to edit the content, page 757 of this file to create, modify, or delete application objects.</p>
Application interface data supporting file	<System identifier>_Pending Interfaces where <i>Pending Interfaces</i> is appended to the file name once the file is created.	<p>Data on application interfaces that the exported instances expose and that are not connected. It includes interfaces that are disabled.</p> <p>The data is organized by instance name.</p> <p>It helps you edit data of the InterfaceLinks group in the application export file by providing the necessary data of each interface that you can connect.</p>

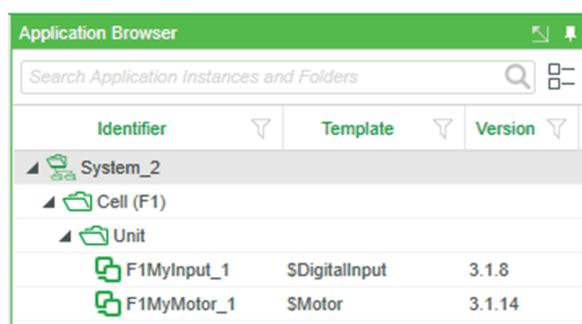
The following table describes the grouping of data in the application export file. Each group has a group header and a header row, which describes the data of objects.

Group header	Description
<i>Folders</i>	<p>Groups that contain folder data, page 236.</p> <p>More than one group exists when you export folders that have a different number of parameters as part of the optional folder properties.</p> <p>The number of columns depends on the folder configuration. A column with its header exists for each system parameter and each configured optional parameter.</p> <p>NOTE: A group header and its header row appear only if the export file contains folder data.</p>
<i>TemplateIdentifier=<Template ID> Version=<Value></i> where: <ul style="list-style-type: none"> <Template ID> is the identifier of the template that is used by the instance. <Value> is the template version. <p>Example:</p> <p><i>TemplateIdentifier=\$Motor Version=3.1.0</i></p>	<p>Groups that contain instance data, page 236.</p> <p>There are as many groups as templates and versions used by exported instances.</p> <p>The number of columns depends on the instance configuration. A column with its header exists for each element and parameter of the instance.</p> <p>NOTE: A group header and its header row appear only if the export file contains instance data.</p>
<i>InterfaceLinks</i>	<p>Group that contains the data of interface links, page 236, which link exported instances.</p> <p>The number of columns and their headers are fixed.</p> <p>NOTE: The group and its header row are always present.</p>

Group header	Description
<i>Workspaces</i>	Group that contains the data of the exported asset workspaces, page 236. NOTE: The group and its header row are always present.
<i>WorkspaceIdentifier=<Asset workspace identifier></i> Example: <i>WorkspaceIdentifier=AssetWorkspace_1</i>	Group that contains the data of instances that belong to an asset workspace. There are as many groups as exported asset workspaces. NOTE: The group and its header row are present even if the exported asset workspace contains no instance.

Example of CSV Export Files

The following example shows two instances, which have been linked by an interface link.

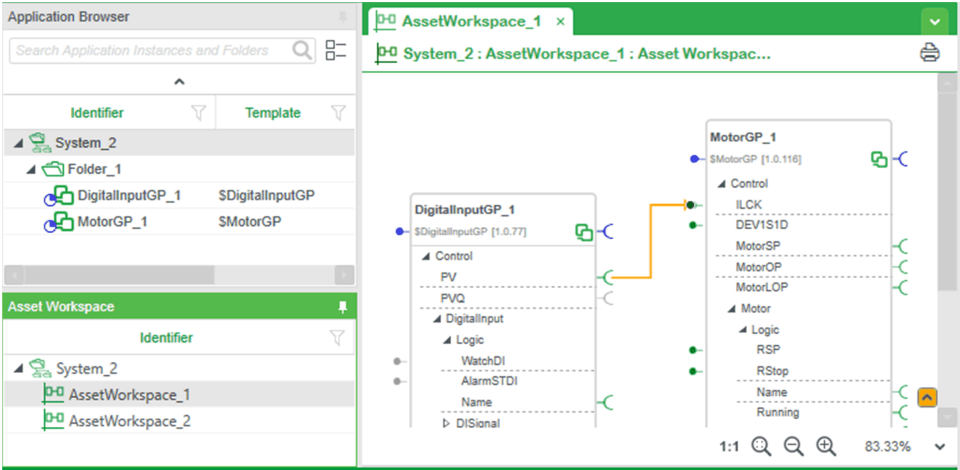


The following figure shows a partial view of the resulting export file when you export folder *Cell* to CSV format.

	A	B	C	D	E	F	G	H
1	Export Version=2019	Created on=05.25.2019 09:04	System=System_2	Encoding=UTF-8				
2	Folders							
3	\$Action	\$Location	\$Identifier	\$Alias	\$Description	\$Area		
4	Create	Root	Cell	F1				
5	Create	Root	Unit					
6								
7	TemplateIdentifier=SDigitalInput Version=3.1.8							
8	\$Action	\$InstanceName	\$Identifier(Non_Editable)	\$IsHierarchical	\$Location	\$Description	Control.Enabled(Non_Editable)	Parameter.D.SignalName
9	Create	MyInput_1	F1MyInput_1	TRUE	Root\Cell\Unit		TRUE	
10								
11	TemplateIdentifier=SMotor Version=3.1.14							
12	\$Action	\$InstanceName	\$Identifier(Non_Editable)	\$IsHierarchical	\$Location	\$Description	Control.Enabled(Non_Editable)	Parameter.Control.Running.Negative
13	Create	MyMotor_1	F1MyMotor_1	TRUE	Root\Cell\Unit		TRUE	FALSE
14								
15	InterfaceLinks							
16	\$Action	\$SourceLocation	\$SourceInstanceName	\$SourceInstance	\$SourceInterface	\$SourceInter	\$DestinationInstanceName	\$DestinationInstanceTemplateId
17	Create	Root\Folder_1	MyInput_1	SDigitalInput	Control	PV	MyMotor_1	SMotor

Item	Description
1	Export file information that is automatically added.
2	Group headers.
3	Header rows.
4	Element of the instance and its selection state.
5	Parameter of the instance and its configured value. Parameters appear right after their parent element.
6	Parameter of the instance for which no value is configured.
7	Data rows of exported objects (one row per object, such as a folder, an instance, or an interface link).

The following example shows two asset workspaces that exist in *System_2*. *AssetWorkspace_1* contains two linked instances and *AssetWorkspace_2* contains none.



The following figure shows the workspace related portion of the resulting export file when you export *System_2* to CSV format either from the root node of the **Application Browser** or the **Asset Workspace**.

	A	B	C	D	E	F
22						
23	:WorkSpaces					
24	:\$Action	\$Location	\$Identifier			
25	Create	Root	AssetWorkSpace_1			
26	Create	Root	AssetWorkSpace_2			
27						
28	:WorkspaceIdentifier=AssetWorkSpace_1					
29	:\$Action(Non_Editable)	\$InstanceLocation	\$InstanceName	\$InstanceIdentifier	\$TemplateIdentifier	\$Position(X,Y)
30	Create	Root\Folder_1	MotorGP_1	MotorGP_1	\$MotorGP	455,68
31	Create	Root\Folder_1	DigitalInputGP_1	DigitalInputGP_1	\$DigitalInputGP	125,153
32						
33	:WorkspaceIdentifier=AssetWorkSpace_2					
34	:\$Action(Non_Editable)	\$InstanceLocation	\$InstanceName	\$InstanceIdentifier	\$TemplateIdentifier	\$Position(X,Y)
35						
36						

Item	Description
1	Group headers.
2	Header rows.
3	Rows containing data of the exported asset workspaces.
4	Rows containing data of the instances that belong to each exported asset workspace. NOTE: The data of their interface links and configuration parameters appears in their respective groups in the same export file.

The following figure shows a partial view of a supporting file that is created in addition to the actual export file.

	A	B	C	D	E
1	:Interfaces				
2	:SLocation	\$InstanceName	\$InstanceTemplateIdentifier	\$InterfacePath	\$InterfaceIdentifier
3	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	NullInterfaceRoleA	NullInterfaceRoleA
4	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	NullInterfaceRoleB	NullInterfaceRoleB
5	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	Control	PVQ
6	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	Control\DInput\Logic	WatchDI
7	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	Control\DInput\Logic	DInputName
8	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	Control\DInput\Logic	AlarmSTDI
9	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	Control\DInput\DISignal	BoolVar
10	Root\Cell\Unit	F1MyInput_1	\$DigitalInput	Control\DInput\DISignal	DINPUTSignalExt
11	Root\Cell\Unit	F1MyMotor_1	\$Motor	NullInterfaceRoleA	NullInterfaceRoleA
12	Root\Cell\Unit	F1MyMotor_1	\$Motor	NullInterfaceRoleB	NullInterfaceRoleB
13	Root\Cell\Unit	F1MyMotor_1	\$Motor	Control	ILCK
14	Root\Cell\Unit	F1MyMotor_1	\$Motor	Control	MotorOP
15	Root\Cell\Unit	F1MyMotor_1	\$Motor	Control	Dev1S1D
16	Root\Cell\Unit	F1MyMotor_1	\$Motor	Control\Motor\Logic	ZSHPOS
17	Root\Cell\Unit	F1MyMotor_1	\$Motor	Control\Motor\Logic	ZSLPOS
18	Root\Cell\Unit	F1MyMotor_1	\$Motor	Control\Motor\Logic	DEVCTLName
19	Root\Cell\Unit	F1MyMotor_1	\$Motor	Control\Running	BoolVar

Editing the CSV Application Export File

Overview

By editing the application export file (.csv), you can create, update, and/or delete application objects.

Various rules and requirements apply when you open and edit CSV export files. They are described in this topic.

You can use the supporting file (**_PendingInterfaces.csv*) to help with the editing of interface link data.

EcoStruxure Process Expert applies the changes when you import the edited CSV file, page 254 into the application of the system. At that time, you have the possibility to review and modify the list of imported objects (except interface links).

Good Practices

- Before editing an application export file, create a copy of it. If needed, it lets you restore the original application.
- Edit only an up-to-date export file to help avoid data conflicts when importing.
- Carefully remove from the file objects that are not impacted by the import, for example, instances that you are not updating or deleting.
- When you use data from the supporting file to create or update interface links, cut the data rather than copying it to help avoid connecting the same interface several times.

The other way around, add to the supporting file data of interfaces that you are disconnecting.

Opening the CSV File by Using a Spreadsheet Editor

When you open a CSV export file by using a spreadsheet editor, some instance parameter values (for example, durations with millisecond values, certain special characters, or strings starting with - (hyphen)) may get modified by the editor.

So that these values remain unchanged, you first need to import the data into a blank workbook in *Unicode (UTF-8)* format.

Follow this procedure. Microsoft® Excel® is used as an example. Menu items and dialog box names may vary depending on the version.

Step	Action
1	Open a blank workbook.
2	In the Data menu, select to import from text or CSV.
3	In the Import Text File dialog box, select the exported CSV file.
4	Select the following attribute values for the import of data: <ul style="list-style-type: none"> • File origin: <i>65001 : Unicode (UTF-8)</i> • List separator (delimiter): <i>Comma</i> • Format for all columns (data type): <i>Text⁽¹⁾</i> • <i>Use First Row as Header⁽¹⁾</i> (depending on version): Open the menu and select this setting even if it is already selected by default.
5	Load the values to the blank workbook. <p>NOTE: After loading is completed, the first row of the CSV file must start with <i>;Export Version=<Software Version></i>. You may see additional text in columns E and higher of the first row (for example, <i>Column1, _1</i>). This text is treated as comment, page 242 and ignored during import.</p>

Step	Action																																																															
	<p>The following example shows an extract of a CSV export file after selecting <i>Use First Row as Header</i> and loading values to the blank workbook.</p> <table><tr><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr><tr><td>1</td><td>:Export Version=2019</td><td>Created on=05:25:2019 09:04</td><td>System=System_2</td><td>Encoding=UTF-8</td><td>Column1</td><td>_1</td></tr><tr><td>2</td><td>:Folders</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>3</td><td>:Action</td><td>\$Location</td><td>\$Identifier</td><td>\$Alias</td><td>\$Description</td><td>\$Area</td></tr><tr><td>4</td><td>Create</td><td>Root</td><td>Cell</td><td>F1</td><td></td><td></td></tr><tr><td>5</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>6</td><td>:TemplateIdentifier=SDigitalInput Version=3.1</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>7</td><td>:Action</td><td>\$InstanceName</td><td>\$Identifier(Non_Editable)</td><td>\$IsHierarchical</td><td>\$Location</td><td>\$Description</td></tr><tr><td>8</td><td>Create</td><td>MyInput_1</td><td>F1MyInput_1</td><td>TRUE</td><td>Root\Cell\Unit</td><td></td></tr></table>		A	B	C	D	E	F	1	:Export Version=2019	Created on=05:25:2019 09:04	System=System_2	Encoding=UTF-8	Column1	_1	2	:Folders						3	:Action	\$Location	\$Identifier	\$Alias	\$Description	\$Area	4	Create	Root	Cell	F1			5							6	:TemplateIdentifier=SDigitalInput Version=3.1						7	:Action	\$InstanceName	\$Identifier(Non_Editable)	\$IsHierarchical	\$Location	\$Description	8	Create	MyInput_1	F1MyInput_1	TRUE	Root\Cell\Unit	
	A	B	C	D	E	F																																																										
1	:Export Version=2019	Created on=05:25:2019 09:04	System=System_2	Encoding=UTF-8	Column1	_1																																																										
2	:Folders																																																															
3	:Action	\$Location	\$Identifier	\$Alias	\$Description	\$Area																																																										
4	Create	Root	Cell	F1																																																												
5																																																																
6	:TemplateIdentifier=SDigitalInput Version=3.1																																																															
7	:Action	\$InstanceName	\$Identifier(Non_Editable)	\$IsHierarchical	\$Location	\$Description																																																										
8	Create	MyInput_1	F1MyInput_1	TRUE	Root\Cell\Unit																																																											
6	Save the workbook in <i>CSV UTF-8 (comma delimited)</i> format (keep saving in this format when you edit the file).																																																															

(1) You may need to edit the file during the import process to access the attribute.

Setting the List Separator Format

CSV export files use the comma as list separator (delimiter) independently of the setting that is in effect on the computer on which the file is created.

If the format of the list separator on the computer is different from comma, data will not be displayed correctly when you open the export file by using a spreadsheet editor.

To verify the setting and change it to comma if necessary, open the Windows **Control Panel** and click **Clock and Region > Region**. In the **Formats** tab, click **Additional settings....**

NOTE: For information on how to use commas in values of string data types in the CSV file, refer to the topic describing [editing rules](#), page 242.

Using the Invariant Culture

Use the invariant culture to modify or add parameter values in the application export file.

If required, during import EcoStruxure Process Expert converts the format according to the language setting that is used on the target computer.

The table describes the formats of the invariant culture that are used in the export file.

Type	Format	Example
Decimal separator	x.y	2.5
Time	hh:mm:ss	14:30:00
Date	mm/dd/yyyy	11/20/2014
Date Time	mm/dd/yyyy hh:mm:ss	11/20/2014 14:30:00
Duration	DD.hh:mm:ss.ms	10.12:30:10.15 translating into 10 days 12 hours 30 min 10 sec 15 msec

Application Export File Content Requirements

This topic describes the columns that appear in the each type of group and the requirements for each one. Use this information when you edit existing data, add columns, and/or add entire groups. For details on the format of the group headers, refer to the topic describing the [export file](#), page 231.

Folders group, page 231. The information applies to each group that appears in the file or that you add.

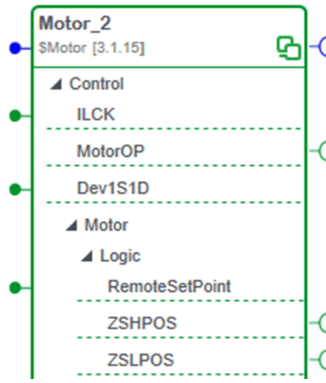

Header	Description	Requirement
\$Action	Action that is performed on the folder during import.	Possible values (not case-sensitive): <ul style="list-style-type: none"> • Create (default) • Update • Delete • Blank: <ul style="list-style-type: none"> ◦ Creates the folder if it does not exist yet. ◦ Updates the folder if it exists.
\$Location	Path to the folder in the application starting from the system root folder. Example: <i>Root\Site\Area\Unit_A</i> NOTE: The name of the system root folder, page 160 is replaced by <i>Root</i> .	A column with this header and a valid value in each row are required. The path must exist in the application or be created by the import. For folders whose parent is the system root folder, enter <i>Root</i> .
\$Identifier	Folder identifier. NOTE: You cannot update the identifier of a folder through import.	A column with this header and a valid value in each row are required.
\$Alias	Folder alias, page 172.	Column and value are optional. If a row contains data, it must be valid.
\$Description	Folder description.	
\$Area	Folder area, page 172. NOTE: If the value is inherited from a parent folder, page 173, it does not appear in the export file.	
Parameter x.\$Identifier	Identifier of the folder parameter, page 172.	Column and value are optional. If a row contains data, it must be valid. NOTE: The numbering of folder parameters (x) corresponds to the order (top to bottom) in which parameters appear in the properties of the folder when no sorting is applied.
Parameter x.\$Description	Description of the folder parameter.	
Parameter x.\$Value	Value of the folder parameter.	



TemplateIdentifier group, page 231 for instances. The information applies to each group that appears in the file or that you add.

Header	Description	Requirement
\$Action	Action that is performed on the instance during import.	Possible values (not case-sensitive): <ul style="list-style-type: none"> • Create⁽¹⁾ (default) • Update • Delete • Blank: <ul style="list-style-type: none"> ◦ Creates the instance if it does not exist yet. ◦ Updates the instance if it exists.
\$InstanceName	Instance name, page 191.	A column with this header and a valid value in each row are required.
\$Identifier (Non_Editable)	Instance identifier, page 177. The value is used to identify the instance or is automatically calculated in the case of an instance creation.	A column with this header is required and: <ul style="list-style-type: none"> • To create an instance, the value can be left blank. • To update or delete an instance, the value must be valid.

Header	Description	Requirement
\$IsHierarchical	Hierarchical Name function, page 189.	Column and value are optional. Possible values: <ul style="list-style-type: none"> • True • False • Blank (or not present): <ul style="list-style-type: none"> ◦ Equals true for an instance that is created. ◦ Keeps the configured value for an existing instance.
\$Location	Path to the instance in the application. Example: <i>Root\Site\Area\Unit_A</i> NOTE: The name of the system root folder, page 160 is replaced by <i>Root</i> .	A column with this header and a valid value in each row are required. The path must exist in the application or be created by the import.
\$Description	Instance description.	Column and value are optional.
\$Area	Instance area, page 191. NOTE: If the value is inherited from a parent folder, page 173, it does not appear in the export file.	If a row contains data, it must be valid.
<Element path>.Enabled	A column exists for each service (element, page 232) of the instance as they appear in the Instance Editor . The instance identifier is excluded from the element path. Example: For the <i>Running</i> service that belongs to the <i>Control</i> element, the column header is: <i>Control.Running.Enabled</i>	Column and value are optional. Possible values (not case-sensitive): <ul style="list-style-type: none"> • True: The service is enabled. • False: The service is disabled. • Blank: <ul style="list-style-type: none"> ◦ Equals false for an instance that is created. ◦ Keeps the configured value for an existing instance. Mandatory elements (whose check box cannot be cleared in the Instance Editor) are identified with (<i>Non-Editable</i>) in the header. If you set the value to false, it is not considered during import and the element remains enabled.
Parameter.<parameter full path>	A column exists for each parameter, page 232 that is featured by the instance, including parameters of disabled services (elements), page 242 and not configured parameters. Example: For the <i>DISignalName</i> parameter of the <i>Running</i> service, the column header is: <i>Parameter.Control.Running.DISignalName</i>	Column and value are optional. If a row contains data, it must be valid. When you edit a parameter, verify that the element, page 232 to which it belongs is set to true; otherwise, the parameter value is not applied during import. This rule does not apply to parameter values of mandatory elements, which are applied even if the element is set to false in the CSV file. NOTE: To view the hierarchy of elements and parameters of an instance, use the Instance Editor or the View Instance window, page 211. NOTE: For parameters of the enumerated data type, page 248, the name is exported as value. For example, if the parameter value, page 183 that is configured in the Instance Editor is <i>Operator</i> (2), the exported value is <i>Operator</i> . To edit the value, you can enter either the name of the enumerator or the associated integer (for example, 3 for an enumerator that is <i>Supervisor</i> (3)).
(1) Instances with action set to create and that you do not want to import can be unselected during import, page 258.		

InterfaceLinks group, page 231.

Parameter	Description	Requirement
\$Action	Action that is performed on the interface link during import.	Possible values (not case-sensitive): <ul style="list-style-type: none"> Create (default) Update⁽¹⁾ Delete Blank: <ul style="list-style-type: none"> Creates the link if it does not exist yet. Updates⁽¹⁾ the link destination if the link exists.
\$SourceLocation	Path to the instance that is the source of the link in the Application Explorer . The name of the system root folder, page 160 is replaced by <i>Root</i> . Example: For instance <i>Motor_1</i> of <i>System_1</i> that is located in <i>Folder_2</i> , which is a subfolder of <i>Folder_1</i> , the value is <i>Root\Folder_1\Folder_2</i> .	A column with this header and a valid value in each row are required.
\$SourceInstanceName	Value of the \$Name parameter, page 191 of the instance that is the source of the link.	
\$SourceInstanceTemplateIdentifier	Identifier of the template that is used by the instance, which is the source of the link.	
\$SourceInterfacePath	Path to the source interface in the composition of the instance. Example: In the following sample instance, the value for interface <i>ZSHPOS</i> is <i>Control\MotorLogic</i> . 	A column with this header and a valid value in each row are required. NOTE: For RTNS interfaces, the column is optional and does not contain values.
\$SourceInterfaceIdentifier	Identifier of the interface that is the source of the link. 	A column with this header and a valid value in each row are required. NOTE: Refer to the description of RTNS interfaces for information on their specific identifiers, page 208.

Parameter	Description	Requirement
\$DestinationLocation	Path to the instance that is the destination of the link in the Application Explorer . The name of the system root folder, page 160 is replaced by <i>Root</i> .	A column with this header and a valid value in each row are required.
\$DestinationInstance Name	Value of the \$Name parameter, page 191 of the instance that is the destination of the link.	
\$DestinationInstance TemplateIdentifier	Identifier of the template that is used by the instance, which is the destination of the link.	
\$DestinationInterface Path	Path to the destination interface in the composition of the instance. Example: In the following sample instance, the value for interface <i>BoolVar</i> is empty (blank) because it has no parent element. 	A column with this header and a valid value in each row are required. NOTE: For RTNS interfaces, the column is optional and does not contain values.
\$DestinationInterface Identifier	Identifier of the interface that is the destination of the link. 	A column with this header and a valid value in each row are required. NOTE: Refer to the description of RTNS interfaces for information on their specific identifiers, page 208.
\$NewDestinationLocation	The description is the same as for the <i>\$Destination*</i> parameters but the data applies to the new destination interface of the same or another instance. The parameters are empty by default.	Columns with these headers are required. Rows can be left blank.
\$NewDestinationInstance Name		The columns must contain a valid value only when you change the destination, page 243 of an existing link.
\$NewDestinationInstance TemplateIdentifier		
\$NewDestinationInterface Path		
\$NewDestinationInterface Identifier		
(1) You can only change the destination, page 243 of an existing link by using the <i>Update</i> action.		

Workspaces group, page 231.

Header	Description	Requirement
\$Action	Action that is performed on the asset workspace during import.	Possible values (not case-sensitive): <ul style="list-style-type: none"> Create (default): Creates the asset workspace or updates an existing one based on the information in the <i>WorkspaceIdentifier</i>=<Asset workspace name> group. Delete: Deletes the workspace only. Does not delete from the application the instances that belong to it nor their interface links. Blank: Same as Create.
\$Location	Path to the asset workspace in the Asset Workspace pane. <i>Root</i> by default.	A column with this header and <i>Root</i> as value in each row are required. Asset workspaces can only be located at the system root node level.
\$Identifier	Asset workspace identifier. NOTE: You cannot update the identifier of an asset workspace through import.	A column with this header and a valid value in each row are required.

WorkspaceIdentifier group, page 231. The information applies to each group that appears in the file or that you add.

Header	Description	Requirement
\$Action	Action that is performed on the instance in the context of the asset workspace during import.	Not editable. To remove an instance from an asset workspace, delete the row. To add an instance, add a row and enter the required information in each column.
\$InstanceLocation	Path to the instance in the application.	A column with this header and the same value as in the <i>\$Location</i> column of the <i>TemplateIdentifier</i> group for this instance are required.
\$InstanceName	Name of the instance.	A column with this header and the same value as in the <i>\$InstanceName</i> column of the <i>TemplateIdentifier</i> group for this instance are required.
\$InstanceIdentifier	Identifier of the instance.	A column with this header and the same value as in the <i>\$Identifier</i> column of the <i>TemplateIdentifier</i> group for this instance are required.
\$TemplateIdentifier	Identifier of the template that is used by the instance.	A column with this header and the same template name as in the <i>TemplateIdentifier</i> group header for this instance are required. For example, <i>\$AnalogInputGP</i> if the header for this instance is : <i>TemplateIdentifier=\$AnalogInputGP</i> <i>Version=1.0.131</i> .
\$Position	Position in pixels of the top left corner of the instance representation in the asset workspace .	Two comma-separated values indicating the position on the X and Y axis respectively. For example, 0,0 positions the instance along the left and top borders of the asset workspace.

Editing Rules

The table indicates the rules that apply when you edit a CSV application export file.

Item	Rule	
Header of a group	Must start with a colon.	
First header of a header row	You can remove a group header and its header row if the group contains no data. You can change the order of groups.	
Headers of non-editable data	Headers of data that does not need to be edited or that is ignored during import are indicated by the <i>Non_Editable</i> suffix.	
Rows of object data	Refer to the requirements, page 236. You can delete the row of an object if you do not want to import it or modify it through import.	
Empty rows	Can be added anywhere.	
Comment rows	Can be added anywhere and must start with a semi-colon. Comments are ignored during import.	
Order of columns	Can be changed within a group given the first header of the row starts with a colon.	
Empty columns	Can be added after the first column.	
Interface links	For any actions on links, page 243, the data of the linked instances must be included in the application export file even if the instances already exist in the target application.	
List separator	Only comma, page 236 is accepted.	
Values	Use the invariant culture, page 236. To use commas or quotation marks in values of string data types when you edit the file by using a text editor, for example, Notepad, follow these rules:	
	Expected value after import	Value to be entered
	Motor, Pump	"Motor, Pump"
	Motor"Pump	"Motor""Pump"

Creating, Updating, and Deleting Folders

To create an instance in a new folder, you must also enter in the CSV file the necessary data to create the folder.

Since each row corresponds to only one folder, to create, for example, the structure *Root\Folder_1\Folder_10*, you must configure two rows.

NOTE: It is not possible to rename a folder by using the *Update* action.

Perform the following steps in the *Folders* groups of the CSV application export file.

Step	Action
1	Edit the rows that contain data of folders and/or create rows. Verify that at least the required columns and data, page 236 are present.
2	Specify a value for <i>\$Action</i> for each row as needed.

Creating, Updating, and Deleting Instances

Perform the following steps in the *TemplateIdentifier* groups of the CSV application export file.

Step	Action
1	Edit the rows that contain data of instances and/or create new groups and/or rows. Verify that at least the required columns and data, page 236 are present.
2	Verify that the data for <i>\$Location</i> is consistent with the target application and/or <i>\$Location</i> in the <i>Folders</i> group.
3	Specify a value for <i>\$Action</i> for each row as needed.

NOTE: If you update data of instances that are already linked and want the link to be maintained, it is not required to update the row of the *InterfaceLinks* group that pertains to the link as long as the linked interfaces remain enabled. This applies even if you change the value of *\$InstanceName* or the instance identifier (for example, by configuring an alias for a folder) for either or both instances.

Creating and Deleting Interface Links

Perform the following steps in the CSV application export file.

Step	Action
1	In the <i>InterfaceLinks</i> group, edit the rows that contain data of the links and/or create new rows. Verify that at least the required columns and data, page 236 are present.
2	Verify that the data for <i>\$SourceLocation</i> and <i>\$DestinationLocation</i> is consistent with the target application and/or <i>\$Location</i> in the other groups (if applicable).
3	Specify a value for <i>\$Action</i> for each row as needed.
4	<p>In the <i>TemplateIdentifier</i> group, verify that the data of the linked instances is entered.</p> <p>If the linked instances already exist in the target application with the same configuration and you do not want to modify them, modify the value of <i>\$Action</i> to <i>Update</i> or leave it blank for each instance.</p> <p>NOTE: If the instances already exist in the target application and you leave the value as <i>Create</i>, the Resolve Uniqueness Conflicts dialog box, page 260 opens during import and you need to select the <i>Update</i> action for the instances there.</p>

Changing the Destination of An Existing Interface Link

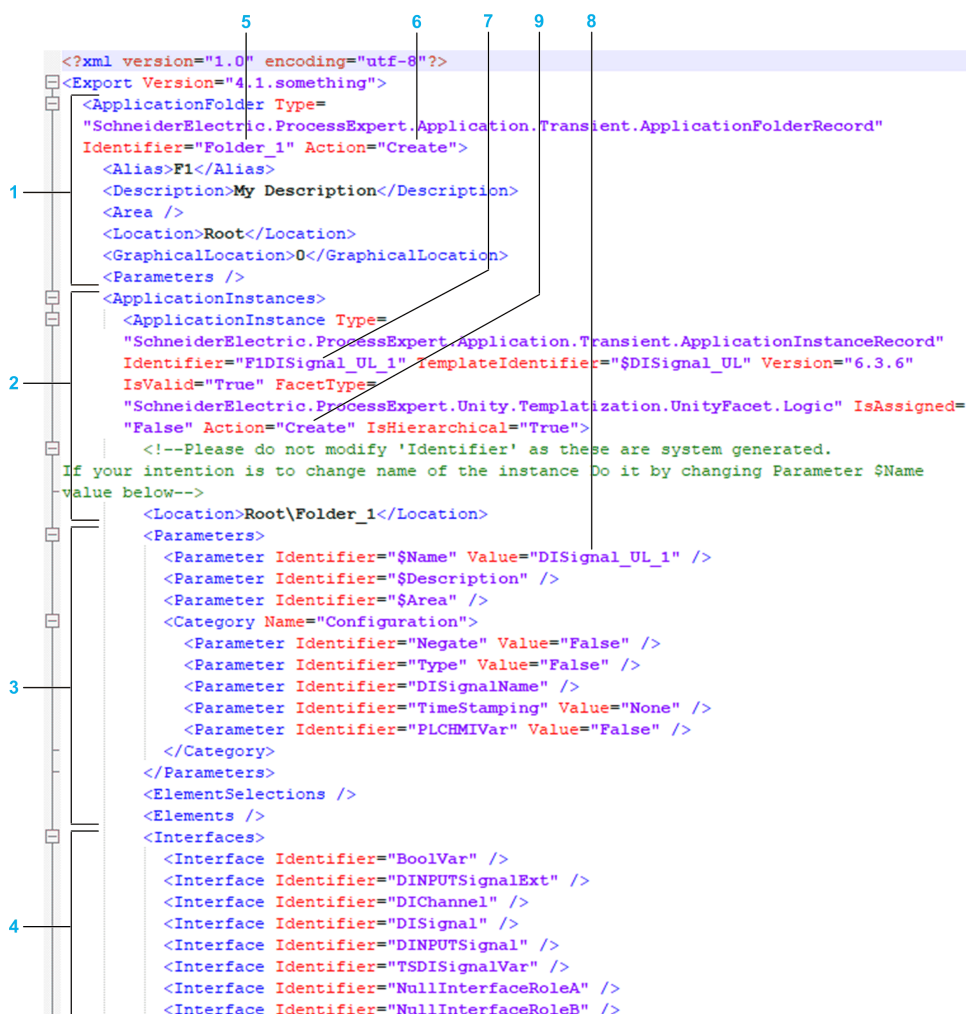
To move the destination of an existing link to a different interface, edit the CSV application export file as follows. The new destination interface must be available for connection during import.

Step	Action
1	In the <i>InterfaceLinks</i> group, for the row that contains the data of the existing link, modify the value of <i>\$Action</i> to <i>Update</i> or leave it blank.
2	<p>Enter the appropriate data in each of the five columns <i>\$NewDestinationLocation</i> to <i>\$NewDestinationInterfaceIdentifier</i>.</p> <p>NOTE: You do not need to edit the data of the existing interface link. If the interface link does not exist in the target application, a link from the specified source to the new destination is created anyway.</p>
3	<p>In the <i>TemplateIdentifier</i> group, verify that the data of the linked instances is included.</p> <p>If the instances already exist in the target application with the same configuration and you do not want to modify them, modify the value of <i>\$Action</i> to <i>Update</i> or leave it blank for each instance.</p> <p>NOTE: If the instances already exist in the target application and you leave the value as <i>Create</i>, the Resolve Uniqueness Conflicts dialog box, page 260 opens during import and you need to set the action to <i>Update</i> there.</p>

XML Application Export File


Export File Description

The following example shows the structure of an XML application export file (partial view) that was created by exporting *Folder_1*, which contains instance *DISignal_UL_1*.



Item	Description
1	Data pertaining to the folder.
2	Data pertaining to the instance.
3	Data of the parameters and elements of the instance.
4	Data of the application interfaces, page 246 that the instance exposes.
5	Folder identifier.
6	Folder <i>Action</i> parameter.
7	<i>\$InstanceID</i> parameter of the instance that is automatically generated. This is the identifier that is used to identify the instance for delete and update actions. Do not edit this parameter.
8	<i>\$Name</i> parameter of the instance.
9	Instance <i>Action</i> parameter.

When instances that are linked are exported, the information of the interface links is grouped in the *InterfaceLinks* element, which appears at the end of the file. The following example shows an extract of an export file, which contains two instances linked by two interface links.



```
<InterfaceLinks Type="SchneiderElectric.ProcessExpert.Application.Source.Model.InterfaceLink">
  <InterfaceLink Action="Create" DestinationPath="Root\Folder_1" DestinationInstanceName="DISignal_UL_1"
    DestinationInstanceTemplateIdentifier="$DISignal_UL" DestinationInstanceIdentifier="DISignal_UL_1"
    DestinationInterfacePath="" DestinationInterfaceIdentifier="DISignal"
    IsDestinationInstanceHierarchical="True" SourcePath="Root\Folder_1" SourceInstanceName="Motor_1"
    SourceInstanceTemplateIdentifier="$Motor" SourceInstanceIdentifier="Motor_1" SourceInterfacePath=
    "Control\Motor\Logic" SourceInterfaceIdentifier="RSP" IsSourceInstanceHierarchical="True" />
  <InterfaceLink Action="Create" DestinationPath="Root\Folder_1" DestinationInstanceName="DISignal_UL_1"
    DestinationInstanceTemplateIdentifier="$DISignal_UL" DestinationInstanceIdentifier="DISignal_UL_1"
    DestinationInterfacePath="" DestinationInterfaceIdentifier="NullInterfaceRoleA"
    IsDestinationInstanceHierarchical="True" SourcePath="Root\Folder_1" SourceInstanceName="Motor_1"
    SourceInstanceTemplateIdentifier="$Motor" SourceInstanceIdentifier="Motor_1" SourceInterfacePath=""
    SourceInterfaceIdentifier="NullInterfaceRoleB" IsSourceInstanceHierarchical="True" />
</InterfaceLinks>
```

Item	Description
1	Action parameter plus source and destination information pertaining to the first interface link.
2	Same information pertaining to the second interface link.

Editing the XML Application Export File to Manage Folders and Instances

Overview

You can edit the XML export file by using any XML editing software to perform one or more of the following actions on application folders and/or instances:

- Create an object by using the parameter values of the export file.
- Update an object that has the same identifier by using the parameter values in the export file.
- Delete an object that has the same identifier as in the export file.

You can also create and delete interface links, [page 251](#).

EcoStruxure Process Expert applies the changes when you import the edited XML file, [page 254](#) into the application of the system. At that time, you have the possibility to review and modify the list of imported objects (except interface links).

NOTE: You cannot update the identifier of a folder through import.

Good Practices

- Before editing an application export file, create a copy of it. If needed, it lets you restore the original application.
- Edit only an up-to-date export file to help avoid data conflicts when importing.

Application Export File Content

The table shows the objects and their respective parameters that are exported to the XML file and that are configurable.

Object	Parameters
Application folder	For the folder and any subfolders
	<i>Action</i>
	<i>Identifier</i>
	<i>Alias</i>
	<i>Description</i>
	<i>Area</i> ⁽¹⁾
	<i>Location</i>
	<i>Parameters</i>
Instance	For the instances in the folder and its subfolders
	<i>Action</i>
	<i>TemplateIdentifier</i>
	<i>Version</i>
	<i>IsHierarchical</i>
	<i>Location</i>
	<i>\$Name</i>
	<i>\$Description</i>
	<i>\$Area</i> ⁽¹⁾
	Elements that are enabled (selected in the Instance Editor)
	Parameters of each element
	For parameters of the enumerated data type, page 248, the name is exported as value. For example, if the parameter value, page 183 that is configured in the Instance Editor is <i>Operator (2)</i> , the exported value is <i>Operator</i> . To edit the value, you can enter either the name of the enumerator or the associated integer (for example, 3 for an enumerator that is <i>Supervisor (3)</i>).
	Identifier of exposed interfaces and their path in the composition of the instance. Includes data of disabled and connected interfaces. When an alias is configured for an interface, both the alias and the identifier are indicated. The alias is the name that appears in the Asset Workspace Editor or Links Editor .
Interface link	Parameters of interface links that exist between exported source and destination instances. Refer to the topic describing how to edit the XML file to manage interface links, page 251.
(1) If the value is inherited from a parent folder, page 173, it does not appear in the export file.	

Action Parameter

Configuring the *Action* parameter is optional.

Possible values (not case-sensitive):

- Create (default)
- Update (except for interface links, page 251)
- Delete
- Blank:
 - Creates the object if it does not exist yet.
 - Updates the object if it exists (except interface links).

NOTE: Instances with action set to create and that you do not want to import can be removed from the list of imported objects during import, page 258.

Using the Invariant Culture

Use the invariant culture, page 236 to modify or add parameter values in the application export file.

Editing Rules for XML Application Export Files

To maintain the consistency of the folder structure and instances contained in the export file, follow these rules.

Item	Rule
Instance	Do not modify Identifier of the instance (<i>\$InstanceID</i> parameter) because it is automatically generated.
Parameters	Modify only the parameters described in the procedures that appear in this topic.
Object hierarchy	Do not modify the structure of child elements. For example, a folder must be the parent of an instance and not the opposite. However, you can modify the order of <i>Elements</i> or <i>Parameters</i> inside an instance (<i>ApplicationInstance</i> section).
	When the export file contains data of a folder structure, do not change the <i><Location></i> parameter of the highest level folder (the folder from which the export file was created). For subfolders, the hierarchy from the export file prevails.
	When the export file contains data of a single instance or single folder, do not modify the <i><Location></i> parameter.

The following figure shows an extract of an export file of instance *Motor_3*, which was exported from *Folder_5*. The instance is located in *Folder_1*, a subfolder of *Folder_5*. The *Location* parameters of these folders are highlighted.

```

<Alias />
<Description />
<Area />
▶ <Location>Root\Folder_5</Location>
<GraphicalLocation>0</GraphicalLocation>
<Parameters />
<ApplicationInstances>
- <ApplicationInstance Type="SchneiderElectric.ProcessExpert.Application
  Version="2.4.4" IsValid="True" Action="Create" IsHierarchical="True">
  <!-- Please do not modify 'Identifier' as these are system gener
  If your intention is to change name of the instance Do it by ch
  ▶ <Location>Root\Folder_5\Folder_1</Location>
  - <Parameters>
    <Parameter Identifier="$Name" Value="Motor_3" />
    <Parameter Identifier="$Description" />
    <Parameter Identifier="$Area" />
  </Parameters>

```

Creating an Instance

To edit the XML export file to create a new instance in an existing folder, proceed as follows.

Step	Action
1	<p>Change the value of the <i>\$Name</i> parameter to the name of the new instance that you want to create.</p> <p>NOTE: To create multiple instances during import:</p> <ul style="list-style-type: none">• Copy the entire <i>ApplicationInstance</i> section of the export file that contains the data of the instance.• Paste it at the end of the section that you have copied, as many times as you want to create instances.• Change the value of <i>\$Name</i> to the new name of each instance.
2	Verify that <i>Action</i> is set to <i>Create</i> ; otherwise replace the value by <i>Create</i> .
3	Save your changes.

Updating an Instance

To edit the XML export file format to update an existing instance, proceed as follows.

Step	Action
1	For the instance that you want to update, change the value of <i>Action</i> from <i>Create</i> to <i>Update</i> .
2	Change the value of the parameters, page 246 that you want to update.
3	Save your changes.

NOTE: If you update data of instances that are already linked and want the link to be maintained, it is not required to update the data of the *InterfaceLinks* element that pertains to the link as long as the linked interfaces remain enabled. This applies even if you change the value of *\$Name* or the instance identifier (for example, by configuring an alias for a folder) for either or both instances.

Deleting an Instance and/or a Folder

To edit the XML export file to delete an existing instance and/or folder, proceed as follows.

Step	Action
1	<p>If you want to delete:</p> <ul style="list-style-type: none">• The instance and the folder containing the instance, verify that the export file also includes the information pertaining to the folder.• The instance only, proceed to step 3.
2	For the folder that you want to delete, change the value of <i>Action</i> from <i>Create</i> to <i>Delete</i> .
3	For the instance that you want to delete, change the value of <i>Action</i> from <i>Create</i> to <i>Delete</i> .
4	Save your changes.

Creating Folders

To edit the XML export file to create one or more folders or a folder hierarchy, proceed as follows.

Step	Action
1	Verify that <i>Action</i> is set to <i>Create</i> for the folders that you want to import; otherwise replace the value by <i>Create</i> .
2	Save your changes.

Updating a Folder

To edit the XML export file to update an existing folder, proceed as follows.

Step	Action
1	For the folder that you want to update, change the value of <i>Action</i> from <i>Create</i> to <i>Update</i> .
2	<p>Change the value of any of the following folder parameters that you want to update:</p> <ul style="list-style-type: none">• <i>Alias</i>• <i>Description</i>• <i>Area</i>• <i>Parameters</i> <p>If the value of the parameter that you want to update is blank, enter the new value in the following format <code><Parameter>Value</Parameter></code>, where <i>Parameter</i> represents one of the four parameters that you can change and <i>Value</i> the new parameter value.</p> <p>For example, to modify the blank Area parameter of a folder to <i>100</i>, enter <code><Area>100</Area></code>.</p> <p>NOTE: You must not change the <i>Folder Identifier</i> value.</p>
3	Save your changes.

Editing the XML File to Manage Interface Links

Overview

You can use any XML editing software to edit the export file. This lets you create and delete interface links.

The software applies the configuration when you import the edited XML file, page 254 into the application of the system.

If you modify the data of a link that is contained in the export file, the existing link is not deleted after import. The new link is created in addition.

Good Practices

- Before editing an application export file, create a copy of it. If needed, it lets you restore the original application.
- Edit only an up-to-date export file to help avoid data conflicts when importing.
- Use the data of exposed interfaces, page 244 to create interface links.

Linked Instances Data

For any actions on links, the data of the linked instances must appear in the application export file, page 246 even if the instances already exist in the target application.



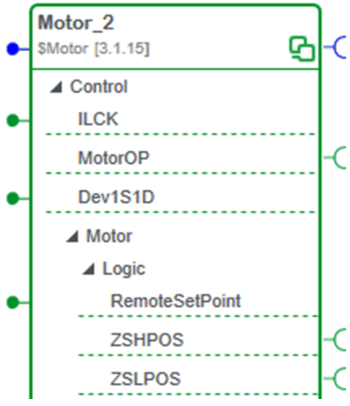

If the linked instances already exist in the target application with the same configuration and you do not want to modify them, change the value of *\$Action* to *Update* for each instance.

NOTE: If the instances already exist in the target application and you leave the value as *Create*, the **Resolve Uniqueness Conflicts** dialog box, page 260 opens during import and you need to select the *Update* action for the instances there.

Contents of the *InterfaceLinks* Element

The table describes the parameters of the *InterfaceLinks* element, page 244 that appears at the end of the XML application export file.

Parameter	Description
<i>Action</i>	Action performed by using the interface link data. Possible values: <ul style="list-style-type: none"> • Create (default): Creates new links in addition to existing links whose data has been exported. • Delete: Deletes links whose configuration matches the data in the imported file. NOTE: To create a new link in place of an existing link, you need to explicitly delete the existing link. NOTE: Entering <i>Update</i> produces the same result as <i>Create</i> .
<i>DestinationPath</i>	Path to the instance that is the destination of the link in the Application Explorer . The name of the system root folder, page 160 is replaced by <i>Root</i> .
<i>DestinationInstanceName</i>	Value of the <i>\$Name</i> parameter, page 191 of the instance that is the destination of the link.
<i>DestinationInstanceTemplateIdentifier</i>	Identifier of the template that is used by the instance, which is the destination of the link.
<i>DestinationInstanceIdentifier</i>	Identifier, page 177 of the instance that is the destination of the link.

Parameter	Description
<i>DestinationInterfacePath</i>	<p>Path to the destination interface in the composition of the instance.</p> <p>Example:</p> <p>In the following sample instance, value for interface <i>BoolVar</i> is "" (blank) because it has no parent element.</p> 
<i>DestinationInterfaceIdentifier</i>	<p>Identifier of the interface that is the destination of the link.</p> 
<i>IsDestinationInstanceHierarchical</i>	<p>Indicates if the Hierarchical Name function, page 189 is enabled for the instance.</p>
<i>SourcePath</i>	<p>Path to the instance that is the source of the link in the Application Explorer.</p> <p>The name of the system root folder, page 160 is replaced by <i>Root</i>.</p> <p>Example:</p> <p>For instance <i>Motor_1</i> of <i>System_1</i> that is located in <i>Folder_2</i>, which is a subfolder of <i>Folder_1</i>, the value is <i>Root\Folder_1\Folder_2</i>.</p>
<i>SourceInstanceName</i>	<p>Value of the \$Name parameter, page 191 of the instance that is the source of the link.</p>
<i>SourceInstanceTemplateIdentifier</i>	<p>Identifier of the template that is used by the instance, which is the source of the link.</p>
<i>SourceInstanceIdentifier</i>	<p>Identifier, page 177 of the instance that is the source of the link.</p>
<i>SourceInterfacePath</i>	<p>Path to the source interface in the composition of the instance.</p> <p>Example:</p> <p>In the following sample instance, the value for interface <i>ZSHPOS</i> is <i>Control\MotorLogic</i>.</p> 
<i>SourceInterfaceIdentifier</i>	<p>Identifier of the interface that is the source of the link.</p> 
<i>IsSourceInstanceHierarchical</i>	<p>Indicates if the Hierarchical Name function, page 189 is enabled for the instance.</p>

Creating a Link

To edit the XML export file to create a link, proceed as follows.

Step	Action
1	In the <i>InterfaceLinks</i> element, copy the source and destination data that pertains to an existing interface link and paste it inside the element.
2	Edit the necessary source and destination data.
3	Verify that the value of <i>Action</i> is <i>Create</i> .
4	Save your changes.

Deleting a Link

To edit the XML export file to delete a link, proceed as follows.

Step	Action
1	In the <i>InterfaceLinks</i> element, locate the source and destination data that pertains to the interface link and change the value of <i>Action</i> to <i>Delete</i> .
2	Save your changes.

Importing a CSV or XML File Into the Application

Overview

By using the **Import** command, you can modify the application with the configurations that you have made in the CSV or XML file.

The import process is performed in steps during which EcoStruxure Process Expert:

1. Validates the structure of the CSV or XML file.
2. If valid, displays the folders, asset workspaces, and instances contained in the imported file along with the action that you have configured for each one, informs you of detected issues, and lets you select which objects to import.
3. Imports selected objects that satisfy applicable rules (for example, uniqueness of instance identifiers and presence of template).
4. For the other objects, informs you of the detected uniqueness conflicts, page 260 and lets you resolve them so that you can import these objects.
5. Displays a summary of the import operation, which includes information about incorrect configurations and objects that were not imported.

Creating and deleting instances through import affects the instance count, page 180.

NOTE: Verify that the version of the templates that are used by the instances that you are importing exists in the Global Templates Library before you proceed with the import; otherwise you will not be able to import these instances. Refer to the topics describing how to:

- Browse application templates, page 163.
- Export/import templates, page 1035.

Software Version Compatibility

You can import into a system instances that were exported to XML format from Process Expert 4.0 and subsequent supporting versions.

If the XML export file was created from a version earlier than Process Expert 4.0, you need to import the file into the version from which it was created, migrate the database in steps to version 4.0, and export the instances again from this version.

Instance Import Behavior

The table describes the default import behavior when you configure the export file to *create* instances in the following scenarios.

Scenario	Result
An instance parameter value is invalid or no value is configured.	<p>The instance is created and configured with parameters whose value is valid.</p> <p>For the invalid or not configured parameter, the default value of the template is used.</p>
An instance element selection value is invalid or no value is configured.	<p>The instance is created and configured with element selections whose value is valid.</p> <p>For the invalid or not configured element selection, the value is set to false. The values that are configured for its parameters are not applied.</p>

Folders	Instance	Result
No <i>Folders</i> group, page 231 or no data, page 236 in the group.	Data is valid.	<ul style="list-style-type: none"> If the folders exist in the target application: The instance is created. If at least one folder in the instance path does not exist: The instance is not created.
Invalid data for at least one folder in the instance path.	Data is valid.	<ul style="list-style-type: none"> If the folders exist in the target application: The instance is created. If the folders do not exist, only those that have valid data and a parent folder with valid data are created. <p>The instance is created only if the folder that contains it is created.</p>

The table describes the import behavior when you configure the export file to *update* instances in the following scenarios.

Scenario	Result
An instance parameter value is invalid or no value is configured.	<p>Only parameters whose value is valid are updated.</p> <p>For the invalid or not configured parameter, the current value of the instance is retained.</p>
An instance element selection value is invalid or no value is configured.	<p>Only element selections whose value is valid are updated.</p> <p>For the invalid or not configured element selection, the current value of the instance is retained.</p>

Folders	Instance	Result
No <i>Folders</i> group, page 231 or no data, page 236 in the group.	Data is valid.	<p>The instance is updated.</p> <p>NOTE: When you import an XML file, folders whose data is invalid or missing are tagged with a red dot in the Import window.</p>
Invalid or missing data for at least one folder in instance path.	Data is valid.	

Folder Import Behavior

The table describes the default import behavior when you configure the export file to *create* folders in the following scenarios.

Scenario	Result
The parameter <i>Action</i> for all the folders in the export file is set to <i>Create</i> and the parent folder already exists in the target application.	<p>Only its subfolders are created if these do not exist yet.</p> <p>If a subfolder already exists, it is not created but any lower-level folder in the hierarchy is created if it does not exist yet. And so on.</p>
The data of at least one folder in the folder structure is invalid.	Only folders that have valid data and a parent folder with valid data are created.

The table describes the import behavior when you configure the export file to *update* folders in the following scenario.

Scenario	Result
The data of at least one folder in the folder structure is invalid.	Only folders that have valid data are updated.

NOTE: If EcoStruxure Process Expert is unable to propagate the value of the **Area** parameter to any object because the object is locked, it proceeds as follows:

- It displays a notification to inform you which item is locked.
- It undoes the area value propagation.
- It reverts the **Area** parameter value that you entered in the folder properties to its previous value.

NOTICE

NO CHANGE PROPAGATION

Ensure that the **Area** parameter value of a folder is propagated to the entire folder structure by referring to the notification panel. Otherwise, manually enter the **Area** parameter value in the folders where it was not propagated.

Failure to follow these instructions can result in equipment damage.

NOTE: If the **Area** parameter value of a folder is not propagated to the entire folder structure, alarms may be signaled in the wrong area in runtime.

The table describes the import behavior when you configure the export file to *delete* folders in the following scenario.

Scenario	Result
The folder that you are deleting contains subfolders and instances.	The subfolders and instances are deleted.

Interface Link Import Behavior

When you import two instances that are linked and select the **Relative Path** import setting, [page 258](#), which imports the instances to a location that is different from the one that is specified for the source and destination instance in the interface links data group, the link is maintained.

Aborting Import Tasks

After confirming the import by clicking **OK** in the **Import** window, the Abort icon is displayed in the [notification panel, page 85](#). Click the icon to cancel the task.

Only the user who selected the command is allowed to cancel it.

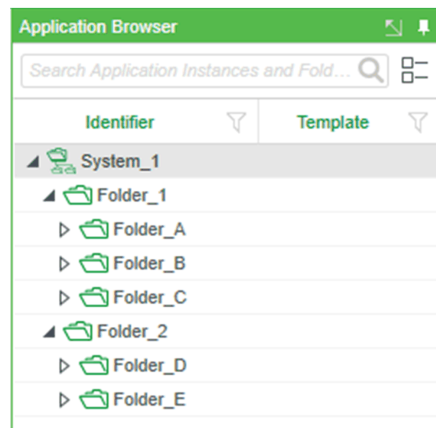
Folders and instances that already appear in the **Application Browser** at the moment you abort the task are retained given they belong to a tree structure that starts from the system root folder and that is completely imported at the moment the import task is aborted.

If you click the Abort icon after clicking **OK** in the **Resolve Uniqueness Conflicts** dialog box, instances and folders that you had selected to be created and/or updated are not imported. Already imported objects (non-conflicting ones) are retained.

In any case, links that exist between instances are not recreated when you abort an import task even if these instances are retained. This is because recreating links is the last step performed before completing an import task.

NOTE: If you are importing only a few objects, the task may complete before you are able to click the Abort icon.

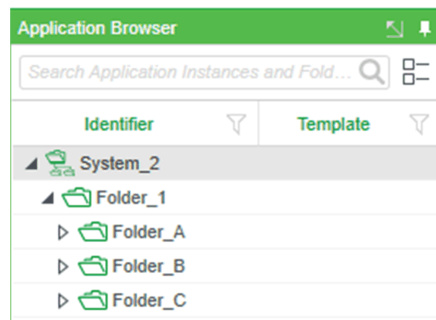
The following example illustrates two scenarios. Parent folders *Folder_1* and *Folder_2* are exported with their respective subfolders from the system root folder (*System_1*).



When you import the two structures into a system, folders are imported in sequence, structure after structure. If the import task is aborted:

- Before *Folder_C* and its contents is completely imported (displayed), nothing is retained.
- After *Folder_C* and its contents is completely imported (displayed), only the complete *Folder_1* tree structure and the instances therein are retained.

The following figure shows the objects that are retained when the import task is aborted after *Folder_C* and its contents has finished importing.



If a link exists, for example, between an instance in *Folder_A* and an instance in *Folder_C*, it is not recreated.

Opening the Application Import Window

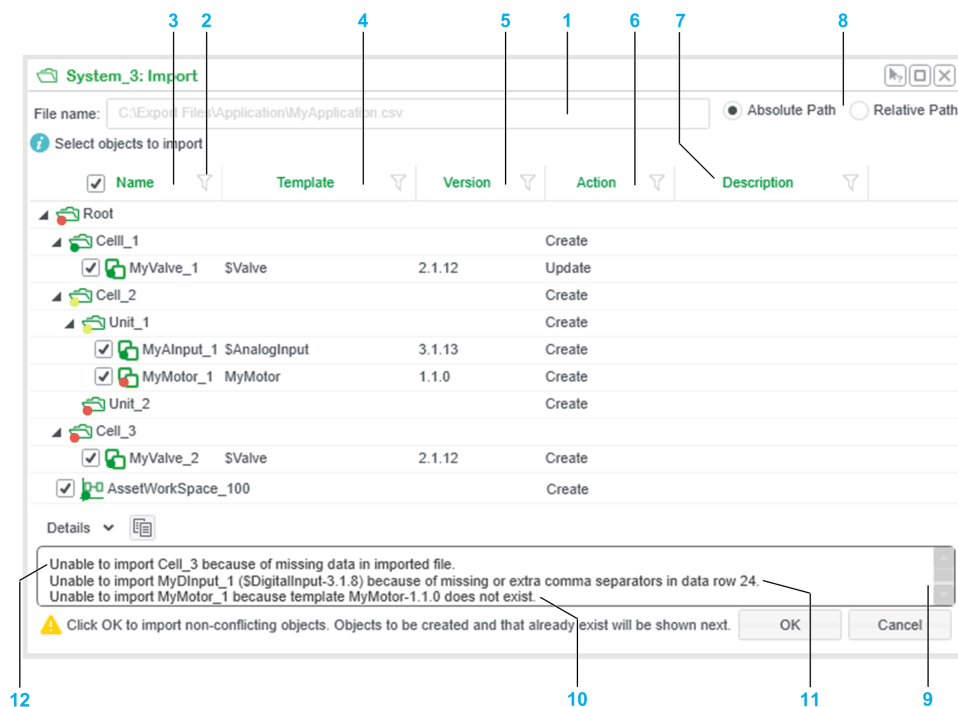
To open the application **Import** window, proceed as follows.

Step	Action
1	In the Application Browser , right-click the folder in which you want to import instances or that is to become the parent folder and select Import . Result: The Open dialog box opens.
2	Select the application export file and click Open . Result: The application Import window opens and displays the contents of the file.

NOTE: If you are importing only asset workspaces, you can right-click independently the root node in the **Application Browser** or **Asset Workspace** pane.

Import Window Description

The following figure shows an example of the application **Import** window. This example illustrates a scenario where three instances and one folder cannot be imported. Details are provided in the legend.



Item	Description
1	Full path of the file that you are importing.
2	You can sort and filter, page 120 data.

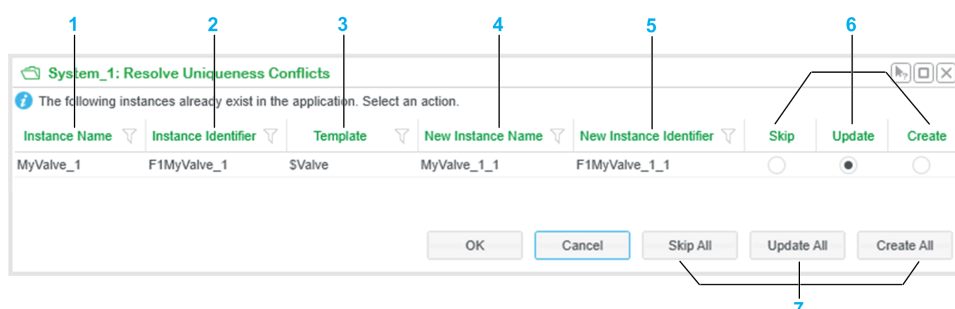
Item	Description	
3	<p>Identifier of the folders and asset workspaces and name (<i>\$Name</i>) of the instances contained in the imported file.</p> <p>A selected check box indicates that the object is imported if possible.</p> <p>A colored dot next to each object indicates the import status. Details are shown in a tooltip.</p> <p>NOTE:</p> <ul style="list-style-type: none"> Instances whose data is invalid are not shown. Instances whose data is valid are shown but may not be created, page 254. <p>Refer to the Details section.</p>	
	Color	Colored dot description
	Green	For instances: The template that is used by the instance (same identifier and version) is available in the Global Templates Library.
		For folders: Subfolders and instances inside the folder and its subfolders are marked with a green dot.
	Red	For instances: The template that is used by the instance (same identifier and version) is not available in the Global Templates Library. If you proceed, the instance is not imported.
		<p>For folders, may appear in these situations:</p> <ul style="list-style-type: none"> Subfolders and instances inside the folder and its subfolders are marked with a red dot. The folder data in the CSV file is invalid. The folder contains an instance whose data is invalid. <p>If the location of the instance is not specified, the red dot appears on the system root folder.</p>
		For asset workspaces, appears if the asset workspace identifier in the CSV file is invalid.
	Yellow	For folders only: The folder or its subfolders contains at least one subfolder or instance that are marked with a red dot while at least one instance or subfolder is marked with a green dot.
4	Identifier of the template that instance is using.	
5	Version of the template that the instance is using.	
6	<p>Indicates the action that are performed on the selected object according to the value of the <i>Action</i> parameter in the imported file.</p> <p>NOTE: When you create an instance, in the import summary that is shown in the notification panel, the action appears as <i>Create</i> for the instance but as <i>Update</i> for its elements and parameters.</p>	
7	Contents of the <i>\$Description</i> parameter of the object.	
8	Absolute Path	<p>During import, EcoStruxure Process Expert maintains the folder structure of the imported file independently from where you select the import command given the required folder data is present in the imported file or the folders exist in the target application.</p> <p>The system root folder appears as <i>Root</i>.</p>
	Relative Path	<p>During import, the folder from which you select the import command becomes the parent folder of the imported folder structure.</p> <p>However, folders that do not exist in the target application and whose data is missing in the import file are not imported. This may change the location of instances.</p> <p>NOTE: To be able to import an asset workspace when you are importing at the folder level, you need to select Absolute Path.</p>
9	Notification area, which contains information about detected issues.	
10	The message indicates that the instance cannot be imported because its template is not available in the Global Templates Library. As a result, the instance is tagged with a red dot.	

Item	Description
11	The message indicates that the instance that is located in folder <i>Unit_2</i> cannot be imported. As a result, the folder is tagged with a red dot and the instance is not shown in the window.
12	Folder <i>Cell_3</i> cannot be imported because of missing data in the imported file. As a result, the folder is tagged with a red dot and no action appears for it. Also, instance <i>MyValve_2</i> will not be created. It is tagged with a green dot because its template is available in the Global Templates Library.

Resolving Conflicts During Import

You can resolve the object-uniqueness conflicts that is detected during import, for example, if you are importing an instance or asset workspace and one with the same name or identifier, page 191 already exists in the application.

The **Resolve Uniqueness Conflicts** dialog box opens once non-conflicting objects have been imported. The following figure shows an example for an instance.



Item	Description
1	<i>\$Name</i> parameter of the instance as it appears in the imported file. The column can also show the identifier of an asset workspace.
2	Identifier of the instance based on the data of the imported file.
3	Identifier of the template that the instance uses.
4	New value for the <i>\$Name</i> parameter of the instance that is used to satisfy applicable naming rules if you select to create the instance. The column can also show the new identifier of an asset workspace.
5	Identifier of the object after import if you select to create it.
6	Buttons that let you select an action for individual objects: <ul style="list-style-type: none"> Skip: The existing instance and its links are not modified or the asset workspace is not imported. Update: Default action. Modifies the existing object with the imported parameter and link configuration. The name of the object is not modified. Create: Creates a new object with New Instance Name and the imported folder, parameter, and link configuration. <p>NOTE: Update is disabled if you have selected the Relative Path import setting and, as a result, the instance is imported in a different folder.</p>
7	Buttons that let you select an action for the conflicting objects as a whole. The actions are the same as when you select the action for objects individually. NOTE: Clicking these buttons overrides the selection you have made for individual objects.

Importing Objects into the Application

To import the content of an application export file into the application of a system, proceed as follows.

Step	Action
1	<p>In the application Import window:</p> <ul style="list-style-type: none"> Review information that is shown in the Details pane (if applicable). Select the objects that you want to import. Select the import settings that you want to apply.
2	<p>Click OK.</p> <p>Result: If you select to import with relative path, a confirmation dialog box opens.</p> <p>NOTE: Click Cancel to close the Import window without importing any object.</p>
3	<p>Click OK.</p> <p>Result: If no conflicting objects are detected:</p> <ul style="list-style-type: none"> The content of the file is imported and the import settings that you selected are applied. The imported objects appear in the Application Browser. A summary of the import process appears in the notification panel. <p>If conflicts are detected:</p> <ul style="list-style-type: none"> The objects that satisfy import rules are imported and the import settings that you selected are applied. For conflicting objects, the Resolve Uniqueness Conflicts dialog box opens; proceed to step 4.
4	<p>In the Resolve Uniqueness Conflicts dialog box, select the actions that you want to perform for conflicting objects.</p> <p>NOTE: Click Cancel to close the Resolve Uniqueness Conflicts dialog box without importing conflicting objects. Objects that satisfied import rules remain imported.</p>
5	<p>Click OK.</p> <p>Result:</p> <ul style="list-style-type: none"> The remainder of the objects are imported with the import settings that you selected. The entirety of the imported objects appear in the Application Browser. A summary of the entire import process appears in the notification panel.

Project Definition Stage

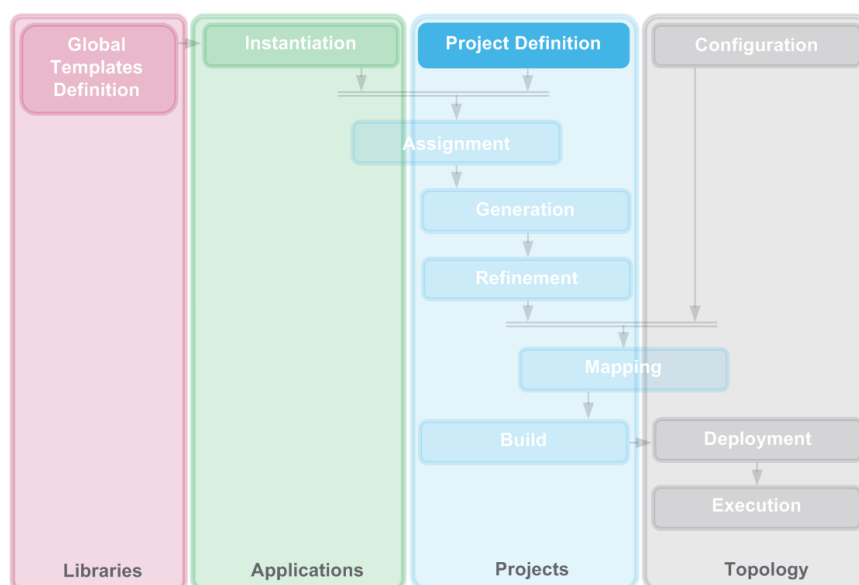
What's in This Chapter

Project Explorer and Project Browser	263
Control Project Definition Stage	268
Managing Control Projects	274
Supervision Project Definition Stage	298
Creating Supervision Project Components	304
Managing Supervision Projects	317
Participant Project Indicators	333

Overview

This chapter contains information to help you create the Control and Supervision projects of the system.

The following figure shows the position of the Project Definition stage within the system engineering life cycle.



Refer to the Project Definition stage, page 53 for a description of the purpose of this stage.

Project Explorer and Project Browser

Overview

This section describes the **Project Explorer**, which features the various editors that let you manage Control and Supervision Participant project components. It also describes the **Control Project Browser** and **Supervision Project Browser**, which let you create these projects and open their respective editors.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Project Explorer

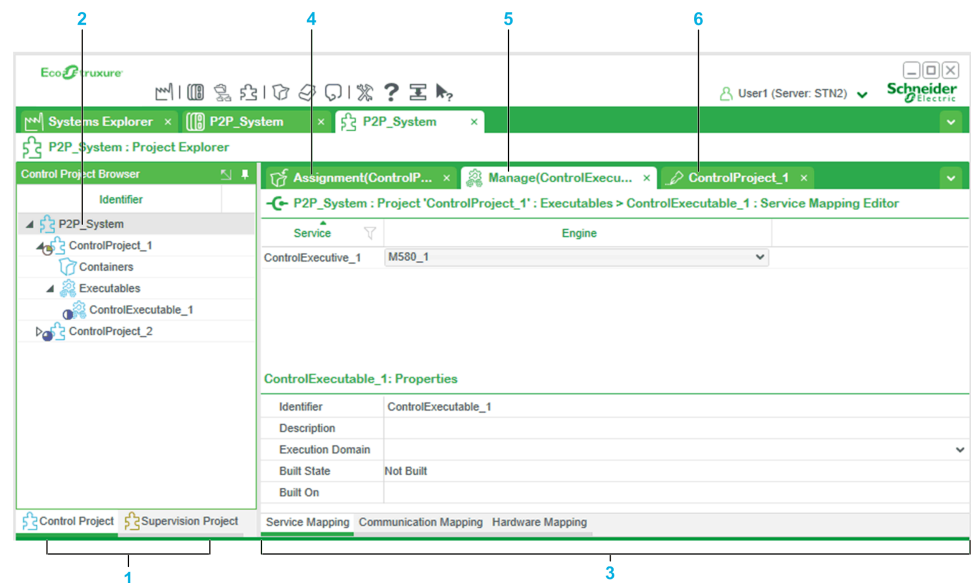
Opening the Project Explorer

To open the **Project Explorer** for a system, verify that the system is selected in the **Systems Explorer** or that at least one explorer/editor is already open for this system. Then, click the **Open Project Explorer** button in the engineering client toolbar.

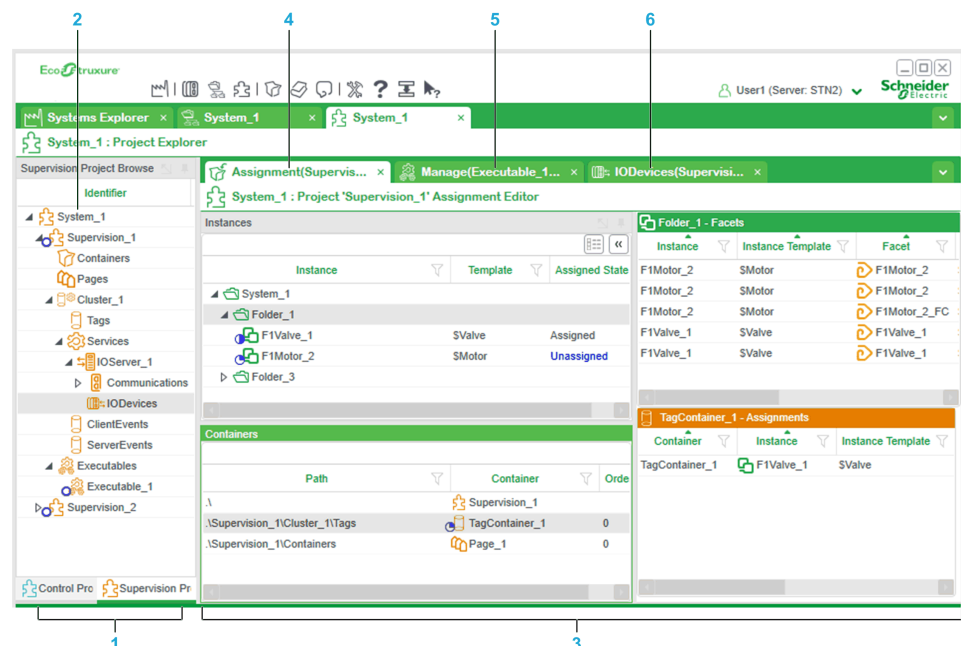
Otherwise, right-click the system in the **Systems Explorer** and select **Open Project**.

Project Explorer Window

The following figure shows an example of the **Project Explorer** for a system in which two Participant projects exist. The **Control Project Browser** is selected.



The following figure shows an example of the **Project Explorer** for a system in which two Participant projects exist. The **Supervision Project Browser** is selected.



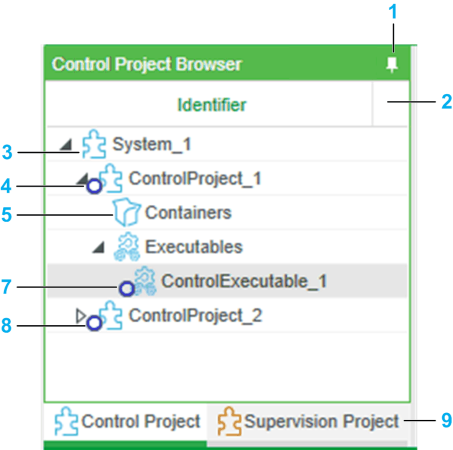
Item	Description
1	Tabs to show the Control Project Browser or Supervision Project Browser pane, page 266.
2	System root folder.
3	Remaining work area, page 107 of the Project Explorer when the project browser pane is pinned. The data that is shown depends on the project browser tab that is selected. The work area is empty by default.
4	Assignment Editor tab, page 341. It shows the containers of the Participant projects and lets you assign facets to them.
5	Executable management, page 720 tab, which lets you view and edit executable properties and perform the various mappings by selecting the corresponding subtabs: <ul style="list-style-type: none">• Service Mapping• Communication Mapping• Hardware Mapping (Control only)
6	Properties tab, page 270 for the Participant projects and their components other than executables.

Project Browser

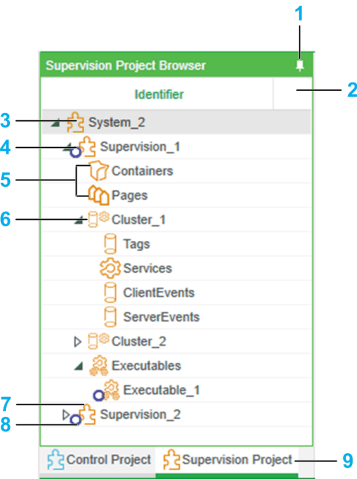
Description

The project browsers let you see, create, and interact with the Control and Supervision Participant projects that exist in a system.

The following figure shows an example of the **Control Project Browser** in which two projects have been created.



The following figure shows an example of the **Supervision Project Browser** in which one project has been created.



Item	Description
1	Pin button, page 113. Lets you hide the pane and restore it when it is docked.
2	You can customize, page 267 which columns appear.
3	System root folder.
4	Node giving access to project-level context menu commands.
5	Node giving access to container-level context menu commands.
6	Node giving access to cluster-specific context menu commands.
7	User-created executable, page 720.
8	Indicator icons, page 333.
9	Project type selection tabs.

Navigating Inside the Project Browser

The keys to navigate are the same as in the **Application Browser** pane, page 168.

Customizing Column Display

You can choose which columns you want to see by right-clicking a column header and selecting **Customize**. Certain columns are always shown.

For a description of the columns, refer to the topic describing properties of Control executables, page 722 and Supervision executables, page 765.

System Root Folder Actions

Right-click the system root folder to open a context menu with the following commands. The availability of commands depends on the tab of the Project Browser that is selected.

Command	Description
Create Control Project , page 269	Creates a new Control project that is associated to one of the supported controller families and displays it in the tree view of the Project Explorer .
Import Control Project , page 294	Opens an Open dialog box, which allows you to create a Control project by importing data from a project that was created with Control Expert. Export files in .stu or .xef format are supported.
Create Supervision Project , page 300	Creates a new Supervision project, and displays it in the tree view of the Project Explorer .
Open Application	Opens the Application Explorer , page 160, which allows you to start the instantiation stage or access the applications of the system.
Open Topology	Opens the Topology Explorer , page 270, which allows you to start the configuration stage or access the topology of the system.
Export	Opens a Save As dialog box, which allows you to create an export file containing the information of the Control projects, page 276 and Supervision projects, page 318 that exist in the system.
Import	Opens an Import dialog box, which allows you to display the compatible contents of an export file (.sbk) in the Import window. You can import entire Control projects, page 280 and/or Supervision projects, page 323, or parts of them into the system.

Control Project Definition Stage

Overview

This section describes how to create Control projects of a system and their sections.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Creating Control Projects

Before Creating a Control Project

At this stage of the system engineering life cycle, you need to know:

- Which family of controllers, page 591 you will be using to run the Control project.
- How you will organize the function block diagram (FBD) sections of the project.

NOTE: For information on working with Control projects of M580 safety controllers, refer to the topic describing the engineering of M580 safety systems, page 908.

Creating Control Projects

To create a Control project, in the **Control Project Browser**, right-click the system root folder, select **Create Control Project** and the controller platform, page 270 that is associated to the project.

Result: The Control project is shown in the tree view.

NOTE: In each system, you can create as many Control projects as needed. For example, you can create a project, which is engineered for a Modicon M580 controller in the plant and a project, which is engineered for an M340 controller for test purposes in the laboratory.

Control Project Actions

The table describes the Control project context menu commands.

Command	Description	Applies to
Assign Facets	Opens the Assignment Editor , which lets you create and manage sections and assign facets to them.	Assignment stage, page 340
Generate , page 426	Starts a process to create or update the logic at the project level.	Generation stage
Refine	Opens the Control Participant, which allows you to make changes to the logic of the project.	Refinement stage, page 446
Manage Network Variables , page 471	Opens the Manage Network Variables dialog box, which allows creating, modifying, and deleting network variables. These are variables that are used to share data by using peer to peer communication.	
Manage Data Backup Files , page 296	Opens the Manage Data Backup Files dialog box, which lets you view and delete data backup files that exist for Control Participant project.	Project Definition stage
Manage Peer to Peer , page 475	Opens the Peer to Peer Communication Configuration window, which lets you create and manage existing peer to peer communication mappings between controllers of a system. This functionality uses the specialized function blocks of the Control Participant. The command is available only if, for the executable of the other Control project, the service mapping has been performed.	Refinement stage, page 446
ReGenerate , page 430	Starts a process to recreate the logic from the beginning at the project level. NOTE: Certain changes that you have made during refinement may be discarded.	Generation stage
Last Action Summary , page 339	Opens a dialog box that indicates the last action that was performed and a summary of the result.	Project Definition stage

Command	Description	Applies to
Export , page 276	Opens a Save As dialog box, which allows you to create an export file containing the Control project information.	
Settings	Opens the Settings window, page 272, which lets you configure various project-related settings.	Assignment stage
Delete , page 271	Deletes the Control project including: <ul style="list-style-type: none"> Sections Assignments Executables, page 721 NOTE: For M580 safety projects, deletes also the associated M580 safety controller, page 923.	–
Rename	Allows you to enter a new identifier, page 270 for the project.	–
Properties	Opens the Properties window, which allows you to view and/or edit the properties of the Control project.	–

Containers Node Actions

The table describes the **Containers** node context menu commands.

Command	Description	Applies to
Assign Facets	Opens the Assignment Editor , which lets you create and manage sections and assign facets to them.	Assignment stage, page 340
Create FBD Section	Lets you create an FBD section, page 348 in the task folder that you select. The Assignment Editor opens in the background.	
Generate	Starts a process to create or update the logic at the section level.	Generation stage, page 396
ReGenerate	Starts a process to recreate the logic from the beginning at the section level. NOTE: Certain changes that you have made during refinement may be discarded.	

Executables Node Actions

The table describes the **Executables** node context menu commands.

Command	Description	Applies to
Create Executable	Creates one or more Control executables, page 721, which let you associate one or more controllers to the Control project, map project facets to hardware, and create communication channels by using the I/O scanner service of the controller.	Mapping stage, page 720

Control Project Properties

Right-click the Control project and select **Properties**.

You can view and/or edit the following items.

Item		Description
General	Identifier	The identifier must be unique within the system.
	Description	Optional. You can enter a description of the project with free-form text.
Controller Family	Value	Indicates the controller platform that you have associated to the project at the time of creation. Possible values: <ul style="list-style-type: none"> • M580 • M580 Safety • M340 • Quantum Read-only. NOTE: When you associate a Control project with a controller family, the assignment of facets to the project is limited to the facets that have the corresponding attribute (see EcoStruxure Process Expert, Global Templates, Reference Manual) defined.

Deleting a Control Project

When you select the command to delete a Control project, a confirmation dialog box opens. Once you have confirmed the deletion, EcoStruxure Process Expert executes the following actions in relation to the project that you are deleting:

- Deletes any components of the project and their contents, including any assignments and mappings.
- Deletes any associated files (for example, data backup files and expressions, page 363 for container naming rules).

NOTE: For M580 safety projects, deletes also the associated M580 safety controller, page 923.

Project Settings Window

The **Settings** window opens when you select the **Settings** command in the context menu of a Control or Supervision project.

Section	Description
Container Naming Rules	<p>Lets you configure expressions, page 363, which define how sections are created and how Control facets are assigned to the project.</p> <p>The section appears only for Control projects.</p>
Assignment	<p>Lets you set the functionality, which can automatically assign remaining candidate facets, page 381 of instances.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Enabled • Disabled
Project Analysis	<p>Lets you enable or disable the automatic analysis of the Control project, which is performed when you generate the project/sections or save refinement changes.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Enabled • Disabled (default) <p>However, if Configuration Change On The Fly (CCOTF) is enabled, page 446 and Project Analysis is disabled, the setting is overridden and an analysis is performed when you save offline refinement changes.</p> <p>When the analysis of the Control project is not performed:</p> <ul style="list-style-type: none"> • The Control project validity indicators, page 336 are not functional. • The time required to generate and save refinement changes is reduced. <p>You can analyze the Control project manually by refining it offline and clicking Build > Analyze on the Participant menu bar. If you save refinement changes, the project validity indicators indicate the status of the project.</p> <p>The Project Analysis section appears only for Control projects.</p> <p>NOTE: For M580 safety-related Control projects, the setting also impacts the configuration, page 923 of the associated M580 safety controller.</p>
Generation	<p>Automatically generates sections of the Control project that are impacted by changes to facets, page 407, which are being assigned or already assigned to them.</p> <p>Other sections whose facet assignment is not impacted need to be generated manually.</p> <p>The setting remains effective until you change it.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Enabled • Disabled: Default value. You need to generate all sections of the Control project manually. <p>NOTE: You can still use the Generate and Generate and Build commands while automatic generation is enabled.</p>

NOTE: The **OK** and **Cancel** buttons apply to all the sections of the **Settings** window.

Creating Control Project Sections

Overview

In the **Project Explorer**, you can create only FBD sections in a Control Participant project by using context menu commands.

Creating FBD Sections

To create FBD sections in task folders, open the **Assignment Editor** by selecting **Assign Facets** from the context menu of the **Containers** node in the **Control Project Browser**.

For a description of sections and actions that you can perform, refer to the topic describing the **Assignment Editor**, page 343.

NOTE: You can create FBD sections also by refining the Control Participant project or importing them.

Creating Non-FBD Sections

You can create non-FBD sections by:

- Refining, page 446 the Control Participant project and creating the sections.
- Refining the Control Participant project online, page 874, creating the sections, and updating, page 897 the Control project.
- Importing, page 280 the sections.

Non-FBD sections are sections in any of the following languages: IL, LD, LL984, SFC, and ST. They appear also in the **Containers** pane of the **Assignment Editor**.

NOTE: In the following cases, existing non-FBD sections may be displayed only after you make a change by using the **Refine** command, page 446 at the project level and save it:

- The Control project is contained in a database of an earlier version that you have migrated.
- The Control project is contained in a system that you have restored by using a backup file created with an earlier version, page 147.

Managing Control Projects

Overview

This section describes how to use the **Project Explorer** to manage entire Control projects or sections of such projects.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Control Project Section Order

Overview

EcoStruxure Process Expert assigns an order to sections, starting at 0, and displays it in the **Containers** pane, page 342. The values appear in the **Order** column.

It indicates the order in which EcoStruxure Process Expert generates the sections of the Control project and thereafter the execution order of the sections by the Control Participant.

You can change the order of sections inside the **Containers** pane.

EcoStruxure Process Expert applies changes to the order of sections in the logical Control Participant project when you perform either action:

- Select the **Refine** command at the project level (it is not required to make a change or save).
- **Generate**, page 430 the entire project or a section of the project.
- **Regenerate**, page 434 the entire project or a section of the project.

You then need to propagate the change downward in the system engineering life cycle.

NOTE: You can also change the order of sections by refining, page 446 the Control Participant project or by refining the deployed Control project online, page 874 and updating the Control Participant project.

Moving Sections

In the **Assignment Editor**, you can move sections only within the **MAST** or **FAST** task folder.

To move sections from the **MAST** to the **FAST** task folder or the other way around, refine the Control Participant project.

Changing the Order of Sections

To change the order of sections of a Control project, proceed as follows.

Step	Action
1	In the Containers pane of the Assignment Editor , clear filters, page 122.
2	Drag one section up or down to a new position. Result: A tooltip indicates the position that the section will occupy. NOTE: In the Assignment Editor , you can select multiple, contiguous or non-contiguous sections that belong to the same task folder.
3	Release the mouse button. Result: The section is inserted at the selected position and the values of the Order column are updated.
4	Apply the new order to the logical Control project by performing the necessary action.

NOTE: To change the order of the facets inside an FBD section, refer to *Order of Control Facets*, page 394.

Exporting Control Projects

Overview

The export functionality lets you create a non-editable export file (.sbk), which contains selected Control project information.

You can import, [page 280](#) this data to transfer project information to a different system or to merge different Control projects.

However, you cannot merge data related to communication mapping by import. You can either replace the existing [communication channel, page 725](#) and/or peer to peer [communication data, page 471](#) or not import the data.

NOTE: To export the Control and Supervision projects of a system at once in one export file, right-click the system root folder in either **Project Browser**, select **Export**, and follow the project export procedure described in this section. If applicable, also refer to the topic describing the export and import of Supervision projects, [page 317](#).

Exporting Control Projects of M580 Safety Controllers

For information about exporting Control Projects of M580 Safety controllers, refer to the topic describing how to [engineer systems with M580 safety controllers, page 914](#).

Software Version Compatibility

The version of EcoStruxure Process Expert where you import an export file must be the same as the version that was used to create the export file (a service pack is considered the same version); otherwise, you cannot import the file.

Project Related Data

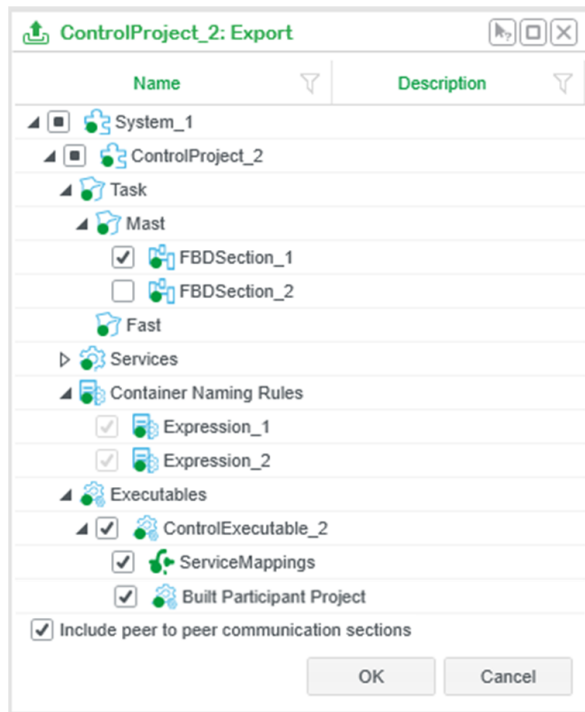
To be able to use the export file to import a project into a different system, verify that the instances, their templates, and topological entities related to the project exist in the target system [before importing the project, page 282](#).

NOTE: The parameter values of instances, which are the source of the exported facets are not contained in the export file. However, the export file contains information that allows to determine during import if the configuration of an instance in the target system is different from the source instance.

Project Export Window

The Control window lets you select, which project information the export file will contain.

The following figure shows an example of the **Export** window where the check box for *FBDSection_2* has been cleared. To indicate that at least one item of the Control project is not selected, the check boxes of parent nodes are filled with a square.



The following table indicates the meaning of the colored dots that appear in the **Export** window next to Control project items.

Dot color	Description
Green	You can export the item.
Red	<p>You cannot export the item because it contains at least one facet with the following state:</p> <ul style="list-style-type: none"> Deleted Unassigned Out of Date Moved <p>Generate sections that contain facets with these states to export the Control project.</p>
Yellow	<p>Indicates that the item contains at least one sub-item that is tagged with a red dot.</p> <p>The item can only be exported partially.</p>

NOTE: To expand/collapse a node in the **Export** window, select the node and press **Enter**.

Export Settings

The table describes which Control project information you can export. The information is exported by default.

Item		Description
<i>System_n</i>		Unselecting the item excludes any Control project and disables the creation of the export file.
<i>ControlProject_n</i>		Unselect the item to exclude the entire project from the export file.
Task folder		You cannot unselect the item.
	Mast and Fast task folders	You cannot unselect the item.
	Program sections	You can unselect sections individually. Selecting an FBD section automatically selects the facets that are assigned to it.
	Peer to peer communication sections ⁽¹⁾	To select the peer to peer communication sections, page 475 that were created by using the Manage Peer to Peer command, select the Include peer to peer communication sections check box in the Export window. You cannot select sections individually.
Safe Program		The item is available only for Control projects of M580 safety controllers, page 908. You cannot unselect the item.
Container Naming Rules		Corresponds to the expressions, page 363 that are configured. You cannot unselect expressions.
Executables		You cannot unselect the item.
	<i>ControlExecutable_n</i>	You can unselect each executable and its contents individually. Selecting an executable automatically selects associated mappings.
	Service mappings	Selecting the item exports any data related to the service mapping including any required contents in the contents repository. Unselecting the item also unselects both the associated hardware and communication mapping. NOTE: If the project contains peer to peer communication configurations, refer to the section describing the export of peer to peer communication configurations, page 279.
	Hardware mappings	You cannot select hardware mappings individually. Selecting the item also selects the associated control executable and service mapping.
	Communication mappings	You cannot select communication mappings individually. Selecting the item also selects the associated executable and exports the complete communication mapping data. NOTE: If the project contains peer to peer communication data, refer to the section describing the export of peer to peer communication channels and variable mappings, page 279.
	Built Participant Project	Corresponds to the built Control Participant file (.stu).
Backups		You can select data backup files individually. The time stamp and description of each backup file, page 880 are shown. The format of the time stamp is YYYYMMDDHHMMSS.
(1) Not selected for export		

Exporting Peer to Peer Communication Channels and Variable Mappings

To recreate peer to peer communication that exists between owner and consumer Control projects (peer to peer communication channels and variable mappings created by using [peer to peer templates](#), [page 463](#)) when you import the Control projects, you need to export the following:

- The owner project and its executable.
- The consumer project and its communication mapping.

NOTE: Including communication mappings in the export file may also include communication channels other than those for peer to peer communication (for example, communication channels for I/O devices).

Exporting Peer to Peer Communication Configurations

To recreate peer to peer communication configurations that exist between local and remote Control projects (peer to peer communication sections and variable mappings created by using the **Manage Peer to Peer** command, [page 475](#)) when you import the Control projects, do the following:

- Export the local Control project.
- Export the remote Control project and the corresponding executable with its service mapping.
- Select the **Include peer to peer communication sections** check box in the **Export** window.

Aborting Export Tasks

After confirming the export by clicking **Save** in the **Export** window, the Abort icon is displayed in the [notification panel](#), [page 85](#). Click the icon to cancel the task. The export file is not created.

Only the user who selected the command is allowed to cancel it.

Exporting a Control Project

To export a Control project to file, proceed as follows.

Step	Action
1	In the Control Project Browser , right-click the Control project that you want to export and select Export . Result: The Export window opens and displays the Control project. NOTE: To export two or more Control projects, right-click the system root folder.
2	Select the project items that you want to export.
3	Click OK . Result: The Export window closes and the Export dialog box opens.
4	Select a location to save the export file, enter a file name, and click Save . Result: The export file containing the selected Control project information is created.

Importing Control Projects

Overview

The **Import** command lets you import data of Control projects from an export file (.sbk), which was created by using the [export, page 276](#) functionality of the **Project Explorer**.

If the system where you are importing a Control project already contains a Control project with the same identifier, EcoStruxure Process Expert compares the data of both projects. It then proceeds with a partial import, according to the import settings that you select and the import rules that are described in this topic.

Data related to communication mapping, [page 725](#) and peer to peer communication configurations, [page 475](#) cannot be imported partially. You can either import it and replace existing data or not import it. Specific conditions that apply when importing this type of data are described in this topic.

NOTE: If the export file also contains information of a Supervision project, you can see the data that is related to the Supervision project, and have the choice to import it, [page 323](#) as well.

Importing Control Projects of M580 Safety Controllers

For information about importing Control Projects of M580 Safety controllers, refer to the topic describing how to [engineer systems with M580 safety controllers, page 914](#).

Software Version Compatibility

The version of EcoStruxure Process Expert where you import an export file must be the same as the version that was used to create the export file (a service pack is considered the same version); otherwise, you cannot import the file.

Naming Rules for Variable Names and Section Identifiers

Names of variables and identifiers of sections that you import into a system need to satisfy the [naming rules, page 32](#) for the generation process to complete successfully.

Checking Consistency After Import

After importing a Control project, perform a consistency check. EcoStruxure Process Expert verifies if the Control constituents of the project are consistent with the facets that are assigned to the project.

For more information on the process, refer to the topic describing how to [manage the consistency of Control logic, page 460](#).

Import Settings

The following table describes which Control project information you can import into a system. If the information is contained in the export file, it is imported unless mentioned otherwise.

Item		Description	
System_n		Unselecting the item excludes any Control project and disables the import of the export file.	
ControlProject_n		Unselect the item to exclude the entire project from the import process.	
Task folder		You cannot unselect the item.	
	Mast and Fast task folders		You cannot unselect the item.
		Program sections	<p>You can select sections individually.</p> <p>Displays the identifier and path of the facets that are assigned to the section.</p> <p>A Constituent File is associated to each FBD section.</p> <p>Selecting an FBD section imports any facets that are assigned to this section.</p> <p>NOTE: A section can be selected only if it is tagged with a green dot, page 290.</p>
		Peer to peer communication sections	<p>Imports the peer to peer communication sections, page 475 that were created by using the Manage Peer to Peer command. Refer to the topic describing peer to peer communication configuration import rules.</p> <p>To not import the sections, clear the Include peer to peer communication sections check box in the Import window.</p> <p>You cannot select sections individually.</p> <p>NOTE: The sections and the check box are displayed only if the export file contains the required project items, page 279.</p>
		Constituent File (FBD sections only)	<p>If the identifier of the imported FBD section already exists in the target project, the selection of the constituent file has an impact on the result of the import, page 283.</p> <p>NOTE: The Constituent File check box is selected and disabled if the identifier of the imported FBD section does not exist in the target project.</p>
Safe Program		<p>The item is available only for Control projects of M580 safety controllers, page 908.</p> <p>Unselect the item to exclude the entire safety program from the import process.</p>	
Container Naming Rules ⁽¹⁾		Expressions, page 363 are imported only if a Control project with the same identifier does not exist in the target system.	
Executables		You cannot unselect the item.	
	ControlExecutable_n		<p>You can select each Control executable and its contents individually.</p> <p>Selecting the Control executable automatically selects existing mappings.</p>
		Service mappings	<p>Data related to the service mapping including required contents in the Content Repository.</p> <p>Selecting the item automatically selects the associated executable.</p> <p>NOTE: The service mapping of the executable that is used for peer to peer communication configurations, page 279 is required to import the peer to peer communication sections.</p>
		Hardware mappings	<p>You cannot select hardware mappings individually.</p> <p>Selecting the item automatically selects the associated executable and service mapping.</p>
		Communication mappings	<p>The communication mapping data created by using peer to peer templates and communication with I/O devices.</p> <p>Selecting the item automatically selects the associated executable and service mapping.</p>
		Built Participant Project	<p>Corresponds to the Control Participant file (.stu).</p> <p>It appears and is selected only if the built Participant project was selected at the time of export.</p>

Item	Description
Backups	<p>You can select data backup files individually</p> <p>The time stamp and description of the backup file, page 880 are shown. The format of the time stamp is YYYYMMDDHHMMSS.</p> <p>Importing a data backup file makes it available in the Manage Data Backup File dialog box, page 296 and Deploy Data dialog box, page 824.</p>
(1) Project dependent.	

Import Rules for Project-Related Objects

Facets assigned to Control projects are related to instances of an application and mappings of project executables are related to entities of the topology. For the Control project import to complete entirely, these project-related objects need to be present in the system in which you want to import the project.

The following table describes the impact on the import operation when project-related objects are not present or have a different configuration. It also describes the corrective action.

Project-related object that is not present or different	Impact if not present in the target system	Required action
An instance whose facets are assigned to the Control project that you are importing (same identifier, template identifier, template version, and status Valid) is not present.	<p>You can import the project but not the sections to which such facets are assigned. This requires that the project contains at least one section that can be imported, page 290, even empty.</p> <p>A dialog box opens, page 291 asking you to confirm the partial import.</p> <p>NOTE: Performing a partial import may cancel the entire import operation if import rules for peer to peer communication variables, page 287 are not satisfied.</p>	<p>Do either of the following first to create missing instances:</p> <ul style="list-style-type: none"> Use the export/import functionality, page 227 of the Application Explorer to import missing instances. Create instances. This may require importing templates first. <p>You can obtain information about the missing instances and their templates in the tooltip of the container that is tagged with a red dot.</p>
An element of the instance creating a facet that is being imported is not selected.	<p>The facet that is normally created by this element is not imported.</p> <p>The notification panel provides information about the facet that is not imported.</p> <p>For example, you import a Control project to which the facet created by the <i>Maintenance</i> element of an instance is assigned but in the application of the target system, the <i>Maintenance</i> element of the same instance is not selected.</p>	Edit the instance in the Application Explorer and select the corresponding element.
The properties of an instance are different from the instance configuration at the time of export. The difference lies in the element selection (services) and/or the parameter values, page 182.	The assignment status of facets that are imported is set to Out Of Date .	Generate the project o section to apply the changes.
A topological entity to which you have mapped the executable of the project (service and/or communication) that you are importing (same identifier, same devices, and same services).	You can import the project but not the Control project executable and any of its associated mappings.	Use the export-import functionality, page 647 of the Topology Explorer first to create the missing topological entities.

Import Rules for Sections

The following table describes the import rules that apply to FBD and non-FBD sections with respect to the section order if a Control project with the same identifier already exists in the target system.

Contents of export file selected for import	Section in Control project of target system	Result in Control project of target system after import
Section1, order 0 Section2, order 1 Section4, order 2	Section3, order 0 Section1, order 1	Section3, order 0 Section1, order 1 Section2, order 2 Section4, order 3 The order of existing sections is maintained and imported sections are added at the end. NOTE: If a section already exists in the target system, its order is maintained even if the order of the section in the export file is different.

The following table describes the import rules that apply to non-FBD sections when a Control project with the same identifier already exists in the target system.

Contents of export file selected for import	Section in Control project of target system	Result in Control project of target system after import
Section1	Section1	Section1 from export file replaces Section1 in the Control project of target of the target system.
Section1	Section2	Section1 and Section2 exist in the Control project of target of the target system.

Import Rules for Facets

EcoStruxure Process Expert applies import rules to facets if the system into which you are importing a Control project already contains a Control project with the same identifier.

The following are the general import rules that apply if facets with the same identifier contained in FBD sections with the same identifier exist in both the target project and the export file that you are importing:

- If you *select* the constituent file of the section:
 - Facets from the export file overwrite facets of the target system and maintain the state they have in the export file.
 - Facets that exist only in the export file are imported into the target system and maintain the state they have in the export file.
 - Facets that exist only in the target system become **Assigned** and **Non Generated**. If the status of the facet is deleted, unassigned, or unlinked, the facet is removed from the section after import.
- If you *unselect* the constituent file of the section:
 - Facets with the same identifier maintain the state they have in the target system.
 - Facets that exist only in the export file are imported into the target system and become **Assigned** and **Non Generated** unless the generation status of the facet is **Unlinked**.
 - Facets that exist only in the target system maintain their state.

NOTE: EcoStruxure Process Expert changes the assignment status of any facet of an instance to **Out Of Date** if a property of an instance, page 182 is different from what it was at the time of export. The notification panel provides information about the change of the assignment status after import is complete.

The following tables illustrate these import rules based on the different use cases. The section and facet names are examples, and represent sections and facets in the **MAST** or **FAST** task folders of the projects.

If you *select* the FBD section **Constituent File** check box.

Contents of export file selected for import		Item in Control project of target system		Result in Control project of target system after import
Section identifier is identical				
Section1	Facet1	Section1	–	Facet1 from export file with status from export file.
Section1	Facet1	Section1	Facet1	Facet1 from export file with status from export file.
Section1	Facet1	Section1	Facet2	Facet1 from export file with status from export file. Facet2 with status non-generated. The imported facets are added at the end of existing facets in the target section. If the status of Facet2 is deleted, unassigned or unlinked, the facet is removed from the section after import.
Section identifier does not exist in target project				
Section2	Facet3	–	–	Section2 with Facet3 from export file with status of export file. The imported sections are added at the end of existing sections in the target task folder.
– The item does not exist				

If you *unselect* the FBD section **Constituent File** check box.

Contents of export file selected for import		Item in Control project of target system		Result in Control project of target system after import
Section identifier is identical				
Section1	Facet1	Section1	–	Facet1 from export file with status non-generated unless generation status of Facet1 is Unlinked .
Section1	Facet1	Section1	Facet1	Facet1 from target system with same status.
Section1	Facet1	Section1	Facet2	Facet1 from target system with status non-generated. Facet2 from target system with same status. The imported facets are added at the end of existing facets in the target section.
– The item does not exist				

Import Rules for Executables and Mappings

Communication mapping is created with the mapping information from the export file if the following conditions are fulfilled:

- For peer to peer communication: The export file contains *necessary data*, [page 279](#) of both the consumer and the owner projects and the corresponding elements are selected for import.
- The I/O scanner service of the client controller that is mapped to the imported executable is enabled.
- Sufficient free memory is available in the client and/or server memory length property of the controllers to fit the communication channels that exist in the export file.
- The client and server controllers that are mapped to the imported executables are connected to the same logical network.

- For I/O device communication channels, in addition:
 - The I/O scanner service that is imported needs to match with the service of the topological I/O device entity with which communication is to be implemented. Refer to the description of the **Service** column in the **Device IO** section of the **Communication Mapping Editor** window, page 726.
 - The topological I/O device entity with which communication is to be implemented needs to be connected to the same logical network as the controller.

If the export file contains mapped peer to peer variables, refer to the topic describing import rules for peer to peer communication variables, page 287.

If EcoStruxure Process Expert is unable to import communication mappings, a dialog box opens, page 292 indicating the reason.

The **Build State**, page 781 of a **Built** executable is updated after import depending on the scenario.

NOTE: If the system into which you are importing a Control project already contains a Control project and an executable with the same identifier, EcoStruxure Process Expert applies import rules to executables and service, hardware, and communication mappings. These rules are described in the following tables.

The following table describes the import rules that apply to **Built** executables based on different scenarios.

Contents of export file selected for import	Item in Control project of target system	Resulting Build State in Control project of target system after import
Executable1	Does not exist	Out Of Date
Executable1	Executable1	Out Of Date
Executable1	Executable2	Executable1: Out Of Date . Executable2: Out Of Date .

The following table describes the import rules that apply to mappings based on different scenarios.

Contents of export file selected for import	Item in Control project of target system	Result in Control project of target system after import
Service mapping	Exists	Service mapping of target system is not changed. NOTE: If the export file contains peer to peer communication configurations, refer to the section describing the Import rules for peer to peer communication configurations, page 287.
	Does not exist	Service mapping is created with the mapping information from the export file.
Hardware mapping	Exists	Hardware mapping of the target project is updated with the mapping information from the export file in the following way: <ul style="list-style-type: none"> • Existing mappings of the target project are overwritten. • New mappings are created. • Mappings of the target project that are not affected by the import process are maintained. NOTE: Any new mapping for which the required mapping interface from the topological instance of the target project is not present is rejected.
	Does not exist	Hardware mapping is created with the mapping information from the export file. NOTE: Any mapping for which the required mapping interface from the topological instance of the target project is not present is rejected.

Contents of export file selected for import	Item in Control project of target system	Result in Control project of target system after import
Communication mapping	Exists	<p>Communication mapping is overwritten as a whole if the following conditions are fulfilled:</p> <ul style="list-style-type: none"> For peer to peer communication: The export file contains <i>necessary data</i>, page 279 of both the consumer and the owner projects and the corresponding elements are selected for import. Sufficient free memory is available in the client and/or server memory length property of the controllers to fit the communication channels that exist in the export file. For I/O device communication channels, in addition: <ul style="list-style-type: none"> The I/O scanner service that is imported needs to match with the service of the topological I/O device entity with which communication is to be implemented. Refer to the description of the Service column in the Device IO section of the Communication Mapping Editor window, page 726. The topological I/O device entity with which communication is to be implemented must be connected to the same logical network as the controller. If a service mapping exists, it needs to be identical to the service mapping contained in the export file for the Control project that you want to import. <p>NOTE: If the export file contains mapped peer to peer variables, refer to the topic describing import rules for peer to peer communication variables, page 287.</p>
Communication mapping	Does not exist	<p>Communication mapping is created with the mapping information from the export file if the following conditions are fulfilled:</p> <ul style="list-style-type: none"> For peer to peer communication: The export file contains <i>necessary data</i>, page 279 of both the consumer and the owner projects and the corresponding elements are selected for import. The I/O scanner service of the client controller that is mapped to the imported executable is enabled. Sufficient free memory is available in the client and/or server memory length property of the controllers to fit the communication channels that exist in the export file. The client and server controllers that are mapped to the imported executables are connected to the same logical network. For I/O device communication channels, in addition: <ul style="list-style-type: none"> The I/O scanner service that is imported needs to match with the service of the topological I/O device entity with which communication is to be implemented. Refer to the description of the Service column in the Device IO section of the Communication Mapping Editor window, page 726. The topological I/O device entity with which communication is to be implemented must be connected to the same logical network as the controller. If a service mapping exists, it needs to be identical to the service mapping contained in the export file for the Control project that you want to import. <p>NOTE: If the export file contains mapped peer to peer variables, refer to the topic describing import rules for peer to peer communication variables, page 287.</p>

Import Rules for Peer to Peer Communication Configurations

To recreate peer to peer communication configurations, page 475 that exist between exported local and remote Control projects, ensure to do the following:

- Import the local Control project.
- Import the remote Control project and the corresponding executable with its service mapping.
- Select the **Include peer to peer communication sections** check box in the **Import** window.

If a Control project with the same identifier exists in the target system and you select the **Include peer to peer communication sections** check box in the **Import** window, peer to peer communication sections and variable mappings in this Control project, which were created by using the **Manage Peer to Peer** command, page 475 are replaced with those of the export file.

Import Rules for Peer to Peer Communication Variables

To import communication mappings as a whole, EcoStruxure Process Expert must be able to create variable mappings that exist in the export file.

If none of the imported Control projects exist in the target system, the necessary variables are created, given that import rules for project-related items are satisfied.

If either Control project that you are importing already exists in the target system but a variable that is mapped to a peer to peer communication channel present in the export file has been modified or deleted, two scenarios are possible:

- The variable was created by refining the Control project. In this case, when you import communication mappings, EcoStruxure Process Expert is unable to recreate the variable. Therefore, it cannot create variables mappings and the entire import process is canceled and rolled back. A dialog box opens to inform you.

You need to recreate the variable manually to be able to import communication mapping.

- The variable was created by a facet assigned to the Control project. In this case, if you select the constituent file of the section to which the facet is assigned, the import process is able to recreate the required variable and the import succeeds, given that other necessary conditions are fulfilled.

The following example illustrates the second scenario: You have created two Control projects, *Owner* and *Consumer* between which a peer to peer communication channel exists. *Var_1* is mapped to this communication channel. The variable exists as network variable in *Consumer* (with *P2P* as custom attribute) and as variable in *Owner*.

You have exported both projects with their complete contents and have kept on working on both projects, which altered the communication mapping. After a while, you realize that you are not satisfied with some changes you made and therefore, you want to recover the original projects and communication mapping while keeping some of your changes (for example, you want to keep an *FDB* section that you have added to *Consumer* after exporting).

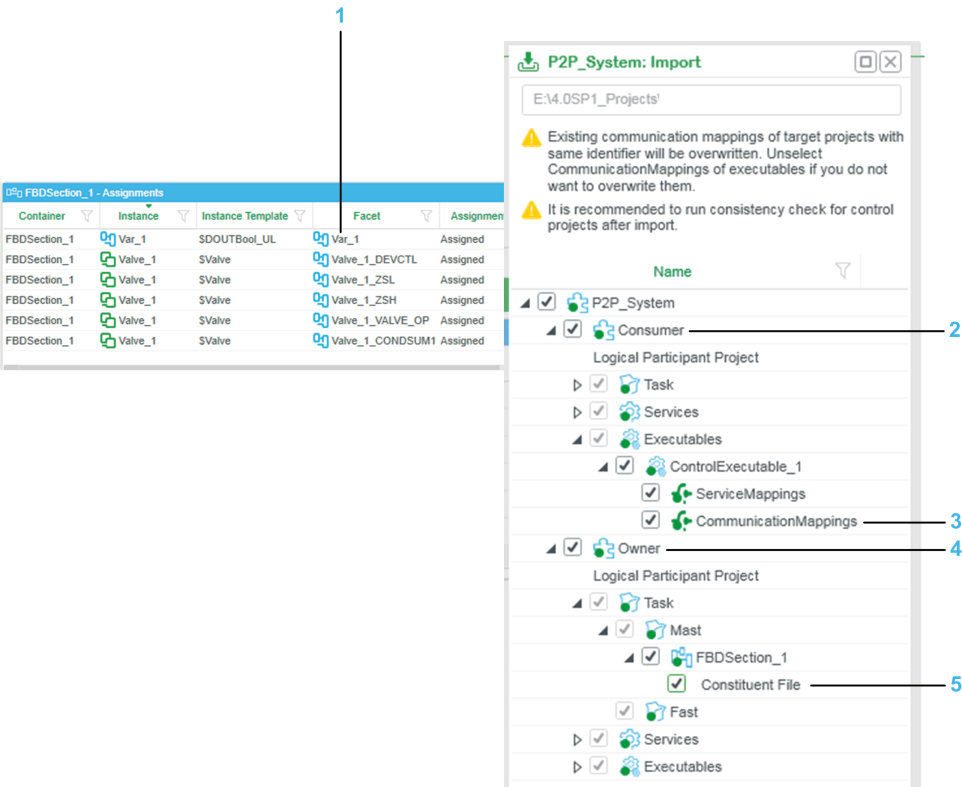
You import the export file in the **Projects Explorer** where *Owner* and *Consumer* Control exist. However, you have renamed *Var_1* in the *Owner* project.

Because you want to import communication mappings, you have selected both Control projects in the **Import** window.

Because *Var_1* had been created by a facet that is assigned, for example, to *FBDSection_1* in the *Owner* Control project, by selecting the constituent file, page 283 of *FBDSection_1* in the **Import** window, you are able to recreate the variable and import communication mappings.

NOTE: A network variable can be used only once per project, independently of the number of Control executables that exist in the project.

The following figure illustrates the import example by using a target system in which the *Owner* and *Consumer* projects exist and showing the objects that are selected for import.



Item	Description
1	FBDSection_1 of Owner Control project showing the facet that creates Var_1.
2	Consumer Control project.
3	Communication mappings of Consumer.
4	Owner Control project.
5	Constituent file of FBDSection_1 to which facet that creates Var_1 is assigned.

Import Rules for Data Backup Files

When you import a Control project and a Control project with the same identifier already exists in the target system, if you have selected to import the data backup file and a file with the following identical identification information, page 880 already exists, the file is not imported:

- Time stamp.
- Identifier of the executable.
- Identifier of the controller and role.
- Description.
- Type of data that is backed up.

NOTE: The notification panel displays information about the not imported data backup file.

Opening the Project Import Window

To open the **Import** window, proceed as follows.

Step	Action
1	<p>Right-click the system root folder in a Project Browser of the Project Explorer and select Import.</p> <p>Result: The Import dialog box opens.</p>
2	<p>Browse to the export file (.sbk) that contains the Control project information that you want to import and click Open.</p> <p>Result: The Import window opens and displays the contents of the file that you have selected once it has read the data contained in the file. This can take time depending on the number and size of projects contained in the file.</p> <p>NOTE: You can abort the command, page 292.</p>

Project Import Window

The **Import** window lets you select Control project information that is contained in an export file and that you can import into an existing system. You can select one or more entire Control Participant project and/or parts of them.

The following figure shows an example of the **Import** window displaying the contents of a Control project export file. In this example, the check box for *FBDSection_2* is cleared because the section cannot be imported. To indicate that at least one item of the Control project is not selected, the check box of parent nodes are filled with a square.



The table indicates the meaning of the colored dots that appear in the **Import** window next to items of a Control project.

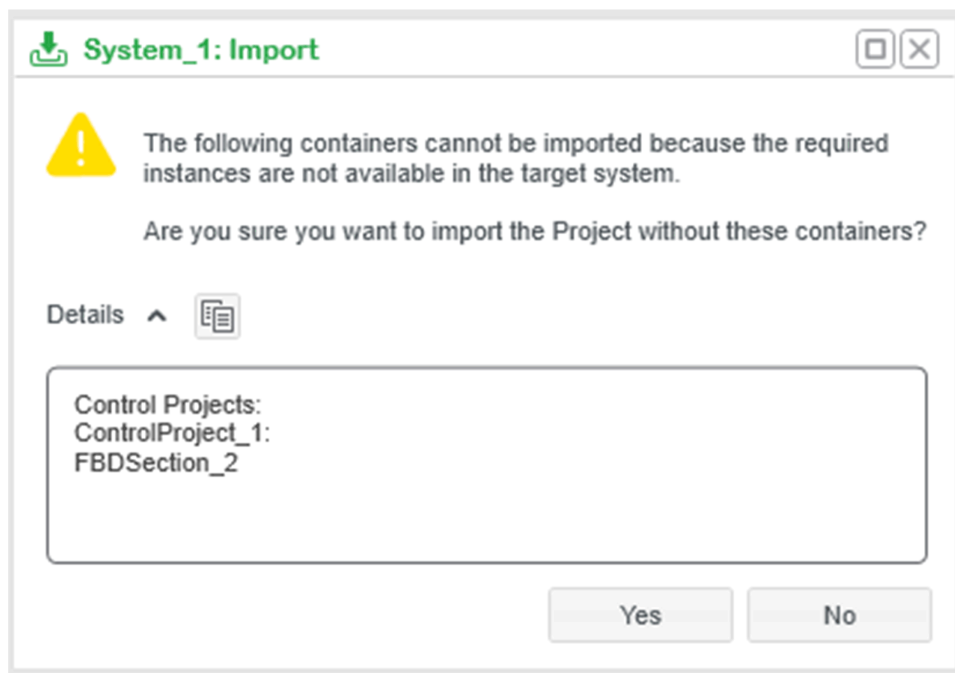
Dot color	Description
Green	You can import the item.
Red	<ul style="list-style-type: none"> At the section level: The section contains at least one facet that is referenced by an instance, which is not present in the application, page 282. At the executable level: A project-related topological object, page 282 is not present in the target system. <p>A tooltip provides additional information about the issue.</p> <p>NOTE: You cannot import items that are tagged with a red dot.</p>
Yellow	<p>The item contains at least one subitem that is tagged with a red dot.</p> <p>The item can be imported partially only.</p>

NOTE: To expand/collapse a node in the **Import** window, select the node and press **Enter**.

Confirmation Dialog Box For Sections That Cannot Be Imported

The following figure shows an example of the dialog box that opens when a section of a Control project selected for import is tagged with a red dot and you click **OK** in the **Import** window.

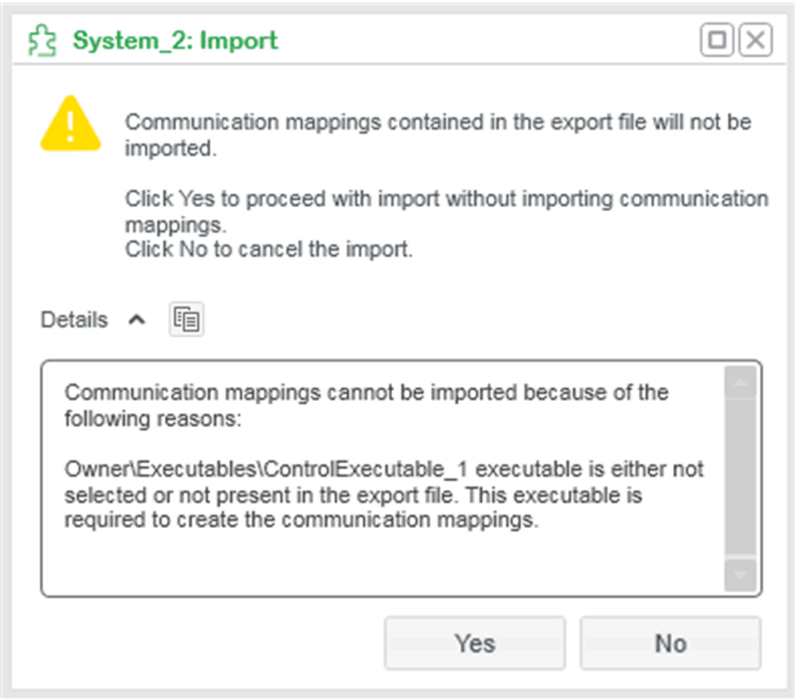
If you are importing also Supervision projects, containers are grouped by Participant then by project.



Button	Description
Yes	Closes the dialog box, the Import window, and imports the selected projects without sections tagged with a red dot.
No	Closes the dialog box and reverts to the Import window without importing projects. You can modify the selection of projects or cancel the import operation.

Confirmation Dialog Box For Communication Mappings That Cannot Be Imported

The following figure shows an example of the dialog box that opens when you click **OK** in the **Import** window and EcoStruxure Process Expert detects that communication mappings selected for import cannot be imported. It indicates the identifier of the project and the corresponding executable, and the reason. The **Import** window is closed before the dialog box opens.



Button	Description
Yes	Closes the dialog box and imports the selected projects without communication mappings. NOTE: This may result in variable mappings being present in the target system while the required variables do not exist.
No	Closes the dialog box and cancels the import operation.

Aborting Import Tasks

- The Abort icon is displayed in the notification panel, page 85 at two different stages of the import process:
- After clicking **Open** when you select the export file, page 289. Clicking the icon cancels the operation to read the contents of the export file. The **Import** window does not open.
 - After confirming the import by clicking **OK** in the **Import** window, the Abort icon is displayed. Clicking the icon cancels the task. The project is not imported.
- Only the user who selected the command is allowed to cancel it.

Importing a Control Project

To import data of a Control project into a system, proceed as follows.

Step	Action	Result
1	In the Import window, page 289, select the items that you want to import.	-
2	Click OK .	If applicable rules are satisfied, EcoStruxure Process Expert: <ul style="list-style-type: none"> • Applies the import rules based on your import selection. • Proceeds with the import. • Replaces existing communication mapping if you had selected to import them. • Displays an import summary dialog box.
3	In the Project Browser , right-click the Control project into which you have made the import and select Refine .	The Control Participant window opens.
4	Click the consistency check button, page 462.	A consistency check is performed and a notification about the findings is shown once the check is completed.
5	Take the necessary corrective actions, page 462.	-

NOTE: If EcoStruxure Process Expert detects incompatibilities during the import process, it cancels and rolls back the import operation, and displays a dialog box to inform you. In this case, take the necessary corrective actions and start the import over. For example, if the incompatibility is related to variable mappings, you can clear the check box next to **CommunicationMappings** to import Control project data but no communication mapping.

Importing a Control Expert Project

Overview

The **Import Control Project** command allows you to create a Control project in the **Project Explorer** by importing data of a project that was created by using Control Expert Classic and which is contained in a compatible project file (.stu or .xef format).

After import, you will have access to the same commands from the project node as if you had created the project in the **Project Browser**.

Project Requirements

To be able to import a project created with Control Expert, the following requirements need to be satisfied:

- The version of Control Expert Classic with which the export file was created needs to be identical to the version of the Control Participant. To determine the version of the Control Participant, open it (for example, by refining a Control project) and open the **About** dialog box from the **Help** menu.
- The version of the EcoStruxure Control Expert DTM Library that is installed in Control Expert needs to be the same as the one installed in the Control Participant. For information on the installed version, refer to the platform release notes.
- No application password is set for the Control project.

Imported Data

Project data contained in the file (stu or .xef)	Contained in the Control Participant project after import	Visible in the Project Explorer after import
FBD sections	Yes	Yes
Sections in the following programming languages: <ul style="list-style-type: none"> • Ladder Diagram (LD) • Instruction List (IL) • Structured Text (ST) • Sequential Function Chart (SFC) • Ladder Logic 984 (LL984) 	Yes	Sections are created in their respective Mast and Fast task folders in the Assignment Editor .
Constituents of any section	Yes	No
Configuration	No	Refine the imported project to view the information.
Other global project information (for example, variables, types, project settings...)	Yes	NOTE: Only data related to the program becomes part of the logical Control project. Other data (for example, data related to the configuration) is ignored during build although it is visible when you refine the logical project.
Project platform	Yes	Yes Controller Family attribute, page 270.

NOTE: The name of the file containing the project data becomes the identifier of the Control project that is created during the import process.

Importing a Control Expert Project

To import a Control Expert project into an existing system, proceed as follows.

Step	Action
1	Right-click the system root folder in the Control Project Browser of the Project Explorer and select Import Control Project . Result: The Open dialog box opens.
2	Select the file format that you want to import.
3	Browse to the location of the Control Expert project file that you want to import and click Open . Result: A Control project is created and displayed in the Control Project Browser .

Managing Data Backup Files

Overview

The **Manage Data Backup Files** dialog box lets you view and delete data backup files, page 880 that exist for all the executables of a Control Participant project.

The dialog box shows the information that was associated to each backup file at the time of creation, including the type of data that was selected.

By default, backup files are listed in descending order of time stamp (most recent one on top). You can sort and filter, page 120 the information that is displayed.

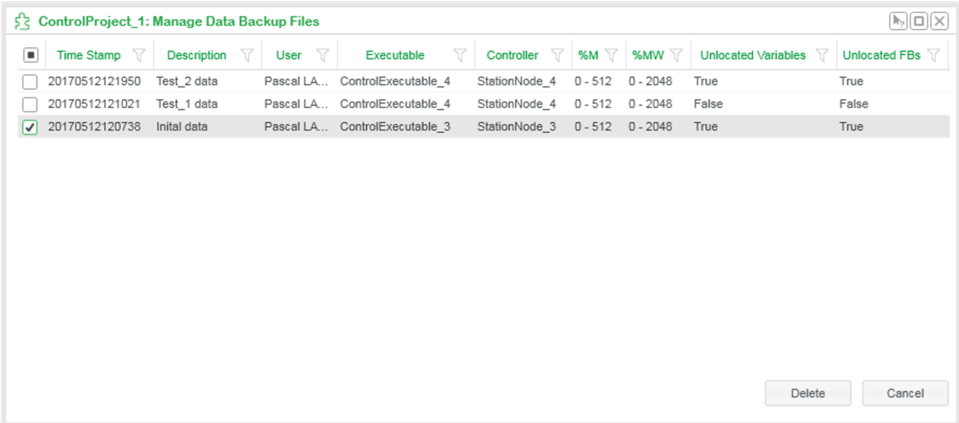
Opening the Manage Data Backup Files Dialog Box

To open the **Manage Data Backup Files** dialog box, right-click a Control Participant project in the **Project Browser** and select **Manage Data Backup Files**.

If no data backup file exists, the command is not available.

Manage Data Backup Files Dialog Box

The following figure shows an example of the **Manage Data Backup Files** dialog box, which shows the data backup files that have been created for executables of *ControlProject_1*.



Column header	Description
Time Stamp	Date and time of the backup file creation.
Description	Description that was entered when the backup file was created.
User	Name of the user who was logged on when the backup file was created.
Executable	Identifier of the Control executable whose data is contained in the backup file.
Controller	Identifier of the controller that was running the executable whose data was backed up. For redundant controllers, it also indicates the role of the controller that was selected when the backup file was created.
%M	Indicates the memory range of located variables of boolean data type that are contained in the backup file. If no such data was selected during data backup, the value shown is N/A .
%MW	Indicates the memory range of located variables of WORD data type that are contained in the backup file. If no such data was selected during data backup, the value shown is N/A .

Column header	Description
Unlocated Variables	True indicates that this type of data was selected during data backup. Possible values: <ul style="list-style-type: none">• True• False
Unlocated FBs	

Deleting Data Backup Files

To delete a data backup file, select the corresponding row and click **Delete**. The backup file is deleted after you confirm the command.

You can select more than one backup file at a time.

NOTE: When you delete a Control Participant project or the controller to which it is deployed, the corresponding data backup files are deleted as well.

Supervision Project Definition Stage

Overview

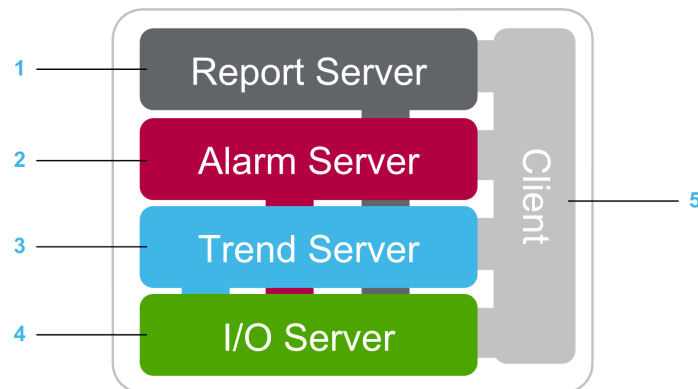
This section describes how to create Supervision projects of a system.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Internal Supervision Architecture

Task Description

As a reminder, the following figure describes the internal Supervision architecture for the execution of tasks:



1 Report scheduling and execution

2 Alarm acquisition and processing

3 Trend acquisition and storage

4 I/O communications

5 Graphical interface, events, local I/O, and client services

NOTE: Report, alarm, and trend servers also act as clients of the I/O server.

Creating Supervision Projects

Creating Supervision Projects

To create a Supervision project, in the **Supervision Project Browser**, right-click the system root folder and select **Create Supervision Project**.

Result: The Supervision is shown in the tree view.

NOTE: For each system, you can create as many Supervision projects as needed to meet the requirements of the system.

Supervision Project Actions

The table describes the Supervision project context menu commands.

Command	Description	Applies to
Create Cluster , page 305	Creates a new cluster with its default contents and displays it in the tree view.	Project Definition stage
Last Action Summary , page 339	Opens a dialog box that indicates the last action that was performed and a summary of the result.	
Assign Facets	Opens the Assignment Editor , which lets you create and manage tag containers and Supervision pages of the Supervision project and assign facets to them.	Assignment stage, page 340
Generate	Starts a process that creates and/or updates in the Supervision Participant project the data of the entire Supervision project.	Generation stage, page 396
Refine	Opens the Supervision Participant, which allows you to modify the project.	Refinement stage, page 508
Advanced Settings , page 509	Opens the Supervision Participant, which allows you to change advanced project settings that will be activated on the workstations to which you deploy the Supervision project.	Refinement stage
Include Projects , page 510	Opens a dialog box, which allows you to: <ul style="list-style-type: none"> Add external included projects to be deployed to the Supervision nodes. To remove included projects that are attached to the Supervision and stored in the contents repository. NOTE: You cannot add an included project if EcoStruxure Process Expert already added to the Participant project an included project with the same name.	
Attach User Files , page 511	Opens a dialog box, which allows you to: <ul style="list-style-type: none"> Attach external files to be deployed to the Supervision nodes. To remove files that are attached to the Supervision project and stored in the contents repository. 	
Export , page 318	Opens a Save As dialog box, which allows you to create an export file containing the Supervision project information.	Project Definition stage
Settings	Opens the Settings window, page 272, which lets you configure various project-related settings.	Assignment stage
Delete	Deletes the Supervision project including components, assignments, and mappings.	–
Rename	Allows you to enter a new identifier for the project.	–
Properties	Opens the Properties window, which allows you to edit the properties of the Supervision project.	–

Containers Node Actions

The table describes the **Containers** node context menu commands.

Command	Description	Applies to
Assign Facets	Opens the Assignment Editor , which lets you create and manage tag containers and Supervision pages of all clusters and assign facets to them.	Assignment stage, page 340
Create Tag Container	Lets you create a tag container, page 353 in an existing cluster. The Assignment Editor opens in the background.	
Create Page	Opens the Assignment Editor and creates a Supervision page, page 358 in the project.	
Generate	Starts a process to create and/or update in the Supervision Participant project the data for the containers of all clusters that exist in the project.	Generation stage, page 396

Pages Node Actions

The table describes the **Pages** node context menu commands.

Command	Description	Applies to
Assign Facets	Opens the Assignment Editor , which lets you create and manage tag containers and Supervision pages of all clusters and assign facets to them.	Assignment stage, page 340
Create Page	Opens the Assignment Editor and creates a Supervision page in the project.	Assignment stage, page 341
Generate	Starts a process to create and/or update in the Supervision Participant project the data for the pages that exist in the project.	Generation Stage, page 396

Cluster Node Actions

For a description of the context menu commands, refer to the topic describing how to create clusters, page 305.

Tags Node Actions

The table describes the **Tags** node context menu commands.

Command	Description	Applies to
Assign Facets	Opens the Assignment Editor , which lets you create and manage tag containers and Supervision pages of the parent cluster and assign facets to them.	Assignment stage, page 340
Generate	Starts the generation process of tag containers and pages that exist in the parent cluster to populate new tags, update out-of-date tags, and/or remove deleted tags in the Supervision Participant project.	Generation stage, page 396

Services Node Actions

The table describes some of the **Services** node context menu commands. For a description of the other commands, refer to the topic describing how to create Supervision servers and clients, page 306.

Command	Description	Applies to
Open	Opens the Services window, which lets you view and edit properties of the Alarm, report, and trend servers as well as clients that exist in the parent cluster.	Assignment stage, page 340
Generate	Starts a process to create and/or update in the Supervision Participant project the data related to services.	Generation stage, page 396

ClientEvents and ServerEvents Node Actions

The table describes the **ClientEvents** and **ServerEvents** nodes context menu commands.

Command	Description	Applies to
Open	Opens the corresponding window, which lets you view the Supervision facets that are assigned to client or server events in the parent cluster.	Assignment stage, page 340
Generate	Starts a process to create and/or update in the Supervision Participant project the data related to services in the parent cluster.	Generation stage, page 396

Executables Node Actions

The table describes the **Executables** node context menu commands.

Command	Description	Applies to
Create Executable	Creates one or more Supervision executables, page 764, which let you associate the Supervision project infrastructure to workstations and the I/O devices to controllers.	Mapping stage

Supervision Project Properties

Right-click the Supervision project and select **Properties**.

You can edit the following items.

Item		Description
General	Identifier	The identifier of the Supervision project, which must be unique within the system.
	Description	Optional. You can enter a description of the project with free-form text.

Project Settings Window

The **Settings** window opens when you select the **Settings** command in the context menu of the Supervision project.

The following table describes the settings that are specific to Supervision projects. For a description of the other settings, refer to the topic describing [project settings](#), page 272.

Section	Description
Page Templates	<p>Lets you add and remove Supervision page templates and select a template to be used by default when you create pages.</p> <p>A page template is the combination of a style, a resolution, and a template.</p> <p>Some page templates are provided by default.</p> <p>The page templates that appear in this section and your selection apply only to this Supervision project.</p> <p>For details on how to add, remove, and use page templates, refer to the topic describing the selection of Supervision page templates, page 359.</p>

Creating Supervision Project Components

Overview

This section describes how to use the **Project Explorer** to create the components of Supervision projects.

During build, one or more OPC Factory Server and/or OPC UA Server Expert configuration files (.xml) are automatically generated, [page 788](#) based on the configuration of properties of the Supervision project components.

For information on how to create tag containers and Supervision pages, refer to the topic describing the [creation of containers](#), [page 341](#) in the Assignment stage.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Creating Clusters

Creating Clusters

To create a cluster, right-click the Supervision project and select **Create Cluster**.

Result: A new cluster node and its child nodes are created.

Cluster Actions

The table describes the cluster context menu commands.

Command	Description
Generate	Starts the generation process to populate the database of the Supervision Participant project for this cluster.
Delete	Deletes the cluster including its contents and unassigns the assigned Supervision data facets.
Rename	Lets you enter a new identifier for the cluster.
Properties	Opens the Properties window, which allows you to edit the properties of the cluster.

Cluster Properties

To open the **Properties** window of the cluster, right-click the cluster and select **Properties**.

You can edit the following items,

Item		Description
General	Identifier	The identifier must be unique within the Supervision project and satisfy applicable naming rules, page 32. NOTE: Changing the identifier of the cluster changes the assignment status of facets that are assigned to a tag container of the cluster to Out Of Date .
	Description	Optional. You can enter a description of the cluster with free form text.

Creating Supervision Servers and Clients

Overview

You can create Supervision servers in the **Services** node of a cluster.

The I/O server instance represents a server that is set up on an operation server (see EcoStruxure Process Expert, Installation and Configuration Guide), on which Supervision and OPC Factory Server or OPC UA Server Expert software is installed. The operation server is represented in the topology of the system by a workstation, page 597, which acts as an engine to run the Supervision project.

The alarm, trend, and report server instances model operation servers that are installed on a single or distinct server computers.

You can create up to two alarm, report, and trend servers per cluster.

NOTE: Generate the Supervision project after you make changes related to components described in this topic.

Creating Supervision Servers

To create I/O, alarm, trend, and report servers in the cluster, right-click the **Services** node and select the corresponding command.

NOTE: Only I/O servers are shown in the project browser. To view other servers, open the **Services** node.

NOTE: EcoStruxure Process Expert may automatically create servers during the Assignment stage if you have not created them already. For more information, refer to the Supervision project Assignment stage, page 377.

Supervision I/O Server Properties

To open the **Properties** window of an I/O server, right-click the server and select **Properties**.

You can edit the following items.

Item		Description
General	Identifier	The identifier must be unique within the Supervision project.
	Description	Optional. You can enter a description of the server with free-form text.
Attributes	Port	Communication port the server listens on. Default value: 2082.
	Legacy Port	Communication port that is used for legacy connections to the server. Default value: 2078.
	Allow RPC	Determines if the I/O server accepts remote <i>MsgRPC</i> and <i>ServerRPC</i> calls. Default value: False. NOTE: If you enable Allow RPC (True), ensure that the connection between the remote computer and the I/O server is encrypted. For details, refer to <i>Secure the Connection from a Remote CtAPI Application</i> in the Supervision Participant Help.

Supervision Servers Common Properties

To view the alarm, trend, and report servers that exist in the cluster and their properties, right-click the **Services** node and select **Open**.

You can view and/or edit the following items.

Item		Description
General	Identifier	<p>The identifier must be unique within the Supervision project.</p> <p>Default value: <i>Server_n_P</i> where:</p> <ul style="list-style-type: none"> <i>Server</i> is the name of the service provided by the server (for example, <i>Alarm</i>). <i>n</i> is an incremental number starting at 1. <i>P</i> stands for primary. The suffix of the second server instance that you create is <i>S</i> for standby, even if you rename the first instance.
	Description	<p>Optional.</p> <p>You can enter a description of the server with free-form text.</p>
Attributes	Port	<p>Communication port the server listens on.</p> <p>Default value:</p> <ul style="list-style-type: none"> Alarm server: 2080 Report server: 2084 Trend server: 2085 <p>Edit the value if you are running more than one server on the workstation.</p>
	Legacy Port	<p>Communication port that is used for legacy connections to the server.</p> <p>Default value:</p> <ul style="list-style-type: none"> Alarm server: 2076 Report server: 2075 Trend server: 2077
	Database Port	<p>The attribute is available for alarm servers only.</p> <p>Communication port the server database listens on.</p> <p>Default value: 5482</p> <p>Edit the value if you are running more than one alarm server on the workstation.</p>
	Mode	<p>You can select one of the following values from the menu:</p> <ul style="list-style-type: none"> None Primary: Defines the server as the primary server. Default value for the server with <i>P</i> suffix. Standby: Defines the server as the standby server. Default value for the server with <i>S</i> suffix.
	Extended Memory	<p>The attribute is available for alarm servers only.</p> <p>Possible values:</p> <ul style="list-style-type: none"> False: Default value. True: The alarm server operates in 64-bit extended memory mode. This allows the alarm server process to utilize memory beyond the 4 GB limit. <p>For more information on this mode of operation, refer to <i>Alarm Server Process</i> in the help of the Supervision Participant.</p>
	Allow RPC	<p>Determines if the server accepts remote <i>MsgRPC</i> and <i>ServerRPC</i> calls.</p> <p>Default value: False.</p> <p>NOTE: If you enable Allow RPC (True), ensure that the connection between the remote computer and the server is encrypted. For details, refer to <i>Secure the Connection from a Remote CtAPI Application</i> in the Supervision Participant Help.</p>

Creating Supervision Clients

You can create Supervision Clients in the **Services** node of a cluster.

The client instance can represent a Supervision client (runtime) that runs:

- On the operation server.
- On an operator station.

To create a client for the Supervision project, right-click the **Services** node and select **Create Client**.

Result: The client is created but not shown in the project browser.

Supervision Client Properties

To view the clients that exist in the cluster and their properties, right-click the **Services** node and select **Open**.

You can edit the following items.

Item		Description
General	Identifier	The identifier must be unique within the Supervision project.
	Description	Optional. You can enter a description of the client with free-form text.

Creating I/O Devices

Overview

Typically, an I/O device is created for one controller. If the data is present in two controllers, create an additional I/O device for the second controller manually. Create a second [tag container](#), [page 353](#) for data of the second controller and associate this tag container to the second I/O device.

EcoStruxure Process Expert creates additional I/O devices, [page 310](#) when you create calculated or disk variable tags.

NOTE: Monitoring data of a device, [page 984](#) communicating by using the Modbus TCP protocol (or Modbus through an Ethernet gateway) from the Supervision infrastructure also requires a dedicated I/O device per Modbus device.

Prerequisites

An I/O server must exist in the cluster.

Creating I/O Devices

To create an I/O device in the cluster, expand the **Services** node and the I/O server, right-click the **IODevices** node, and select **Create IO Device**. The I/O device is created but not shown in the project browser.

NOTE: EcoStruxure Process Expert automatically creates the I/O device during the assignment stage if you have not created it already. For more information, refer to the Supervision project [Assignment stage](#), [page 377](#).

I/O Device Properties

To view the I/O devices that exist in the cluster and their properties, right-click the **IODevices** node and select **Open**.

You can edit the following items.

Item		Description
General	Identifier	The identifier must be unique within the Supervision project. NOTE: Changing the identifier of the I/O device changes the assignment status of facets that are assigned to the tag container that is associated with the I/O device to Out Of Date if they were generated.
	Description	Optional. You can enter a description of the I/O device with free form text.
Attributes	Address	The value needs to be compatible with the protocol that is selected for the associated tag container. The default value is the alias name for OPC Factory Server: <i>Alias_<IODevice_n></i> where <i><IODevice_n></i> corresponds to the value of Identifier . NOTE: The attribute is disabled and cleared when Protocol of the associated tag container is set to <i>OPCUA</i> or <i>OPCUA Embedded</i> . For these two protocols, the following temporary address is created for the I/O device when you generate the Supervision project or the cluster: <ul style="list-style-type: none"> For <i>OPCUA</i>: <i>opc.tcp://127.0.0.x:49152/OPCUAServerExpert</i> For <i>OPCUA Embedded</i>: <i>opc.tcp://127.0.0.x:4840</i> During build, page 788, the IP address placeholder (127.0.0.x) and the port value are updated based on the mapping.
	StartupMode	Select the role of the I/O device in terms of communication with the controller: <ul style="list-style-type: none"> <i>Primary</i> (default value) <i>Standby</i> <i>StandbyWrite</i> <i>None</i>
	Priority	You can assign the order in which the I/O device attempts to communicate with the tag container that it is associated with in case more than one I/O device is associated with the container. Set the values as follows depending on the role of the I/O device: <ul style="list-style-type: none"> Primary: 1 or blank (default value) Secondary: 2 or higher NOTE: If you associate two I/O devices configured as primary to one tag container, verify that Priority is configured with a different value for each.
	Memory	Select one of the following values depending on how you want the I/O device to communicate with the controller: <i>False</i> : Default value. The I/O device is online with the controller <i>True</i> : The I/O device is not connected to the controller. Select this setting for testing purposes, for example.
Relationships	TagContainer	Select the tag container that is associated with the I/O device. NOTE: <ul style="list-style-type: none"> By default, EcoStruxure Process Expert associates the I/O device with the tag container for which the I/O device has been automatically created, page 377 during the assignment of Supervision data facets. Changing the relationship of the I/O device to the tag container changes the assignment status of facets assigned to the tag containers affected by this change to Out Of Date.
	Ports	Select the communication port, page 312 that is associated to the I/O device. NOTE: By default, EcoStruxure Process Expert associates the I/O device with either the communication port that has been automatically created, page 377 during the assignment of Supervision data facets or with the one that you have created manually.

I/O Devices Created for Calculated and Disk Variable Tags

When you generate a Supervision data facet that references a non-disabled *Calculated Variable Tag* or *Disk Variable Tag* element (see EcoStruxure Process Expert, Global Templates, Reference Manual), one specific I/O device is automatically created per I/O server with the same properties as the default I/O device except for the following. These specific I/O devices are only visible when you refine the Supervision project.

Property	For a calculated variable tag	For a disk variable tag
Identifier	<i>CicodeIODevice_n</i> where <i>n</i> is an integer starting at 1 so that the identifier remains unique within the I/O sever.	<i>DiskIODeviceID_n</i> where: <ul style="list-style-type: none">• <i>IODeviceID</i> is the identifier of the I/O device that is associated to the tag container to which the data facet is assigned.• <i>n</i> is an optional integer starting at 1 that is added so that the identifier remains unique within the I/O sever.
Address	Blank	Blank
Ports	Blank	Blank
Memory	Same value as default I/O device	<i>True</i>
Persist		<i>True</i>

Creating Communication Ports

Overview

Before you can create a communication port, you need to create a communication board unless one already exists.

You need to create a communication board for each communication driver that you are using. For example, if you are using both OFSOPC and OPCUA, create one board for each and configure them accordingly.

To view the communication ports that exist, right-click a board and select **Open**.

Prerequisites

An I/O server must exist in the cluster.

Creating a Communication Board

To create a communication board, expand the **Services** node and the I/O server, right-click the **Communications** node, and select **Create Board**.

NOTE: EcoStruxure Process Expert automatically creates one communication board and port during the assignment stage if you have not created it already. For more information, refer to the *Supervision Assignment stage*, page 377.

NOTE: Deleting a communication board deletes the associated communication ports.

Communication Board Properties

To open the **Properties** window of the communication board, right-click it, and select **Properties**.

You can edit the following items. For details, refer to *Board Properties* in the help of the Supervision Participant.

Item		Description
General	Identifier	The identifier must be unique within the I/O server. 16 characters maximum.
	Description	Optional. You can enter a description of the board with free-form text.
Attributes	Protocol	Indicates the protocol for communicating with the I/O device and that is to be used for configuring the I/O device. When the protocol for the tag container is set to <i>OPCUA Embedded</i> , select <i>OPCUA</i> . When you configure a user-defined protocol for a tag container, page 353, enter a protocol for the board that is compatible and installed in the Supervision Participant. To verify which ones are installed, refine the Supervision Participant project and expand an entry in the Board Type column of Components & Mapping in the Topology activity. Default value: <ul style="list-style-type: none"> Same as Protocol (that is, <i>OFSOPC</i> or <i>OPCUA</i>) if the value of Protocol for all the tag containers that are associated to an I/O device whose communication port is associated to the board is the same; otherwise, <i>OFSOPC</i>. <i>OFSOPC</i> when you create the board manually.
	Address	Default value: 0 when the board is automatically created.

NOTE: The *OFSOPC* driver configuration implements an automatic and adaptive strategy, creating and deleting OPC groups and OPC items dynamically to maximize throughput. For information on other parameters affecting performance when using this driver configuration, refer to *Performance Considerations* (see EcoStruxure Process Expert, Supervision Participant Services, User Guide).

NOTE: For the *OPCUA* driver, EcoStruxure Process Expert adds driver-specific parameters to the *citect.ini* file (see EcoStruxure Process Expert, Supervision Participant Services, User Guide) (for example, subscription and connection parameters), which are configured with default values and apply to a given I/O device. For details, refer to *OPCUA Driver* in the *Driver Reference Help* of the Supervision Participant.

Creating Communication Ports

For each communication board, you can create several communication ports.

To create a communication port, expand the **Services** node, right-click the board, and select **Create Port**.

Result: The port is created but not shown in the project browser.

Communication Port Properties

To view the ports that exist for a board and their properties, right-click the board and select **Open**.

You can edit the following items.

Item		Description
General	Identifier	The identifier must be unique within the I/O server.
	Description	Optional. You can enter a description of the port with free-form text.
Attributes	Port Number	Interrupt number of the port. Leave it blank if you are using the COM port of the computer.
	Special Opt	Lets you enter any special options that are supported by the board. 32 characters maximum. For details, refer to <i>Hardware Arrangements</i> in the help of the Supervision Participant.
Relation-ships	IoDevice	Select the I/O devices that are associated with the communication port from the menu. The I/O device, page 309 must exist in the Supervision project cluster to be available for selection. NOTE: By default, EcoStruxure Process Expert associates the port with the I/O device that has been automatically created, page 377 during the assignment of Supervision data facets. If you have created the I/O device manually, associate it with the port.

Implementing Redundancy

Overview

The Supervision Participant allows you to build redundancy into the system at different levels to help increase the degree of protection and/or reliability. The common ways to implement redundancy are:

- I/O server redundancy
- Alarm, trend, and report server redundancy
- Network redundancy

The following topics describe the additional steps that you need to complete in order to implement such redundancy by starting from a typical Supervision project, which consists of one of each:

- Cluster
- Tag container
- I/O server with its client
- I/O device
- Communication board
- Communication port

For more information, refer to the topic of the Supervision Participant help, page 95, which describes how to build redundancy into a system.

I/O Server Redundancy

To implement I/O server redundancy, proceed as follows.

Step	Action	Refer to
1	From the Services node of the Supervision project, create a second I/O server.	Project Definition stage
2	From the Services node, create a client for this second I/O server.	
3	From the Communications node of the second I/O server, create a communication board.	
4	From this board, create a communication port.	
5	From the IODevicees node of the second I/O server, create an I/O device.	
6	In the properties of the second I/O device, associate the I/O device with the tag container of the Supervision project, which is already associated to the first I/O device. Result: The StartupMode of the second I/O device is automatically configured as <i>Standby</i> with Priority 2 (or the next available value).	
7	In the same window, configure the Ports property as <i>Port_2</i> , which is the port of the communication board that you have created in step 4.	
8	In the properties of the communication port of the second I/O server, verify that the port is associated to the standby I/O device under IODevice .	
9	In the properties of the communication board of the second I/O server, configure Protocol with the same values as for the tag container, which is associated to the first I/O device.	
10	From the system root folder, create a workstation modeling the second I/O server.	Configuration stage
11	In the properties of the NIC of this workstation, configure the IP address of the NIC.	
12	On this workstation, create one Supervision and one OFS service instances.	
13	Connect this workstation to the logical network.	

Step	Action	Refer to
14	Once you have generated the complete Supervision project, from the project executable, perform the service mapping of the second I/O server to the workstation that you have created in step 10.	Mapping stage
15	From the project executable, map the standby I/O device to the same controller to which you have mapped the primary I/O device (communication mapping).	
16	Configure the physical computers according to the topology of the system, build, and deploy the Supervision project.	Build and Deployment stage

Alarm, Trend, or Report Server Redundancy

To implement alarm, trend and/or report server redundancy, proceed as follows.

Step	Action	Refer to
1	From the Services node of the Supervision project, create a second alarm, trend and/or report server. Result: <ul style="list-style-type: none"> The server with the _s suffix in its identifier is created. The server Mode is configured as <i>Standby</i>. 	Project Definition stage
2	From the system root folder, create a workstation modeling each redundant alarm, trend and/or report server.	Configuration stage
3	In the Properties window of the NIC of each redundant workstation, configure the IP address of each NIC.	
4	On each redundant workstation, create one Supervision service instance.	
5	Connect each redundant workstation to the logical network.	
6	Once you have generated the complete Supervision project, from the project executable, map each standby server to the corresponding redundant workstation (service mapping).	Mapping stage
7	Configure the physical computers according to the topology of the system, build, and deploy the Supervision project.	Build and Deployment stage

Network Redundancy

Network redundancy can be implemented for any Supervision station.

To implement network redundancy, for example, on the operation server, proceed as follows.

Step	Action	Refer to
1	In the workstation modeling the operation server (I/O server), create a second NIC.	Configuration stage
2	In the Properties window of the second NIC, configure its IP address. NOTE: It is not required to connect this NIC to a logical network.	
3	Configure the physical computers according to the topology of the system.	—
4	Build the Supervision project by using the Build All command. NOTE: In the service mapping, no additional mapping is required as the engine that is mapped to the <i>IOServer</i> service is the workstation.	Build and Mapping stages
5	Deploy the Supervision project either from the executable or from the workstation by selecting either NIC. Result: After restoring the Supervision project on the workstation (operation server), both network addresses are created in the Supervision Participant and associated to the I/O server.	Deployment and Execution stages

NOTE: No specific action is required in the Supervision project.

Managing Supervision Projects

Overview

This section describes how to use the **Project Explorer** to manage Supervision projects and project contents.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Exporting a Supervision Project

Overview

The export functionality lets you create a non-editable export file (.sbk), which contains selected Supervision project information.

You can use this data with the Supervision project [import](#), [page 280](#) functionality to transfer Supervision project information to a different system or merge information from different Supervision projects.

NOTE: To export all the projects (Control and Supervision) of a system at once in one export file, right-click the root folder in either **Project Browser**, select **Export**, and follow the export procedure described in this section. Refer also to the topic describing the export and import of [Control projects](#), [page 274](#).

Software Version Compatibility

The version of EcoStruxure Process Expert where you import an export file must be the same as the version that was used to create the export file (a service pack is considered the same version); otherwise, you cannot import the file.

Project Related Data

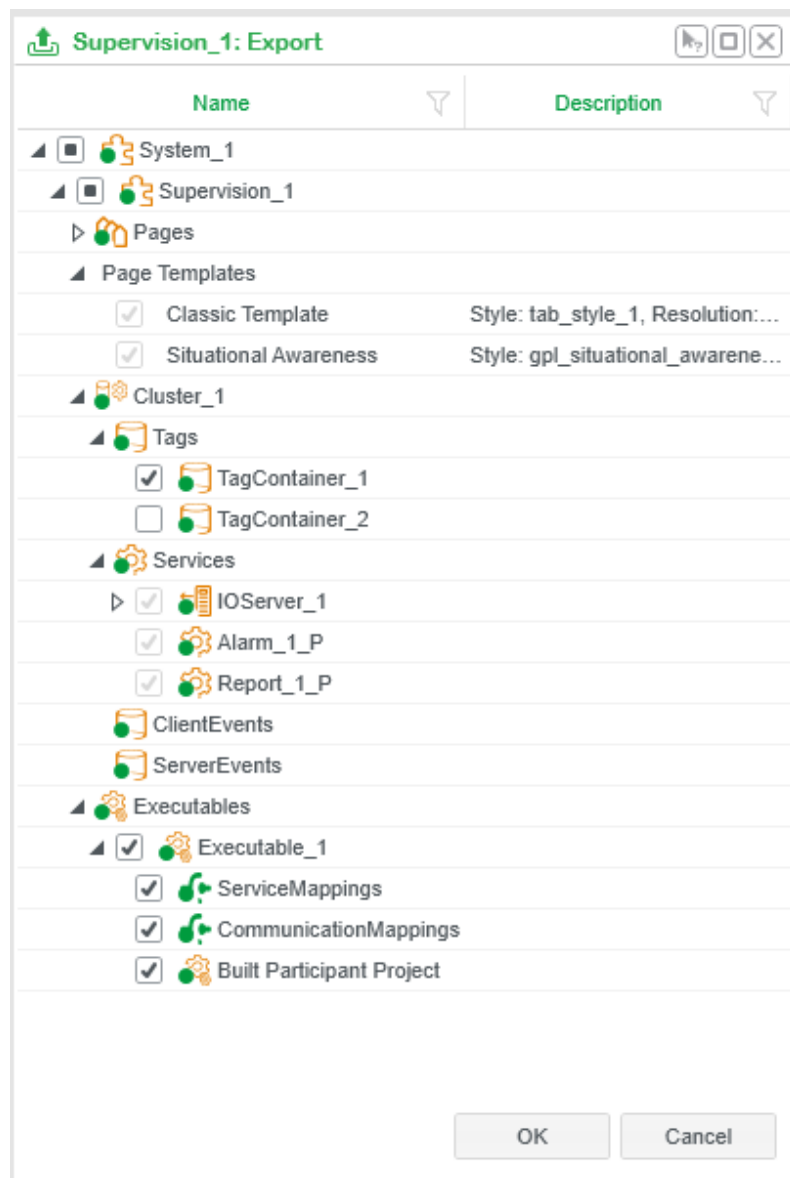
To be able to use the export file for importing a project into a different system, verify that the instances, their templates, and topological entities related to the project exist in the target system [before importing the project](#), [page 326](#). Use the export-import functionality, [page 647](#) of the **Topology Explorer** first.

NOTE: The parameter values of instances that reference the exported facets are not contained in the export file. However, the export file contains information that allows to determine during import if the configuration of an instance is different from when its facets were exported.

Project Export Window

The Supervision project **Export** window lets you select, which information the export file will contain.

The following figure shows an example of the **Export** window where the check box for *TagContainer_2* has been cleared. To indicate that at least one item of the Supervision project is not selected, the check box of parent nodes are filled with a square.



The table indicates the meaning of the colored dots that appear in the **Export** window next to items of a Supervision project.

Dot color	Description
Green	You can export the item.
Red	<p>You cannot export the item because it contains at least one facet with the following state:</p> <ul style="list-style-type: none">• Deleted• Unassigned• Out of Date• Moved <p>Generate containers that contain facets with such states or clear their check box in the Export window in order to export the Supervision project.</p> <p>NOTE: You cannot export items that are tagged with a red dot.</p>
Yellow	<p>Indicates that the item contains at least one subitem that is tagged with a red dot.</p> <p>The item can be exported partially only.</p>

NOTE: To expand/collapse a node in the **Export** window, select the node and press **Enter**.

Export Settings

The table describes which Supervision project information you can export. The information is exported by default.

Item		Description			
SupervisionProject_n		Unselect the item to exclude the entire project from the export file. When you select the project, the associated equipment hierarchy, page 159 is exported.			
Pages node		You cannot unselect the item.			
	Pages	Selecting a page automatically selects any genie facets that are assigned to it. Selecting a page automatically selects the logical Participant project.			
Page Templates		Default and user-created page templates, page 359. Unselecting a page template does not impact imported pages that use this template. You cannot unselect the default page templates.			
Cluster_n		You cannot unselect the item.			
	Tags node		You cannot unselect the item.		
		Tag containers	You can select tag containers individually. Selecting a tag container automatically selects any facets that are assigned to it.		
	Services node		You cannot unselect the item.		
		Alarm servers			
		I/O servers			
			Communications node		Any associated boards and ports are selected by default. You cannot unselect the item.
			IODevicees node		Any associated I/O devices are selected by default. You cannot unselect the item.
		Trend servers		You cannot unselect the item.	
		Report servers			
		Clients			
		ServerEvents			
	ClientEvents				
	Executables				
		Executable_n		You can select each executable and its contents individually. Selecting an executable automatically selects any associated mapping.	
			Service mappings		Contains data related to the service mapping including any required contents in the contents repository.
			Communication mappings		Contains data related to the communication mapping.
			Built Participant Project		Represents the Supervision Participant file (.ctz) (includes the OFS/OPC UA configuration files (.xml)). It appears and is selected only if its build status is Built or Out Of Date .

Aborting Export Tasks

After confirming the export by clicking **Save** in the **Export** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The export file is not created.

Only the user who selected the command is allowed to cancel it.

Exporting a Supervision Project

To export a Supervision project to file, proceed as follows.

Step	Action
1	<p>In the Supervision Project Browser, right-click the Supervision project that you want to export and select Export.</p> <p>Result: The Export window opens and displays the Supervision project that you have selected.</p>
2	<p>Select the project components that you want to export.</p>
3	<p>Click OK.</p> <p>Result: The Export window closes and the Export dialog box opens.</p>
4	<p>Select a location where you want to save the export file, enter a file name, and click Save.</p> <p>Result: The export file containing the selected Supervision project information is created.</p> <p>NOTE: If you click Cancel, the Export dialog box closes without creating the export file and the selection you have made in the Export window is discarded.</p>

Importing a Supervision Project

Overview

The **Import** command lets you import data of a Supervision project from an export file (.sbk) that was created by using the [export, page 276](#) functionality of the **Project Explorer**.

If the system where you are importing a Supervision project already contains a Supervision project with the same identifier, EcoStruxure Process Expert compares the data of both projects. Then, it proceeds with the partial import according to the import settings that you select and the import rules that are described in this topic.

NOTE: If the export file also contains information of a Control project, you can see the data that is related to the Control project and you can [import it, page 280](#) as well.

Software Version Compatibility

The version of EcoStruxure Process Expert where you import an export file must be the same as the version that was used to create the export file (a service pack is considered the same version); otherwise, you cannot import the file.

Import Settings

The table describes the Supervision project information you can import into a system. If the information is contained in the export file, it is imported unless mentioned otherwise.

Item		Description		
SupervisionProject_n		Unselect the item to exclude the entire project from the export file.		
Pages node		You cannot unselect the item.		
	Pages	<p>You can select containers (pages) individually⁽²⁾.</p> <p>Selecting a page automatically selects any genie facets that are assigned to it.</p> <p>Selecting a page automatically selects the logical Participant project.</p>		
Page Templates		<p>Default and user-created page templates, page 359.</p> <p>Unselecting a page template does not impact imported pages that use this template.</p> <p>You cannot unselect the default page templates.</p>		
Cluster_n		<p>At least one cluster must be selected.</p> <p>You cannot unselect a cluster if the export file contains only one cluster.</p>		
	Tags node		You cannot unselect the item.	
		Tag containers	<p>You can select tag containers individually⁽²⁾.</p> <p>Selecting a tag container automatically selects any facets that are assigned to it.</p> <p>A Constituent File is associated to each tag container.</p>	
		Tag container Constituent File	<p>If the identifier of the imported tag container already exists in the target project, the selection of the constituent file has an impact on the import behavior, page 327.</p> <p>NOTE: The Constituent File check box is selected and disabled if the identifier of the imported tag container does not exist in the target project.</p>	
	Services node		You cannot unselect the item.	
		Alarm servers		
		IOServer_n		
				Communications node
	IODevices node			<p>Any associated I/O devices are selected by default.</p> <p>You cannot unselect the item.</p>
		Trend servers		You cannot unselect the item.
		Report servers		
		Clients		
		ClientEvents		
	ServerEvents			
		Executables		

Item			Description
		Executable_n ⁽¹⁾	If the target system already contains an executable with the same identifier, you cannot select the executable for import. Else, you can select each executable and its contents individually. Selecting an executable automatically selects any associated mapping.
		Service mappings ⁽¹⁾	If a service mapping already exists in the target system, you cannot select the service mapping for import. Contains data related to the service mapping including any required contents in the content repository.
		Communication mappings ⁽¹⁾	If a communication mapping already exists in the target system, you cannot select the communication mapping for import. Contains data related to the communication mapping.
		Built Participant Project	Represents the built Supervision Participant file (.ctz) (includes the OFS/OPC UA configuration files (.xml)). It appears and is selected only if the built Participant project was selected at the time of export.

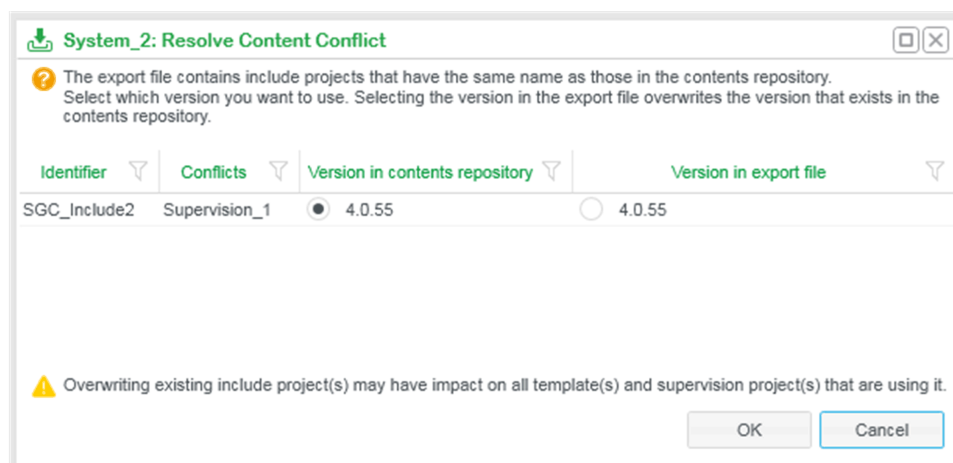
(1) Selected by default or not selectable for import.

(2) The item is available for selection only if it is tagged with a green dot; otherwise the item cannot be selected.

Import Rules for Included Projects

At the time of import, EcoStruxure Process Expert verifies whether the content repository of the target system contains an **Included** project with the same identifier as the one contained in the export file.

The following figure shows an example of the **Resolve Content Conflict** dialog box that opens if an **Included** project with the same identifier is detected.



Header	Description
Identifier	The identifier of the Included project for which a conflict exists.
Conflicts	The identifier of the Supervision Participant projects that exist in the target system and use the version of the Included project that is in the content repository.
Version in contents repository	<p>By selecting this check box, you use the version of the Included project that is in the content repository.</p> <p>Selected by default.</p>
Version in export file	By selecting this check box, you use the version of the Included project that is contained in the export file, page 332 and overwrite the version that is in the repository.

Included projects contain data related to Supervision constituents (for example, Genies and tags) that can be shared by several templates.

Replacing the existing **Included** project has an impact on:

- The templates that use the contents of the **Included** project.
- The Supervision Participant projects that are associated to the **Included** project.

NOTICE

LOSS OF DATA

Resolve an **Included** project conflict that arises during the import of a Supervision Participant project by selecting the appropriate version of the **Included** project to be used.

Failure to follow these instructions can result in equipment damage.

Import Rules for Project-Related Objects

Facets assigned to Supervision projects are related to instances of the application and mappings of project executables are related to entities of the topology. For the Supervision project import to complete, these project-related objects must be present in the system where you want to import the project.

The table describes the impact on the import operation if project-related objects are not present or have a different configuration. It also describes the corrective action.

Project-related object that is not present or different	Impact if not present in the target system	Required action
The instances whose facets are assigned to the Supervision project that you are importing (same identifier, template identifier, template version, and status Valid).	You can import the project but not the tag and page containers to which such facets are assigned. This requires that the project contains at least one container that can be imported, page 329, even empty. A dialog box opens, page 331 asking you to confirm the partial import.	Do either of the following first to create missing instances: <ul style="list-style-type: none"> • Use the export/import functionality, page 227 of the Application Explorer to import missing instances. • Create instances. This may require importing templates first. You can obtain information about the missing instances and their templates in the tooltip of the container that is tagged with a red dot.
An element of the imported instance that creates a facet is not selected.	The facet that is created by this element is not imported. The notification panel provides information about the facet that is not imported. For example, you import a Supervision project to which the data facet created by the <i>Maintenance</i> element of an instance is assigned but in the application of the target system, the <i>Maintenance</i> element of the same instance is not selected.	Edit the instance in the Application Explorer and select the corresponding element.
The properties of an instance are different from the instance configuration at the time of export. The difference pertains to the element selection (services) and/or the parameter values, page 182.	The assignment status of facets that are imported is set to Out Of Date .	Generate the project or container to apply the changes.
Topological entities (for example, NIC for a workstation) and services to which you mapped the executable (service and/or communication) of the imported project.	You can import the project but not the Supervision project executable and its associated mappings.	Use the export-import functionality, page 647 of the Topology Explorer first to create the missing topological entities.

Import Rules for Equipment

If you import a Supervision project and a project with the same identifier exists in the target system, the equipment, page 437 that is already generated in the Supervision project prevails for an instance that exists in the same cluster in both the imported file and the Supervision project. This applies independently of the selection of **Constituent File**, page 327 of the tag container.

Example:

Identifier of instance present in imported file and Supervision project	Equipment name generated by the imported instance	Equipment name generated by the instance in the Supervision project	Equipment in the Participant project after import
<i>MyMotor_1</i> ⁽¹⁾	<i>Folder_1.MyMotor_1</i> ⁽²⁾	<i>Folder_2.MyMotor_1</i> ⁽²⁾	<i>Folder_2.MyMotor_1</i>
(1) The instance is located in <i>Folder_1</i> .			
(2) Facets of the instance are assigned to <i>TagContainer_1</i> in <i>Cluster_1</i> .			

Import Rules for Pages

If you import a page and a page with the same identifier exists in the target system, the contents of the existing page is overwritten with the content of the page that is in the export file.

Import Rules for Page Templates

page templates, page 359 that you import appear in the **Page Templates** section of the Supervision project settings, page 302.

The page template that was selected as default template in the exported Supervision project becomes the default one after import.

If you import a page template and a page template with the same identifier exists in the target system, the **Resolve Conflicts in Page Templates** window opens where you can select to import the page template or not (**Skip** action). Either choice has no impact on the imported pages that use this page template.

If you import the page template (**Create** action), a time stamp suffix in the format YYYYMMDDHHMMSS is added to the name of the imported page template.

Import Rules for Facets

EcoStruxure Process Expert applies import rules to facets if the system where you are importing a Supervision project contains a Supervision project with the same identifier.

The following are the general import rules that apply if facets with the same identifier and assigned to tag containers with the same identifier exist in both the target project and the imported file:

- If you select the **Constituent File** check box of the tag container:
 - Facets from the export file overwrite facets of the target system and keep the state they have in the export file.
 - Facets that exist only in the export file are imported into the target system and keep the state they have in the export file.
 - Facets that exist only in the target system become **Assigned** and **Non Generated**. If the status of the facet is **Deleted**, **Unassigned**, or **Unlinked**, the facet is removed from the tag container after import.

- If you clear the **Constituent File** check box of the tag container:
 - Facets with the same identifier keep the state that they have in the target system.
 - Facets that exist only in the export file are imported into the target system and become **Assigned** and **Non Generated**.
 - Facets that exist only in the target system keep their state.

NOTE: EcoStruxure Process Expert changes the assignment status of facets of an instance to **Out Of Date** if a property of an instance, page 182 is different from when the instance was exported. This includes the location of the instance in the application. The notification panel provides information about the change of the assignment status after import is complete.

The following tables illustrate these import rules based on the different use cases. The tag container and facet names are examples and represent tag containers and facets in a cluster of the project.

Example if you select the **Constituent File** check box of the tag container.

Contents of export file being imported		Item in Supervision project of target system		Result in Supervision project of target system after import
Tag container identifier is identical				
Tag container1	Facet1	Tag container1	–	Facet1 from export file with status from export file.
Tag container1	Facet1	Tag container1	Facet1	Facet1 from export file with status from export file.
Tag container1	Facet1	Tag container1	Facet2	Facet1 from export file with status from export file. Facet2 with status Non Generated . If the status of Facet2 is Deleted, Unassigned or Unlinked , the facet is removed from the tag container after import.
Tag container identifier does not exist in target project				
Tag container2	Facet3	–	–	Tag container2 with Facet3 from export file with status from export file.
– The item does not exist				

Example if you clear the **Constituent File** check box of the tag container.

Contents of export file being imported		Item in Supervision project of target system		Result in Supervision project of target system after import
Container identifier is identical				
Tag container1	Facet1	Tag container1	–	Facet1 from export file with status Non Generated .
Tag container1	Facet1	Tag container1	Facet1	Facet1 from target system with same status.
Tag container1	Facet1	Tag container1	Facet2	Facet1 from target system with status Non Generated . Facet2 from target system with same status.
– The item does not exist				

Import Rules for Executables and Mappings

The **Build State**, page 779 of a **Built** executable is updated after import depending on the scenario.

If the system where you import a Supervision project does not contain a Supervision project with the same identifier, the state of **Built** executables that you import changes to **Out Of Date**.

If the system where you import a Supervision project contains a Supervision project and an executable with the same identifier, EcoStruxure Process Expert applies the following import rules to executables, service mappings, and communication mappings.

Contents of export file being imported	Executable in Supervision project of target system	Resulting Build State in Supervision project of target system after import
Executable1	Does not exist	Out Of Date
Executable1	Executable1	Out Of Date
Executable1	Executable2	Executable1: Out Of Date Executable2: Out Of Date

The following table describes the import rules that are applied to mappings depending on the scenario.

Contents of export file being imported	Mapping in Supervision project of target system	Result on mappings in Supervision project of target system after import
Service mapping	Exists	The service mapping from target system is not changed.
	Does not exist	The service mapping is created with the mapping information from the export file.
Communication mapping	Exists	The communication mapping from target system is not changed.
Communication mapping	Does not exist	The communication mapping is created with the mapping information from the export file.

Opening the Project Import Window

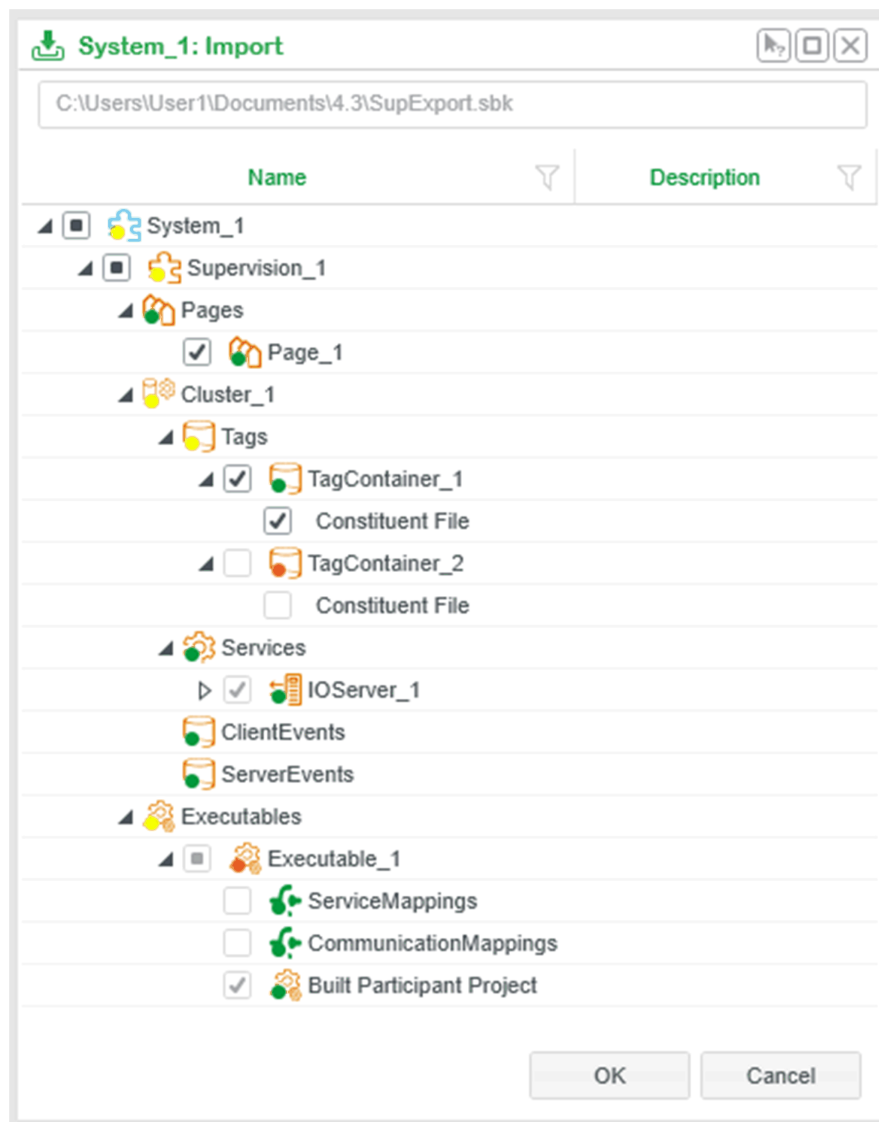
To open the **Import** window, proceed as follows.

Step	Action
1	Right-click the system root folder in a Project Browser of the Project Explorer and select Import . Result: The Import dialog box opens.
2	Browse to the export file (.sbk) that you want to import and click Open . Result: The Import window opens and displays the contents of the file. NOTE: You can cancel the command, page 331.

Project Import Window

The **Import** window lets you select the information of a Supervision project that you want to import into an existing system.

The following figure shows an example of the **Import** window displaying the contents of a Supervision project export file. In this example, the check boxes for *TagContainer_2* and relate mappings are cleared because the items cannot be imported. To indicate that at least one item of the Supervision project is not selected, the check box of parent nodes are filled with a square.



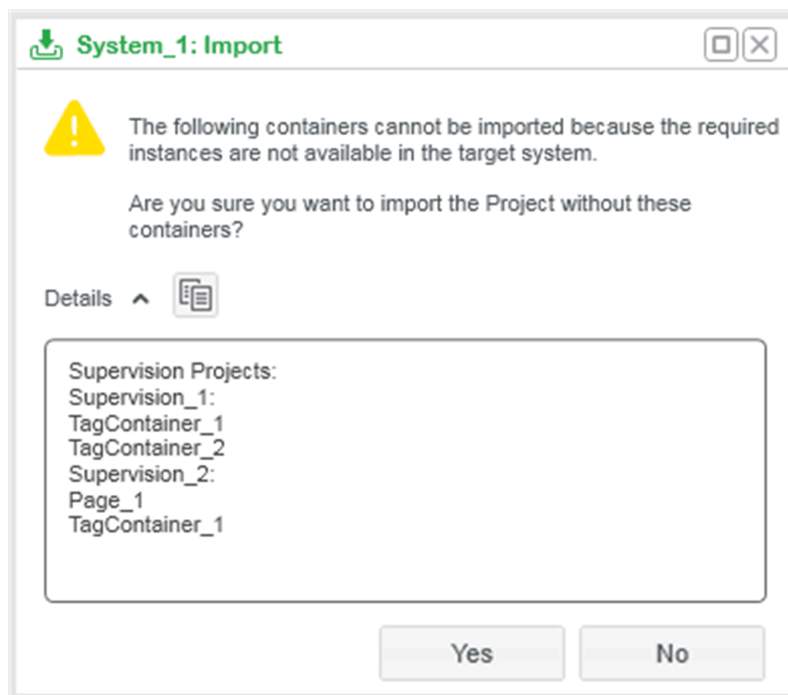
The table indicates the meaning of the colored dots that appear in the **Import** window next to items of a Supervision project.

Dot color	Description
Green	You can import the item.
Red	<ul style="list-style-type: none"> At the container level: The container (for example, the tag container or the page) contains at least one facet that is referenced by an instance that is not present in the application, page 326. At the executable level: <ul style="list-style-type: none"> An executable with the same identifier already exists in the target project. A project-related topological object, page 326 is not present in the target system. <p>A tooltip provides additional information about the issue. NOTE: You cannot import items that are tagged with a red dot.</p>
Yellow	<p>The item contains at least one sub-item that is tagged with a red dot.</p> <p>The item can only be partially imported.</p>

NOTE: To expand/collapse a node in the **Import** window, select the node and press **Enter**.

Confirmation Dialog Box for Containers That Cannot Be Imported

The following figure shows an example of the dialog box that opens if a container of a Supervision project selected for import is tagged with a red dot and you click **OK** in the **Import** window. If you are also importing Control projects, containers are grouped by Participant then, by project.



Button	Description
Yes	Closes the dialog box, the Import window, and imports the selected projects without containers tagged with a red dot.
No	Closes the dialog box and reverts to the Import window without importing projects. You can modify the selection of projects or cancel the import operation.

Aborting Import Tasks

After confirming the import by clicking **OK** in the **Import** window, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The project is not imported.

Only the user who selected the command is allowed to cancel it.

Importing a Supervision Project

Included projects contain data related to Supervision constituents (for example, genres and tags) that can be shared by several templates.

Replacing the existing **Included** project has an impact on:

- The templates that use the contents of the **Included** project.
- The Supervision Participant projects that are associated to the **Included** project.

NOTICE

LOSS OF DATA

Resolve an **Included** project conflict that arises during the import of a Supervision Participant project by selecting the appropriate version of the **Included** project to be used.

Failure to follow these instructions can result in equipment damage.

To import data of a Supervision project into an existing system, proceed as follows.

Step	Action
1	In the Import window, page 329, select the items that you want to import.
2	Click OK . Result: The import is performed if the applicable rules are satisfied. If a conflict is detected for the name of an Included project, the Resolve Content Conflict dialog box opens, page 325. A summary of the completed actions and detected incompatibilities appears in the notification panel.
3	If the Resolve Content Conflict dialog box opens, do one of the following: <ul style="list-style-type: none"> • Select Version in export file to overwrite the existing Included project with the one you are importing. • Select Version in contents repository to keep the existing Included project. NOTE: Click Cancel to discard your selection in the Resolve Content Conflict dialog box and revert to the Import window.
4	Click OK . Result: Actions are performed according to your selection.

Participant Project Indicators

Overview

This topic describes the various indicators that appear on components of Control and Supervision Participant projects.

Participant Project Progress Indicators

Overview





Various icons give an indication of the steps that remain before the system engineering life cycle is completed up to the deployment stage for Control and Supervision projects.

The progress icon is not visible while the checked-out icon, page 338 is shown.





NOTE: The indicator icons are visible to any user who opens the Participant project.

Meaning of Icons





For Control and Supervision projects.

Icon	Condition description
 (0% progress)	Either one applies: <ul style="list-style-type: none"> 0% progress for all project containers. At least one facet is <i>candidate</i>, page 346 for assignment.
	The progress of at least one container is not 100%.
	Either one applies: <ul style="list-style-type: none"> The progress of all containers is 100% and no executable exists. The project progress is 75% and executables are deleted.
	The progress of at least one executable is not 100%.
No icon (100% progress)	The progress of at least one executable is 100%.

For sections and containers.

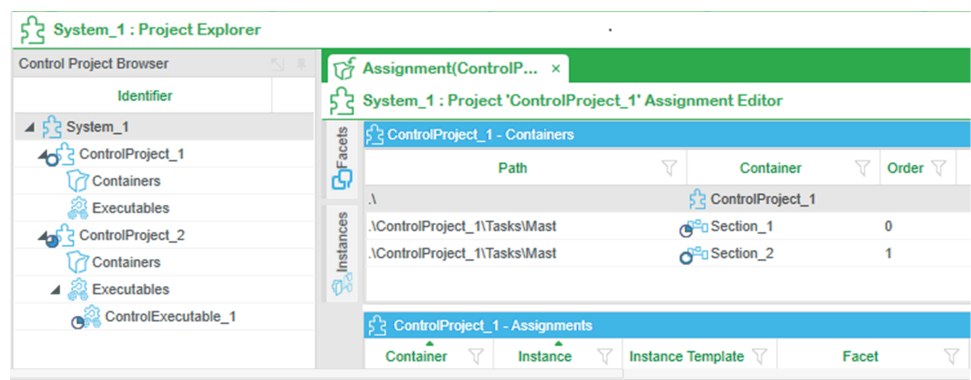
Icon	Condition description
 (0% progress)	Either one applies: <ul style="list-style-type: none"> No assignment is done. All assignments have been removed.
	At least one facet is Non Generated .
	At least one facet, page 424 is Out Of Date, Deleted, Unassigned, Moved, Inconsistent , or Invalid and no facet is Non Generated .
	All facets are Generated but the analysis detected one or more errors, page 336.
No icon (100% progress)	All facets are Generated or Unlinked without detected errors during analysis.

For executables.

Icon	Condition description
 (0% progress)	Either one applies: <ul style="list-style-type: none"> The executable is created. Executable progress is 100% and service mapping, page 719 is removed.
	<ul style="list-style-type: none"> Control: Service mapping is completed. Supervision: Service mapping is done for at least one service. Or, executable progress is 100% and service mapping is modified.
	<ul style="list-style-type: none"> Control: At least one communication mapping or hardware mapping is done. Supervision: Communication mapping is completed. And the executable state is Not Built .
	The executable state is Out Of Date or Built with alerts, page 339.
No icon (100% progress)	The executable state is Built .

How Icons Appear

The following figure shows an example of several Control projects and their components that have various completion states, which are indicated by progress icons.



Participant Project Validity Indicators

Overview

A dot or progress icon, page 334 that appears in red or yellow color indicates an alert or detected error, page 339 during the analysis of components of a Participant project.



When both an alert and the detection of an error occur for the same object, the detected error icon prevails.

When an icon appears on at least one component of a project (for example, a section, a container, an executable), it appears automatically at the project level.

The indicator icons are visible to any user who opens the Participant project.

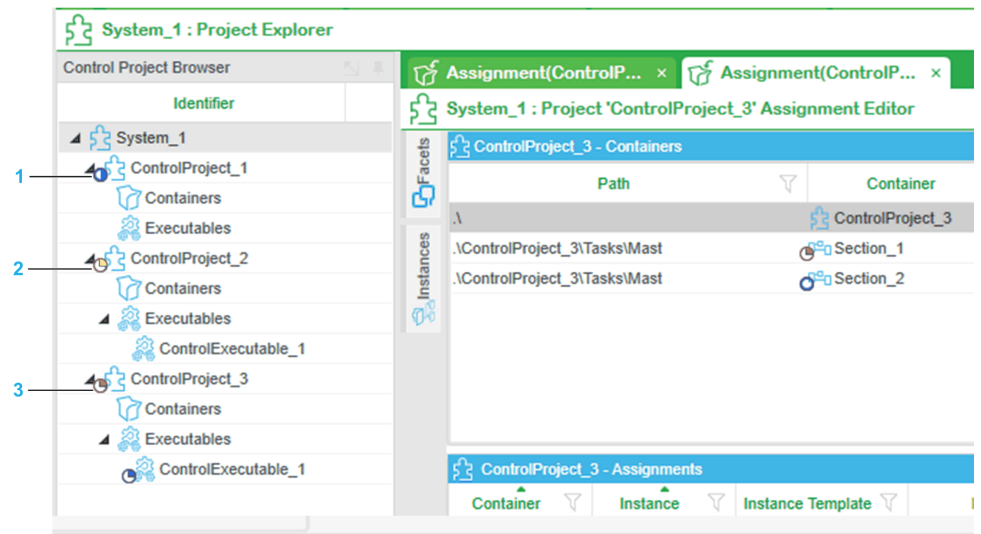
NOTE: The last action summary, page 339 can help provide information about issues that are not indicated by validity icons.

Meaning of Icons

Icon color	Level	Condition description
Yellow (progress icon or dot for objects with 100% progress) Example: 	Project	<ul style="list-style-type: none"> Control projects: The analysis, page 401 of the project or saving refinement changes, page 449 generated an alert indicating that the build operation may not complete successfully. Supervision projects: Saving changes generated an alert. All projects: At least one alert was generated for a project component.
	Section	The analysis, page 401 of the section or saving refinement changes, page 449 generated an alert indicating that the build operation may not complete successfully.
	Tag container or page	Generation completed with at least one alert.
	Page	Saving changes generated at least one alert.
	Executable	Build completed with at least one alert.
Red (progress icon or dot for objects with 100% progress) Example: 	Project	<ul style="list-style-type: none"> Control projects: The analysis, page 401 of the project or saving refinement changes, page 449 detected an error, which prevents the build operation from completing successfully. Supervision projects: Saving changes detected an error. All projects: There is at least one detected error for a project component.
	Section	The analysis, page 401 of the section or saving refinement changes, page 449 detected an error, which prevents the build operation from completing successfully.
	Tag container or page	Generation completed with at least one detected error.
	Page	Saving changes detected at least one error.
	Executable	Build did not complete successfully.

How Icons Appear

The following figure shows an example of several Control projects and their components that have various validity states, which are indicated by the color of the progress icons.



Item	Description
1	Valid project with 50% progress.
2	Project with 25% progress and at least one alert.
3	The detected error at the section level is also shown at the project level.

Participant Project Write-Lock Indicator

Overview

A lock icon lets you see which Participant project components are being used or modified by other processes or users and that you cannot work on them.



Certain components that are locked can be opened in read-only mode.

NOTE: The indicator icon is visible to any user who opens the Participant project.

Meaning of the Icon

The icon indicates that the project or a component is locked because an operation is in progress, such as:

- Assignment⁽¹⁾
- Generation
- Refinement
- Editing
- Mapping⁽¹⁾
- Export/import
- Build

(1) The lock icon appears very briefly while an performed is performed.

Where the Icon Appears

The lock icon can appear at the following levels:

- The Participant project
- Sections
- Containers (such as Supervision pages or tag containers)
- Clusters
- **Services** node and any of its subcomponents (such as servers, I/O devices)
- **ClientEvents** and **ServerEvents** nodes
- Executables

Viewing the Last Action Summary

Overview

The **Last Action Summary** command lets you view the last action that was performed on a given Participant project component and a summary of the result. The command is available in the context menu of projects and their components.

For actions that had not completed successfully or with alerts, and/or detected errors, it shows again the information that was displayed originally.

Supported Actions

An action summary is available after you perform either action:

- Generation
- Regeneration
- Build

Particular Conditions

Last action performed	Result
Not a supported action.	The command is unavailable.
A supported action was aborted.	An action summary for aborted actions is not available. The action summary for the previous action remains available (if applicable).
Import of a Participant project.	The command is unavailable. The action summary that existed at the time of export is cleared. NOTE: The last action summary of the target project is preserved when you perform an incremental import. That is, when you import a project into a system in which a project with the same identifier already exists.

Assignment Stage

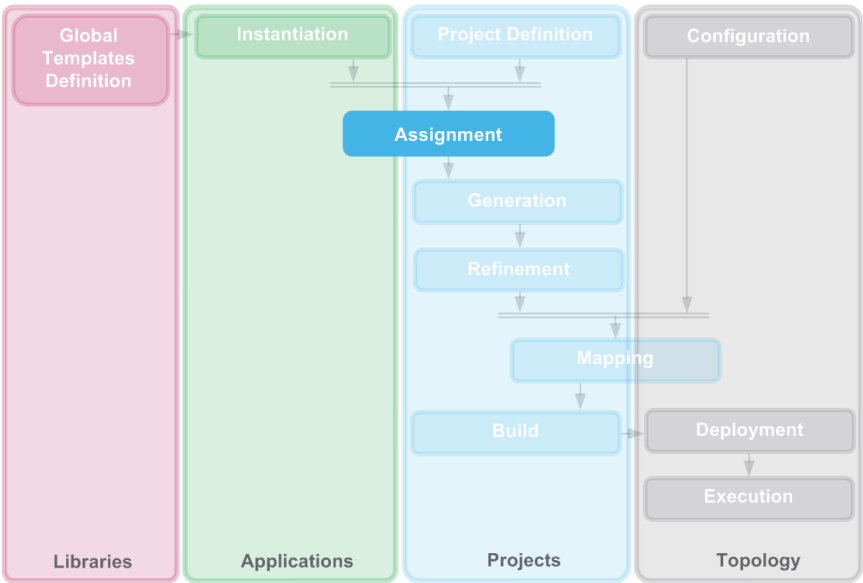
What's in This Chapter

Creating and Managing Sections, Instances, and Containers.....	341
Assigning Facets of Instances	362
Managing Assigned Facets	389

Overview

This chapter describes how to create Control and Supervision containers and assign facets of instances of an application to the corresponding projects that you have created during the Projects Definition stage.

The following figure shows the position of the Assignment stage within the system engineering life cycle.



Refer to the Assignment stage, page 56 for a description of the purpose of this stage.

Creating and Managing Sections, Instances, and Containers

Overview

This section describes the **Assignment Editor** and how to use it to create and manage Control sections and Supervision containers. It also describes section and container context menu commands and properties.

As part of the [simplified workflow](#), page 69, it describes how to create instances in-place.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Assignment Editor

Overview

The **Assignment Editor** lets you create and manage Control and Supervision containers and assign the facets of instances of an application to them.

You can assign by facet, instance, folder, or application depending on the extent of the assignment, page 346 that you want to perform.

The editor gives you the flexibility to assign manually, automatically, or by combining both methods.

NOTE: As part of the simplified workflow, page 69, it lets you create and manage instances in-place by using context menu commands of FBD sections, page 348 and assigned Control facets, page 390.

The following table describes the two methods that you can use to assign facets to Control projects, page 375.

Section creation	Facet assignment	Prerequisite	How to use
Automatic, user-defined. With possibility to: <ul style="list-style-type: none"> Rename sections, page 372 Select which sections to create 	Automatic ⁽¹⁾ , user-defined, with possibility to preview assignments, page 372.	Configure assignment mechanism, page 363.	Assign objects by dragging them to the project.
Manual	Manual	Create sections, page 348	Assign objects by dragging them to a section.
(1) By default, remaining candidate facets are assigned automatically, page 381 during generation			

The following table describes the methods that you can use to assign facets to Supervision projects, page 377.

Tag container creation	Facet assignment	Prerequisite	How to use
Automatic With possibility to: <ul style="list-style-type: none"> Rename tag containers, page 372 Select the cluster 	Automatic ⁽¹⁾ with possibility to preview assignments, page 372.	Create additional clusters, page 305 as needed.	Assign objects by dragging them to the project.
Manual	Manual	Create tag containers, page 353 in the Assignment Editor and additional clusters, page 305 (as needed).	Assign objects by dragging them to the tag container.
	Automatic, page 383	You have associated a Control project to the tag container.	Assigning instances to the Control project automatically assigns the corresponding Supervision facets to the tag container.
(1) By default, remaining candidate facets are assigned automatically, page 381 during generation			

Opening the Assignment Editor

In a Control or Supervision project browser of the **Project Explorer**, double-click the **Containers** node, page 266 of the project to which you want to assign facets.

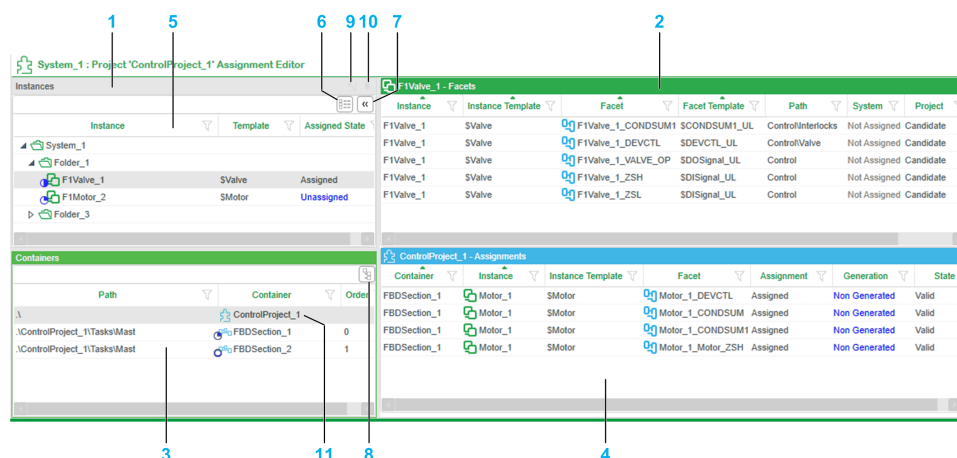
NOTE: You can also open the **Assignment Editor** by using the **View Assignments**, page 392 window.

Assignment Editor

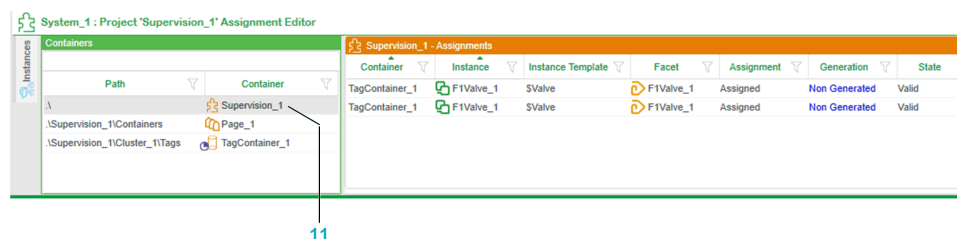
The **Assignment Editor** for Control and Supervision projects is identical except for the color scheme of the **Containers** and **Assignments** pane:

- Blue: Displays facets, containers, and assignments of Control projects.
- Orange: Displays facets, containers, and assignments of Supervision projects.

The following figure shows an example of the **Assignment Editor** for Control projects.



The following figure shows an example of the **Assignment Editor** for Supervision projects. The **Instances** and **Facets** panes are minimized to a tab by unpinning the **Instances** pane.



Item	Description
1	Instances pane. Shows the compatible instances of the applications that have a valid Data and Link status.
2	Facets pane. Shows Participant-specific facets, which are referenced by the items that are selected in the Instances pane. The pane is empty by default. Supervision Genie facets are not displayed.
3	Containers pane. Shows the project and its sections or containers. This pane is used to select the target to which you want to assign facets.
4	Assignments pane. Shows facets that are assigned to the project or container that is selected in the Containers pane. The title bar of the pane is where you need to drag objects to assign them.
5	Columns. You can select which columns, page 345 you want to see in each pane.
6	Button to toggle between grid and tree view in the Instances pane.
7	Button to show or hide facets, page 346 in the Facets pane. The Instances pane must be pinned.
8	Button to toggle between grid and tree view in the Containers pane, page 345.
9	Button to show or hide, page 113 the Instances and Facets panes
10	Button to extract or retract the Instances pane.
11	Project node, which gives you access to project-level context menu commands.

The following tables describe the information that appears in the panes of the **Assignment Editor**.

Instances pane.

Column header	Description
Instance	Identifier of the instance.
Template	Identifier of the template that the instance is using.
Version	Version of the template that the instance is using.
Assigned State	<ul style="list-style-type: none"> • Assigned: At least one Control or Supervision facet of the instance is assigned to a project of the system. • Unassigned: No facet of the instance is assigned to any project of the system.
Path	Path to the instance in the tree view of the Application Browser . The column appears only if you have selected the grid view display mode.
Candidate	Provides information about the assignment of facets at the instance level: <ul style="list-style-type: none"> • Done: The facets of the instance are already assigned to this project. • Candidate: The instance contains at least one facet that is not yet assigned to this project.
Description	Contents of the <i>\$Description</i> parameter of the instance.

Facets pane.

Column header	Description
Instance	Identifier of the instance that references the facet.
Instance Template	Identifier of the template of the instance that references the facet.
Facet	Identifier of the facet.
Facet Template	Identifier of the template that the facet is using.
Path	Path to the facet as it is shown in the Assignment Viewer , page 392.
System	Indicates the assignment ⁽¹⁾ status of the facet at the system level: <ul style="list-style-type: none"> • Assigned: The facet is already assigned to a project of the system. • Not assigned: The facet is not yet assigned to a project of the system.
Project	Indicates the assignment status of the facet at the project level: <ul style="list-style-type: none"> • Candidate: You have not yet assigned the facet to this project. • Done: You have already assigned the facet⁽¹⁾.
Type	Provides information on the subtype of the facet.
(1) Generating the section that contains the facet may change this status depending on the actions that you have executed on the facet (for example, unassigning it after a first generation). The Assignment column of the Assignments pane provides you with a more detailed information on the assignment status of the facet.	

Containers pane.

Column header	Description
Path	For a Control project, indicates whether the section belongs to the Mast or Fast task folder. For a Supervision project, indicates to which cluster the container belongs.
Container	Shows the identifier of the project and the sections or containers that it contains. An icon indicates the language of a section or the type of container. NOTE: Although Supervision pages appear, you cannot assign genie facets, page 508 to them by using the Assignment Editor .
Order	For Control projects, indicates the order in which the sections are generated. You can change the order of sections, page 275. For more detailed information on the section order management, refer to the generation process description, page 403. NOTE: The order is not shown in tree view mode.

For a description of the **Assignments** pane, refer to the topic *Assigned Facet Description*, page 379.

Customizing Column Display

In the various panes of the editor, you can choose which columns you want to see by right-clicking a column header and selecting **Customize**. Certain columns are always shown.

When you hide a column for which a filter is applied, the filter is cleared.

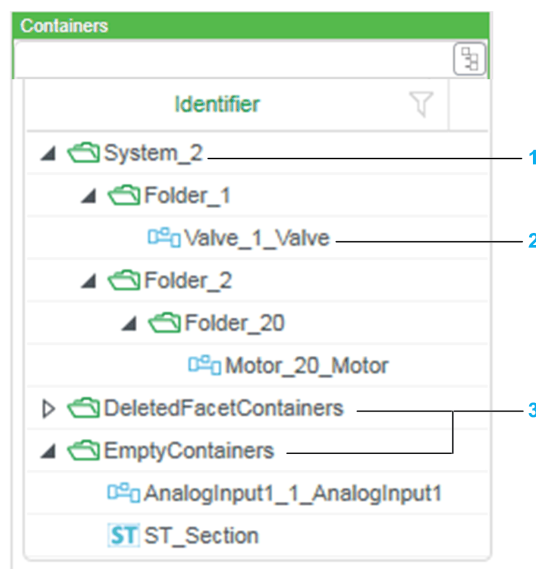
If tree and grid view are available, you may be able to configure the column display independently in each view.

Displaying Sections in Tree View Mode

The tree view mode allows you to view sections to which facets are assigned in relation to the folder structure of the application.

For more details, refer to the topic describing the representation of the application structure in Control projects, page 384.

The figure shows an example of the **Containers** pane in tree view mode once facets have been assigned to some FBD sections.



item	Description
1	System root folder and subfolders representing the folder structure of the application. NOTE: Predefined folder icons, page 176 that you may have assigned in the application to various folder levels are not represented.
2	Section identifier.
3	Special root folders.

Viewing Instances with Facets That Can Be Assigned

To view the instances of applications that have facets that can be assigned to the project, proceed as follows.

Step	Action
1	Switch to grid view mode by clicking the Grid View button in the Instances pane.
2	Click the Show Facets button to show the candidate column in the Instances pane. Result: The Instances pane displays the instances of the applications. Those that have a status of Candidate in the Candidate column have facets that can be assigned to this project.

NOTE: Use the instance progress icons, page 196 to see which instances have facets that need to be assigned.

Viewing Facets That Can Be Assigned

To view the facets of applications that can be assigned to the project, proceed as follows.

Step	Action
1	Switch to tree view mode by clicking tree view the button in the Instances pane.
2	In the Instances pane, select the system root folder.
3	Click the Show Facets button to show facets in the Instances pane. Result: The Facets pane displays the facets of application instances. Those that have a status of Candidate in the Project column can be assigned to this project.

NOTE:

- To view only the facets with status **Candidate**, use the filter on the **Project** column.
- To narrow the display to an application or a part of it, select an application folder instead of the system root folder.

Object Selection

The following table describes the scope of the assignment depending on the objects that you drag from the **Instances** or **Facets** pane of either **Assignment Editor**. Each editor assigns only Participant-specific facets.

Object	Resulting assignment
One or more facets of the same or several instances.	These facets only. Remaining facets are assigned automatically, page 381.
One or more instances.	The facets of all the selected instances.
One or more folders.	The facets of all the instances that are contained in the selected folders. By dragging the system root folder, you can assign facets of the existing applications at once.

NOTE: If the facet that you are trying to assign is not compatible with the selected container, you cannot create the assignment and a tooltip indicates it. If you try to assign multiple items at once and some of them are not compatible, the assignment is created for the compatible items and information about the incompatible items is displayed in the notification panel.

Creating Control Project Sections

Overview

In the **Assignment Editor**, you can create only FBD sections in a Control Participant project.

This topic describes how to create sections manually. Sections can also be created automatically, page 363.

To create non-FBD sections, refer to the topic describing the **Project Explorer**.

Creating FBD Sections

To create an FBD section in the **Assignment Editor**, proceed as follows.

Step	Action
1	Right-click the identifier of the Control project in the Containers pane and select Create FBD Section . Result: The Create FBD Section dialog box opens.
2	Edit the identifier, page 349 of the section if required, select the Mast or Fast task folder, and click OK . Result: A new FBD section is created with the next available order value, page 275.

Inserting FBD Sections

To insert an FBD section, proceed as follows.

Step	Action
1	In the Containers pane of the Assignment Editor , right-click the section where you want to insert the new one and select Insert FBD Section . Result: The Insert FBD Section dialog box opens.
2	Edit the identifier, page 349 of the section and click OK . Result: The newly created section is inserted and has the order, page 275 of the section that you had right-clicked. The order of the section that you right-clicked and those that follow is incremented by 1.

Section Actions

The table describes the context menu commands of sections.

Command	Description
Insert FBD Section	Creates a new FBD section, page 348 at the position of this section.
Paste Instances	Lets create a copy of instances that have been added to the Clipboard by using the Copy Instances facet context menu command, page 390. The pasted instance has the following properties: <ul style="list-style-type: none"> Its identifier is suffixed with _X (where X is an integer) It has the same configuration as the copied instance. It is created in the Application Browser in a folder named <i>HierarchyNotDefined</i>, which is in the root of the system. Its facets are assigned to the section on which you have used the paste command. <p>NOTE: If the instance has Supervision data facets and the automatic assignment of Supervision facets, page 383 is enabled, these facets are assigned.</p>

Command	Description
Create Instance	<p>Opens the Create Instance dialog box, page 351, which lets you select a template of the Global Templates Library to create one instance and assign its facets to the section. The facet selection is based on the default Control service configuration of the template.</p> <p>You can also edit, link, and copy the instance, page 390.</p> <p>The command is only available for FBD sections.</p>
Generate	<p>Refer to <i>Generation Stage</i>, page 396.</p> <p>The command is only available for FBD sections.</p>
Refine	<p>Refer to <i>Refinement Stage</i>, page 446.</p> <p>The command is not available for LL984 sections.</p>
ReGenerate	<p>Refer to <i>Generation Stage</i>, page 434.</p> <p>The command is only available for FBD sections.</p>
Delete	<p>Deletes the section.</p> <p>If you have assigned, page 341 facets to the section:</p> <ul style="list-style-type: none"> Unassigns, page 391 any facets contained in the FBD section that you are deleting. Changes the System status, page 425 of the facets to Not assigned. <p>NOTE: If the FBD section has been generated, deleting it also removes from the logical Control Participant project the section and constituents that were generated by the facets assigned to the section (for example, variables).</p> <p>However, derived types that were generated by these facets are not removed (for example, FBDs and DDTs). To remove these, you can either:</p> <ul style="list-style-type: none"> Regenerate, page 434 the Control project. This discards refinements. Purge the types from the Control Participant during refinement. For details, refer to <i>Purge of Unused Instances</i> in the Control Participant help. <p>NOTE: To make the command available for peer to peer communication sections, refer to the topic describing how to modify and delete peer to peer communication mapping, page 489.</p>
Rename	Lets you edit the identifier, page 349 of the section.
Properties	Opens the Properties window of the section.

Section Properties

To open the **Properties** window of a section, right-click the section and select **Properties**.

You can edit the following items.

Item		Description
General	Identifier	<p>Depending on how you create the section, the default identifier can be either:</p> <ul style="list-style-type: none">• <i>FBDSection_n</i> if you create the FBD section manually, using the Create FBD Section command, where <i>n</i> is an incremental number starting at 1.• <Instance identifier>_<instance template identifier> if the section is automatically created during assignment. <p>NOTE: For identifiers of Schneider Electric templates, the \$ prefix is omitted.</p> <p>The identifier must be unique within the Control project.</p> <p>Identifiers must satisfy the naming rules, page 32.</p>
	Description	<p>Optional.</p> <p>You can enter a description of the section with free-form text.</p>
	Generated On	<p>Date and time of the last successful generation or regeneration of the section.</p> <p>The property is read-only and contains information only for FBD sections.</p>

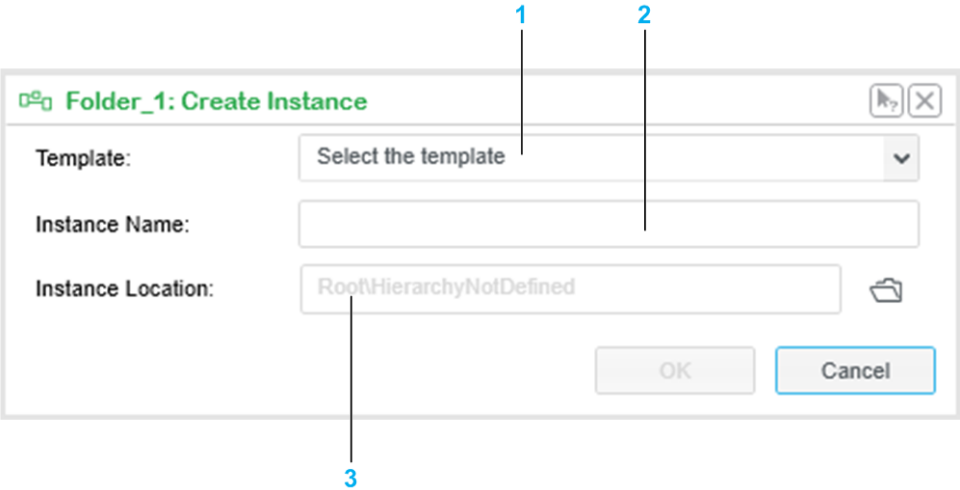
Creating Instances In-Place

Overview

As part of the simplified workflow, page 69, you can create and manage instances in Control projects from within the **Assignment Editor**.

Create Instance Dialog Box

The following figure shows an example of the **Create Instance** dialog box.



Item	Description
1	List of the composite and facet application templates that exist in the Global Templates Library with usability state <i>Approved</i> .
2	Editable name of the instance, page 191 (<i>\$Name</i>).
3	Path in the Application Browser where the instance will be created. You can select an existing folder. By default, the instance is created in the <i>HierarchyNotDefined</i> folder, which is in the root of the system.

Creating Instances

Step	Action
1	In the Containers pane of the Assignment Editor , right-click an FBD section and select Create Instance . Result: The Create Instance dialog box opens.
2	In the Template menu, select the template to use to create the instance.
3	Edit the instance name. NOTE: If you select a different template, the instance name is reset.
4	Select a folder of in the Application Browser where the instance is created.

Step	Action
5	<p>Click OK.</p> <p>Result:</p> <ul style="list-style-type: none">• The instance is created with the default template configuration.• Its Control facets are assigned to the section from which you have created the instance. <p>NOTE: If the automatic generation setting, page 272 is enabled, the section is generated unless an error is detected, page 402.</p> <p>NOTE: If the instance creates Supervision data facets and the automatic assignment of Supervision facets, page 383 is enabled, these facets are assigned. Follow the regular system engineering workflow to manage the Supervision project.</p>
6	You can edit, link, and copy the instance, page 390.

Creating Supervision Tag Containers

Overview

This topic describes how to create tag containers manually by using the **Assignment Editor**, their properties, and context menu commands. EcoStruxure Process Expert creates tag containers automatically when you assign facets, page 342.

You need to create, at least, one tag container per controller to assign Supervision data facets of instances of the application.

These data facets represent the tags that are managed by Supervision components and assigned to the I/O device variables to be associated to the Supervision Participant project:

- Variable tags
- Trend tags
- Alarm tags

Monitoring data of a device, page 984 communicating by using the Modbus TCP protocol (or Modbus through an Ethernet gateway) from the Supervision infrastructure requires creating, in addition, one tag container for each device. Supervision *Data* facets representing device-specific data are assigned to each container. Also, a specific communication protocol needs to be defined for these tag containers.

Creating Tag Containers

To create a tag container in the **Assignment Editor**, proceed as follows.

Step	Action
1	Right-click the identifier of the Supervision project in the Containers pane and select Create Tag Container . Result: The Create Tag Container dialog box opens.
2	Edit the identifier, page 354 of the tag container if required, select an existing cluster, and click OK . Result: A new tag container is created.

Using a User-Defined Protocol

When you enter a protocol in the **Protocol** parameter of the tag container, page 354 instead of selecting a predefined one, verify the following:

- The protocol is supported by the platform of the controller that is the data source of the variable tags; otherwise, the variables cannot be read and/or written to by Supervision software.
- It is supported by the Supervision Participant and installed; otherwise, you cannot build the Supervision executable.

To verify which protocols are installed in the Supervision Participant, refine the Supervision Participant project and expand an entry in the **Protocol** column of **I/O Devices** in the **Topology** activity.

If the protocol is not installed, contact Schneider Electric support.

When using a user-defined protocol, some data generated at the template level, such as variable tag addresses may not be in the required protocol-specific format depending on the template configuration. After generating the Supervision Participant project, refine it and edit the data as needed. No configuration file (.xml) is generated when you build the Supervision Participant project.

NOTE: For communication with the I/O server, ensure that the protocol of the communication board, page 312 is correctly configured.

Tag Container Actions

The table describes the tag container context menu commands.

Com-mand	Description
Create Tag Container	Lets you create a new tag container in an existing cluster.
Generate , page 426	Starts the generation process to populate new tags, update out-of-date tags, and/or remove deleted tags in the Supervision Participant project.
Delete	<p>Deletes the tag container.</p> <p>If you assigned facets to the container:</p> <ul style="list-style-type: none"> • The facets contained in the container that you are deleting are unassigned. • The system status of the facets changes to Not assigned. <p>NOTE: The servers that were automatically created during the assignment of Supervision facets are not deleted.</p> <p>NOTE: If the container has been generated, deleting it also deletes the data in the logical Participant project when you generate the project again.</p>
Rename	Lets you enter a new identifier for the tag container.
Properties	Opens the Properties window, which allows you to edit the properties of the tag container.

Tag Container Properties

To open the **Properties** window of the tag container, right-click the container and select **Properties**.

You can edit the following items.

Item		Description
General	Identifier	<p>The identifier must be unique within the Supervision project.</p> <p>Default value: <i>TagContainer_n</i> where <i>n</i> is an incremental number starting at 1.</p> <p>NOTE: When Protocol for the tag container is set to <i>OPCUA</i>, the identifier is used as follows:</p> <ul style="list-style-type: none"> During generation, it becomes part of the variable tag address in the Supervision Participant project. The format of the address is <i>ns=2;s=0:<tag container ID>!/<variable name></i>, where <i>ns</i> indicates the namespace and 2 corresponds to the <i>OPC UA</i> server as defined by Schneider Electric. <p>For example, <i>ns=2;s=0:Controller1!InputVar1</i></p> <ul style="list-style-type: none"> During build, it becomes the alias for the device in the OFS configuration file, page 788. <p>Therefore, it may be useful to rename the tag container with the identifier of the controller that is mapped to the associated I/O device, page 769.</p>
	Description	<p>Optional.</p> <p>You can enter a description of the container with free text.</p>
	Generated On	Date and time of the last successful generation of the tag container (read-only).
Attributes	ScanTime	<p>The acquisition time (milliseconds) for the tags that are assigned to the container.</p> <p>Default value: 0</p>
	Protocol	<p>The protocol that is used for communication with the I/O device.</p> <p>The menu contains the following values:</p> <ul style="list-style-type: none"> <i>OFSOPC</i> (default) <i>OPCUA</i> <p>NOTE: When using the OPC UA protocol with templates of a Schneider Electric Library, clear the Use Native Types checkbox of the Unity/Control Expert symbols parameter (selected by default) so that the correct Control Participant types are used by the OPC UA server. For details, refer to <i>PLC Software Settings</i> in the OPC UA Server Expert help.</p> <ul style="list-style-type: none"> <i>OPCUA Embedded</i>: When the OPC UA server is embedded in a module (see EcoStruxure Process Expert, Control Participant Services, User Guide) of the controller. Sets the protocol of the associated I/O device to <i>OPCUA</i> in the Supervision Participant project. User-defined protocol, page 353 whose name can be entered by using free-form text. 16 characters maximum. Values entered are added to the menu. <p>When you instantiate Schneider Electric templates and assign their Supervision facets to the Supervision project, only <i>OFSOPC</i>, <i>OPCUA</i>, and <i>OPCUA Embedded</i> are supported. This is because, by default, the facets generate Supervision data (variables and tags) that is specific to these protocols. To use another protocol, you can add the protocol-specific Supervision data to the project either by instantiating and configuring the <i>\$VariableTag_CD</i> template or by refining the Supervision project.</p> <p>NOTE: For each additional tag container that you create to monitor data of a device directly from the Supervision runtime, change the value to <i>OPC</i>.</p> <p>For information on the corresponding driver, refer to the driver help. To locate driver help files, look up <i>Driver Reference Help</i> in the help of the Supervision Participant, page 95</p>

Item		Description
Attributes	Direct TCP IP Addressing Mode ⁽¹⁾	<p>Sets the addressing mode for the <i>TCP/IP</i> driver in the OFS/OPC UA configuration file. Select the mode depending on the Modbus TCP topological entity, page 769 that is mapped to the associated I/O device in the Communication Mapping tab.</p> <p>The parameter is disabled when Protocol for the tag container is set to <i>OPCUA Embedded</i>.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Unallocated/Allocated Data, PAC Modicon (/U) (default): Corresponds to selecting <i>UNITY / Control Expert</i> in the PLCs section of the Device address wizard in the OFS/OPC UA configuration tool. Use this setting when mapping a controller or workstation. • Allocated Data, Offset 0 (/T): Corresponds to selecting <i>PL7</i> in the PLCs section. • Allocated Data, Offset 0 with minimum communication (/NT): Corresponds to selecting <i>PL7</i> in the PLCs section and <i>Minimum communication profile</i>. • Allocated Data, Offset 1 (/Q): Corresponds to selecting <i>CONCEPT</i> in the PLCs section. • Allocated Data, Offset 1 with 32 bits swapping (/QW): Corresponds to selecting <i>CONCEPT</i> in the PLCs section and <i>Swap 32-bit Values</i>. • Allocated Data, Offset 1 (/NQ): Corresponds to selecting <i>MODBUS Device</i> in the PLCs section. <p>The value is available only when Protocol is set to <i>OPCUA</i>.</p> <p>NOTE: The value shown in brackets corresponds to the suffix that is appended to the device address in the configuration file.</p> <p>For more information, refer to:</p> <ul style="list-style-type: none"> • <i>Editing the Device Network Address</i> in the help of OPC Factory Server, page 104. • <i>Managing Device Aliases</i> in the help of OPC UA Server Expert.
Attributes	Supports Optimized Modbus Frames ⁽¹⁾	<p>Corresponds to the <i>Modbus Request Optimization</i> parameter of Communication Settings in the OFS configuration tool.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • <i>Off</i> (default): The check box of the parameter is cleared. • <i>On</i>: The check box of the parameter is selected. <p>The parameter is not available for the OPCUA protocol.</p> <p>NOTE: Because in OFS, this parameter applies to all aliases of the OFS configuration file, if you configure several tag containers in a Supervision project and for one of them you select <i>Off</i>, the check box of the parameter in the OFS configuration file is cleared, overriding the <i>On</i> setting that you may have configured for any other tag container.</p>

Item		Description
Relation-ships	IoDevices	<p>Select the I/O device that you want to associate with the container.</p> <p>The I/O device appears in the menu once you have created it or assigned facets.</p> <p>Default value: <i>IODevice_1</i></p> <p>NOTE:</p> <ul style="list-style-type: none"> By default, EcoStruxure Process Expert associates the tag container with the I/O device that it automatically creates when you assign Supervision facets to the container, page 377. If you create the I/O device manually, page 309, associate it with the tag container before or after assigning facets. Changing the relationship after assigning facets to the tag container changes their assignment status to Out Of Date.
	Associated Control Project	<p>Lets you select one existing Control project that will trigger the automatic assignment, page 383 of Supervision facets to the tag container when you assign Control facets.</p> <p>You can associate a Control project to several tag containers given they belong to a different Supervision project.</p> <p>When the associated Control project is renamed, the value is updated. When it is deleted, the value is reset.</p> <p>Default value: Not Assigned</p>
(1) The parameter becomes read-only when <i>Protocol</i> is different than <i>OFSOPC</i> or <i>OPC UA</i> .		

Creating Supervision Pages

Prerequisites

- Generate the Supervision project before creating pages.
- If you want to use a specific style for the page, verify that it has been added in the Supervision project **settings**, [page 302](#) and that it is selected as the default style.

Creating Supervision Pages

To create a page in the **Assignment Editor**, right-click the identifier of the Supervision project in the **Containers** pane and select **Create Page**.

Supervision Page Actions

The table describes the page context menu commands.

Command	Description
Create Page	Lets you create a new Supervision page.
Edit , page 509	Opens the Supervision Participant, which allows you to assign to the page animated graphics that are associated with the instances of the Supervision project. It also allows you to add graphic elements to the page to represent process hardware.
Generate , page 426	Starts the generation process to update out-of-date Genies, and/or remove deleted Genies in the Supervision Participant project.
Delete	Deletes the page including assignments of animated graphics.
Rename	Lets you enter a new identifier for the page.
Properties	Opens the Properties window, which allows you to edit the properties of the page.

Supervision Page Properties

To access the **Properties** window of the Supervision page, right-click the page and select **Properties**.

You can edit the following items.

Item		Description
General	Identifier	The identifier must be unique within the Supervision project. Identifiers must satisfy the naming rules , page 32 .
	Description	Optional. You can enter a description of the page with free-form text.
	Generated On	Date and time of the last successful generation of the page (read-only).

Selecting Supervision Page Templates

Overview

By using the **Page Templates** section, page 302 of the Supervision project settings, you can add and remove page templates and select the template to be used by default for pages of the Supervision project.

The settings apply to each Supervision project of a system individually.

The following page templates are provided with EcoStruxure Process Expert:

- *Classic Templates*: Best suited for templates of the General Purpose Library (GPL) Classic.
- *Situational Awareness*: Best suited for templates of the General Purpose Library (GPL).

Page templates that you want to add need to be contained in an archive file (.zip).

Creating Page Template Archive Files

To create an archive file (.zip) that contains a page template, proceed as follows.

Step	Action
1	Release (see EcoStruxure Process Expert, Supervision Participant Services User Guide) the Supervision Participant and open AVEVA Plant SCADA Studio outside of EcoStruxure Process Expert.
2	Create a new project, make it active, and open the Graphics Builder .
3	Click File > New... > Page .
4	In the Use Template dialog box, select a style, resolution, and a template to use for the page. Click OK . Result: The new page opens.
5	Save the page in the new project with the name <i>page</i> (Page text box) and click OK . NOTE: You will not be able to use the page template if you save it with a different name.
6	Close the Graphics Builder .
7	In the menu bar of AVEVA Plant SCADA Studio, click Backup . Result: The Backup Project dialog box opens.
8	In the Backup file text box, click Browse... , change the file extension to <i>zip</i> , click Open , and OK to backup the project as an archive in zip format.
9	By using the Windows File Explorer, browse to the location where you have backed up the project and open the archive.
10	Delete all the files in the archive except for the following: <ul style="list-style-type: none"> • Page.ctf • Page.ctg • pgadv.DBF • pgbutton.DBF • pgdynobj.DBF • pgkey.DBF • pgtouch.DBF
11	Save the archive file.
12	To create another page, reopen the Graphics Builder and repeat steps 3 to 11. Ensure to save the new page with the name <i>page</i> , which replaces the page that you had created previously.

Adding Page Templates to a Supervision Project

Step	Action
1	Right-click the Supervision project and select Settings .
2	Select the Page Templates section.
3	Click the Add new template button. Result: The Browse Page Templates dialog box opens.
4	Select an archive file (.zip) containing the page template that you want to add and click Open . Result: The page template is shown in the Page Templates section.
5	Double-click the page template identifier and description to edit them.
6	Click OK to close the Settings window and Yes to save changes.
7	If the page template that you have added is based on an external included project that is not present in the content repository, add the included project to the Supervision project by using the Included Projects context menu command, page 508 of the project. Otherwise, if the page template is used by a page, you cannot build the Supervision project. NOTE: You only need to add the included project the first time you add a page template to the Supervision project of a system. Thereafter, when you add the page template (or another page template using the same include) to another Supervision project, the required included project is already available.

Using a Page Template

Step	Action
1	In the Page Templates section of the Settings window, select the page template that you want to use by selecting its Is Default property.
2	Click OK to close the Settings window and Yes to save changes. Result: Pages that you create from now on in this Supervision project use the page template. If you have added an external included project, it is shown when you use the Included Projects command, page 300 after you create the first page.

Removing Page Templates

To remove a page template, verify that its **Is Default** property is not selected in the **Page Templates** section and click the delete button in the **Action** column. Save changes when closing the **Settings** window.

You cannot remove the page templates provided with EcoStruxure Process Expert.

NOTE: Removing a page template does not impact existing pages that were created with this template and does not remove from the content repository the included project associated to this template.

Changing the Page Templates of Existing Pages

You cannot change the page template of an existing page by using the **Page Templates** section.

You need to edit the page and change the appearance manually.

Exporting and Importing Projects With Page Templates

Page templates are exported when you export a Supervision project and appear in the **Export** window.

For details on the import rules that apply to page templates, refer to the topic describing the import of Supervision projects, page 323.

Assigning Facets of Instances

Overview

This section describes how to use the **Assignment Editor** to assign facets of instances to their respective Control and Supervision containers. It also details the information that is provided on the resulting assignments.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Defining the Control Facet Assignment Mechanism

Overview

The **Container Naming Rules** window allows up to four expressions and lets you use either one to define the assignment mechanism of Control facets, which includes:

- The name of sections that are automatically created when you assign facets.
- Which facets are assigned to which section.

Expressions are project-specific.

By default, one preconfigured expression, page 365 is selected in each project.

The expression is applied when you drag instances or folders containing instances to the Control project in the **Assignment Editor**.

If the resulting container identifier does not satisfy applicable naming rules, it is indicated, page 373.

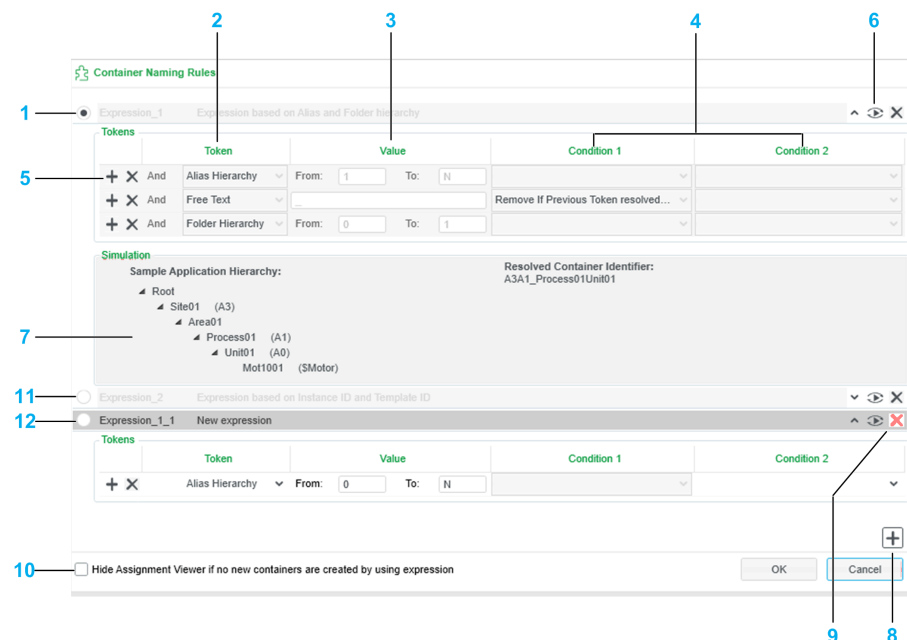
NOTE: You can preview and modify, page 372 the result or override the mechanism by assigning facets to an existing section.

Opening the Container Naming Rules Window

To open the **Container Naming Rules** window, select **Settings** in the context menu, page 269 of the Control project.

Container Naming Rules Window Description

The following example shows the **Container Naming Rules** window for a new Control project with its two preconfigured expressions of which the first one is selected.



Item	Description
1	<p>Preconfigured expression, page 365 (read-only). Select the expression to use it to create sections and assign facets.</p> <p>You can copy expressions, page 366 and edit the name and description of expressions that you add.</p>
2	<p>To create the expression, you can configure up to six tokens. To create the section identifiers, the values that EcoStruxure Process Expert calculated for each of the configured tokens are concatenated.</p> <p>The tokens that you include in the expression have an impact on the number of sections, page 365 that are created.</p> <p>The available tokens are:</p> <ul style="list-style-type: none"> • Alias Hierarchy: Alias of the folder, page 172. • Folder Hierarchy: Identifiers of folders that make up the path of the instance. • Free Text: Free-form text that you can enter. It lets you add a fixed value to all sections that are created or insert a separator. Underscore is the only special character that is allowed. • Instance ID: Identifier of the instance, page 177. • Template ID: Identifier of the template, page 177 that is used by the instance.
3	<p>For Alias Hierarchy and Folder Hierarchy, you can select which levels in the path of the instance to consider.</p> <p>0 corresponds to the folder that contains the instance. 1 corresponds to its parent folder, and so on.</p> <p>N corresponds to the highest folder in the structure. The system root folder is excluded (except for instances that have the system root folder as direct parent).</p> <p>Maximum value: 99</p> <p>Example:</p> <p>For an instance located in <i>System_1\Site_1(S1)\Area_1(A1)\Cell_1(C1)\Unit_1(U1)</i> (where the value in parentheses represents the folder alias):</p> <ul style="list-style-type: none"> • 0 = <i>Unit_1</i> (or its alias, <i>U1</i>) • 1 = <i>Cell_1</i> (or its alias, <i>C1</i>) • 2 = <i>Area_1</i> (or its alias, <i>A1</i>) • N = <i>Site_1</i> (or its alias, <i>S1</i>) <p>Configuring Value as From: 1 To: 2 for each token generates the following result:</p> <ul style="list-style-type: none"> • For Folder Hierarchy: <i>Cell_1Area_1</i> • For Alias Hierarchy: <i>C1A1</i>
4	<p>Optionally, you can define up to two conditions, page 366 for each token (except for the first and last token, which support only one condition each).</p> <p>The following conditions are available:</p> <ul style="list-style-type: none"> • Condition 1: <ul style="list-style-type: none"> ◦ Remove if previous token resolved value is empty ◦ Remove if all previous tokens resolved value is empty ◦ Remove if previous token resolved value is NOT empty ◦ Remove if all previous tokens resolved value is NOT empty • Condition 2: <ul style="list-style-type: none"> ◦ Remove if next token resolved value is empty ◦ Remove if all next tokens resolved value is empty ◦ Remove if next token resolved value is NOT empty ◦ Remove if all next tokens resolved value is NOT empty
5	Buttons to add and remove tokens in an expression.
6	<p>Simulate Expression button.</p> <p>Lets you simulate the resulting section identifier by using the expression that is configured based on a sample folder structure and instance identifier if you were to assign any part of this sample hierarchy to the Control project in the Assignment Editor.</p> <p>To refresh the result after modifying the expression, click the button again.</p>
7	<p>Pane showing the sample folder structure and the resulting section identifier.</p> <p>Values in bracket represent the alias of the folder and the identifier of the template.</p>

Item	Description
8	Lets you add a custom expression. You can add up to two expressions but only one expression can be selected at a time.
9	You can delete expressions that you have added.
10	When selected, opens the Assignment Viewer window, page 372 only if the assignment creates at least one new section based on the selected expression.
11	Additional preconfigured expression, page 365 (read-only).
12	User-created expression that can be configured and selected.

Container Creation Principle

The table describes the number of sections, page 369 that are created depending on the token that the expression contains. When several tokens are included, the one with the lowest granularity determines the number of sections. Tokens are listed in increasing order of granularity.

Token	Number of sections created
Instance ID	As many sections as there are instances in the object that you are dragging.
Folder Hierarchy⁽¹⁾	As many sections as there are folders containing instances in the object that you are dragging (when Value = From: 0 To: N). This token lets you assign facets of instances located in several folders of a same level, page 370 to one section.
Alias Hierarchy⁽¹⁾	As many sections as there are folders that have an alias and that contain instances (instances of subfolders included) in the object that you are dragging. This token lets you assign facets of instances located in several folders of different levels, page 371 to one section.
Template ID	As many sections as there are templates used by instances in the object that you are dragging. This token lets you assign facets of several instances that use the same template, page 368 to one section.
(1) The value that you configure impacts the number of sections that are created and the section identifier	

NOTE: The conditions, page 366 that are configured may impact the results.

Preconfigured Expressions

The following preconfigured expressions exist and are read-only. The section identifier that they create is composed of three parts where each part is the result of a calculated token.

To use a different expression, create one and select it.

For **new** Control projects:

The expression creates a section for each folder containing instances.

Example:

Alias1Alias2Alias3_Folder3Folder4

The table describes how the tokens of the expression are used to calculate the different parts of the section identifier.

Token	Value	Result
Alias Hierarchy	From: 1 To: N	Concatenation of the folder aliases from the highest level folder down to the level-1 folder in the path of the instance.
Free Text	_ (underscore)	An underscore. NOTE: The condition removes the underscore if no aliases have been configured because a section identifier cannot start with an underscore.
Folder Hierarchy	From: 0 To: 1	Concatenation of the identifiers of the level-1 and level-0 folders in the path of the instance starting with the level-1 folder.

For **new** and **migrated** Control projects:

The expression creates a section for each instance. The section identifier is the same as the one that was automatically created when assigning facets by using earlier versions.

It is selected by default for migrated projects.

Example:

MyMotor_1_Motor

The table describes how the tokens of the expression are used to calculate the different parts of the section identifier.

Token	Value	Result
Instance ID	n/a	The identifier of the instance.
Free Text	_ (underscore)	An underscore.
Template ID	n/a	The identifier of the template that is used by the instance (without \$ prefix).

Using Conditions in Expressions

You can use conditions to modify container identifiers. Conditions remove a token from an expression based on the value of one or all previous tokens and/or the value of one or all following tokens.

It lets you, for example, show the folder identifier instead of the folder alias in case no alias is defined.

Example of condition based on previous token:

You have configured an expression as `<token1><token2><token3>` and an underscore is defined as free text for `token2`.

If the value for `token1` is blank, the resulting container identifier starts with an underscore. Configure the condition **Remove if previous token resolved value is empty** for `token2`. The container identifier is then only `token3`.

Example of condition based on next token:

You have configured an expression as `<token1><token2><token3>` and `Section_A` is defined as free text for `token1`. However, you want this text to appear only when `token2` and `token3` contain no value.

You can configure the condition **Remove if all next tokens resolved value is not empty** for `token1` to achieve this.

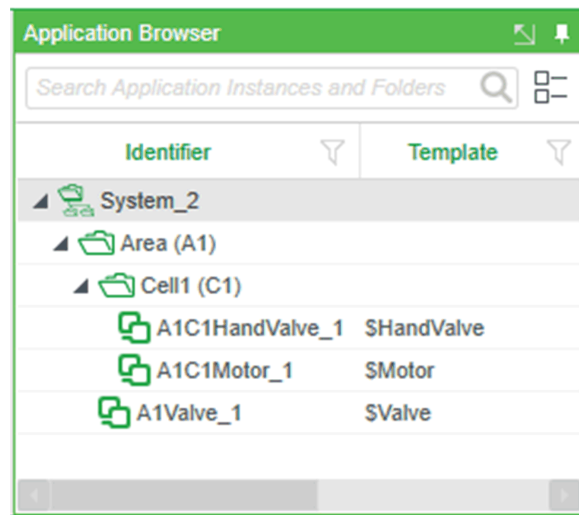
Copying Expressions

You can copy expressions to paste them in the same window or in the window of another project.

Step	Action
1	Right-click the header of the expression and select Copy .
2	Right-click the empty space below an expression and select Paste . Result: The expression is added to the Container Naming Rules window. NOTE: You can only paste as many expressions as is allowed to remain within the limit of expressions per project.

Sample Application for Examples

The following figure shows the sample application that is used in the examples that follow. In each one, *Area* is dragged to the Control project in the **Assignment Editor**

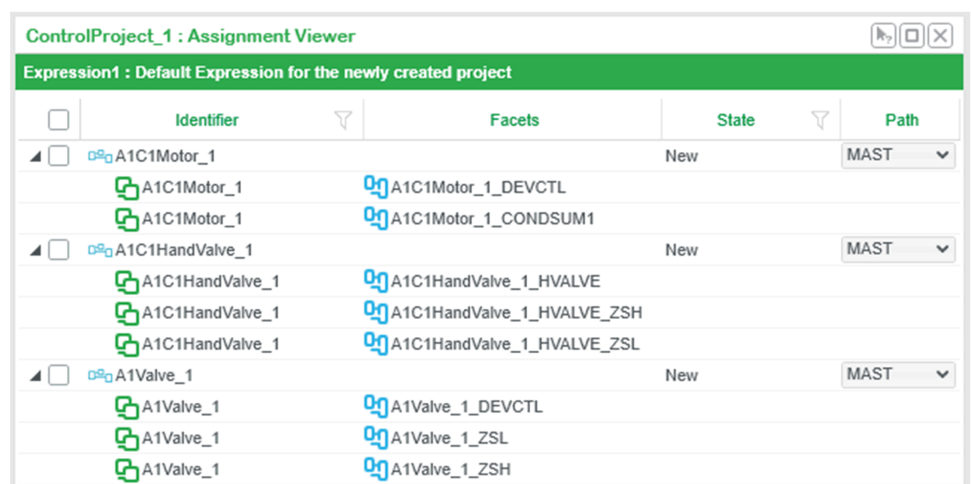


Example Using Instance ID Token

Only **Instance ID** is configured as token without conditions.

Tokens				
	Token	Value	Condition 1	Condition 2
1.	Instance ID			
2.				
3.				

After dragging *Area*, one section for each instance is proposed. The section identifier is the instance identifier.



Example Using Folder Hierarchy and Instance ID Tokens

Folder Hierarchy is added to **Instance ID** as tokens with an underscore as separator. No conditions.

Tokens				
Token	Value		Condition 1	Condition 2
1. Folder Hierarchy	From: 0	To: 1		
2. Free Text				
3. Instance ID				

Compared to the previous example, page 367, the number of sections remains the same. Only the identifier of the sections is changed by adding the identifier of the folder containing the instance (level 0) and its parent (level 1) as prefix.

ControlProject_1 : Assignment Viewer				
Expression1 : Default Expression for the newly created project				
Identifier	Facets	State	Path	
AreaCell1_A1C1Motor_1		New	MAST	
A1C1Motor_1	A1C1Motor_1_DEVCTL			
A1C1Motor_1	A1C1Motor_1_CONDSUM1			
AreaCell1_A1C1HandValve_1		New	MAST	
A1C1HandValve_1	A1C1HandValve_1_HVALVE			
A1C1HandValve_1	A1C1HandValve_1_HVALVE_ZSH			
A1C1HandValve_1	A1C1HandValve_1_HVALVE_ZSL			
Area_A1Valve_1		New	MAST	
A1Valve_1	A1Valve_1_DEVCTL			
A1Valve_1	A1Valve_1_ZSL			
A1Valve_1	A1Valve_1_ZSH			

Example Using Template ID Token

Instance *Motor_1* that is using template *\$Motor* is added to the application so that two instances of the same template exist.

Application Browser	
Search Application Instances and Folders	
Identifier	Template
System_2	
Area (A1)	
Cell1 (C1)	
A1C1HandValve_1	\$HandValve
A1C1Motor_1	\$Motor
A1Motor_1	\$Motor
A1Valve_1	\$Valve

Only **Template ID** is configured as token without conditions.

Tokens				
Token	Value		Condition 1	Condition 2
1. Template ID				
2.				
3.				

One section for each template that is used in the application is proposed. The section identifier is the template identifier (without \$ character).

ControlProject_1 : Assignment Viewer

Expression1 : Default Expression for the newly created project

<input type="checkbox"/>	Identifier	Facets	State	Path
▲ <input type="checkbox"/>	Motor		New	MAST ▼
	A1C1Motor_1	A1C1Motor_1_DEVCTL		
	A1C1Motor_1	A1C1Motor_1_CONDSUM1		
	A1Motor_1	A1Motor_1_DEVCTL		
	A1Motor_1	A1Motor_1_CONDSUM		
	A1Motor_1	A1Motor_1_CONDSUM1		
▲ <input type="checkbox"/>	HandValve		New	MAST ▼
	A1C1HandValve_1	A1C1HandValve_1_HVALVE		
	A1C1HandValve_1	A1C1HandValve_1_HVALVE_ZSH		
	A1C1HandValve_1	A1C1HandValve_1_HVALVE_ZSL		
▲ <input type="checkbox"/>	Valve		New	MAST ▼
	A1Valve_1	A1Valve_1_DEVCTL		
	A1Valve_1	A1Valve_1_ZSL		
	A1Valve_1	A1Valve_1_ZSH		

Example Using Template ID and Alias Hierarchy Tokens

Alias Hierarchy is added to **Template ID** as tokens with an underscore as separator. No conditions.

Tokens

	Token	Value	Condition 1	Condition 2
1.	Alias Hierarchy ▼	From: 0 To: 2	▼	▼
2.	Free Text ▼	▼	▼	▼
3.	Template ID ▼	▼	▼	▼

Compared to the previous example, page 368, one section has been added because the two instances of template *\$Motor* are located in folders that have a different alias. The section identifier is the template identifier with the aliases of *Cell1* and *Area* added as prefix.

ControlProject_1 : Assignment Viewer

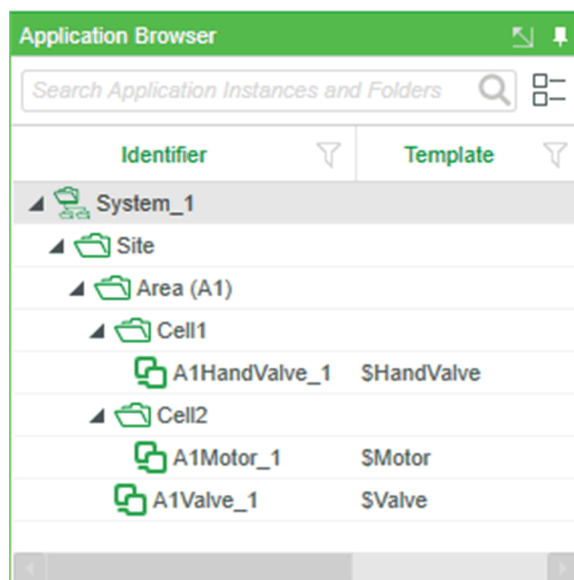
Expression1 : Default Expression for the newly created project

<input type="checkbox"/>	Identifier	Facets	State	Path
▲ <input type="checkbox"/>	A1C1_Motor		New	MAST ▼
	A1C1Motor_1	A1C1Motor_1_DEVCTL		
	A1C1Motor_1	A1C1Motor_1_CONDSUM1		
▲ <input type="checkbox"/>	A1C1_HandValve		New	MAST ▼
	A1C1HandValve_1	A1C1HandValve_1_HVALVE		
	A1C1HandValve_1	A1C1HandValve_1_HVALVE_ZSH		
	A1C1HandValve_1	A1C1HandValve_1_HVALVE_ZSL		
▲ <input type="checkbox"/>	A1_Valve		New	MAST ▼
	A1Valve_1	A1Valve_1_DEVCTL		
	A1Valve_1	A1Valve_1_ZSL		
	A1Valve_1	A1Valve_1_ZSH		
▲ <input type="checkbox"/>	A1_Motor		New	MAST ▼
	A1Motor_1	A1Motor_1_DEVCTL		
	A1Motor_1	A1Motor_1_CONDSUM		
	A1Motor_1	A1Motor_1_CONDSUM1		

NOTE: The same number of folders would have been created if **Folder Hierarchy** had been selected as token instead of **Alias Hierarchy**.

Example to Group Facets of Instances Located in Several Folders of Same Level

This example illustrates how to use the **Folder Hierarchy** token to assign to one section the facets of instances located in folders of a same level in the application (*Cell1* and *Cell2*). The following figure shows the sample application that is used.



Folder Hierarchy is configured as token with **value = From: 1 To: 1**. Because the value represents the parent folder, it creates a section for each folder that contains instances by using the identifier of its parent. When two folders have the same parent, only one section is created. No conditions.

Tokens				
Token	Value		Condition 1	Condition 2
1. Folder Hierarchy	From: 1	To: 1		
2.				
3.				

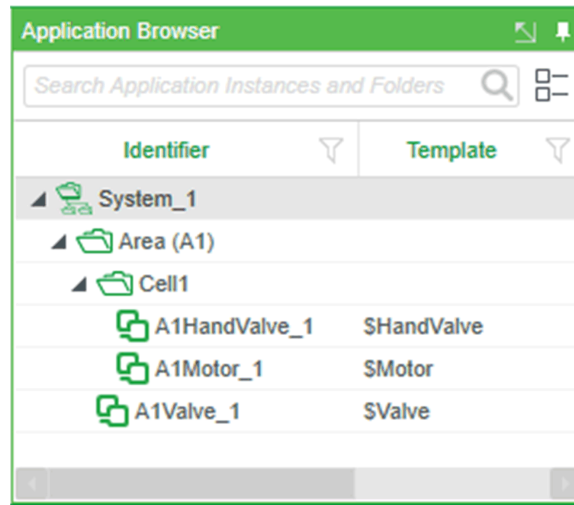
After dragging *Area* to the Control project, one section is proposed; *Area*. It contains the facets of instances *A1HandValve_1* and *A1Motor_1* because it is the parent of folders *Cell1* and *Cell2*.

ControlProject_1 : Assignment Viewer				
Expression_1 : Default expression				
<input checked="" type="checkbox"/>	Identifier	Facets	State	Path
<input checked="" type="checkbox"/>	Area		New	MAST
	A1HandValve_1	A1HandValve_1_HVALVE		
	A1HandValve_1	A1HandValve_1_HVALVE_ZSH		
	A1HandValve_1	A1HandValve_1_HVALVE_ZSL		
	A1Motor_1	A1Motor_1_DEVCTL		
	A1Motor_1	A1Motor_1_CONDSUM1		
<input checked="" type="checkbox"/>	Site		New	MAST
	A1Valve_1	A1Valve_1_DEVCTL		
	A1Valve_1	A1Valve_1_ZSL		
	A1Valve_1	A1Valve_1_ZSH		

NOTE: If the section identifier is not satisfactory, you can edit it in-place, page 373 or add a **Free Text** token to the expression.

Example to Group Facets of Instances Located in Several Folders of Different Levels

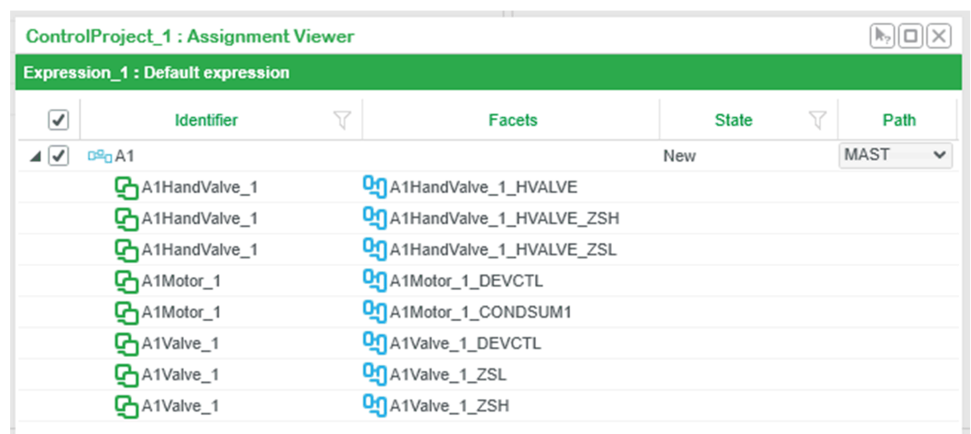
This example illustrates how to use the **Alias Hierarchy** token to assign to one section the facets of instances located in folders of different levels in the application (*Area* and *Cell1*). The following figure shows the sample application that is used.



Alias Hierarchy is configured as token with **Value = From: 0 To: 1**. This creates a section for each folder that has an alias and contains instances. No conditions.

Tokens					
	Token		Value	Condition 1	Condition 2
1.	Alias Hierarchy	From:	0	To:	1
2.					
3.					

After dragging *Area* to the Control project, one section is proposed, which contains the facets of the three instances of the application. *Cell1* contains instances but has no alias. *Area* contains an instance and has an alias. Thus, its alias *A1* is used to create the section identifier to which facets of *A1Valve_1* are assigned. Facets of any instances contained in subfolders (*Cell1*) are assigned there as well.



NOTE: If the section identifier is not satisfactory, you can edit it in-place, page 373 or add a **Free Text** token to the expression.

Previewing Facets Assignments

Overview

The **Assignment Viewer** opens when you drag one or more facets to a project in the **Assignment Editor**.

It does not open if you drag facets to a container that you have created manually.

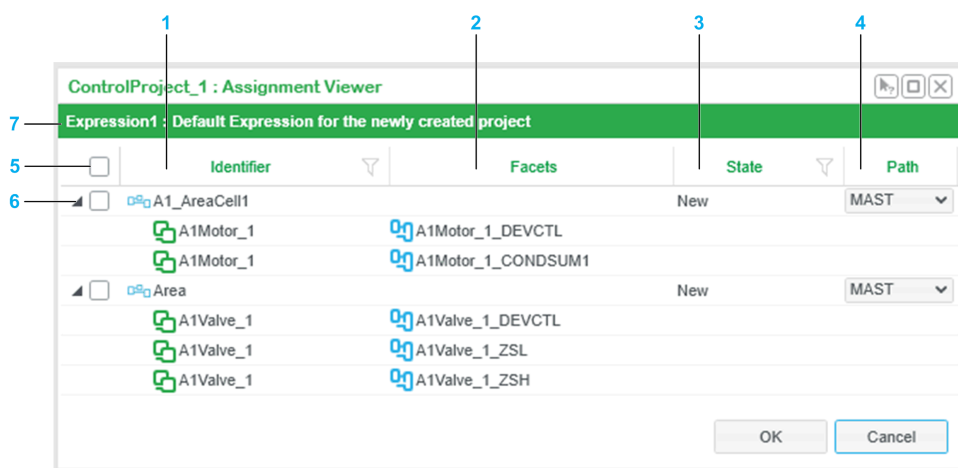
The window lets you:

- For Control projects, preview the identifier of newly created sections, which is the result of the expression that you have selected, page 363.
- Preview which facets will be assigned to which container.
- Edit the identifier, page 373 of containers in-place.
- Select in which task folder or cluster to create new containers.
- Select which assignments to perform.

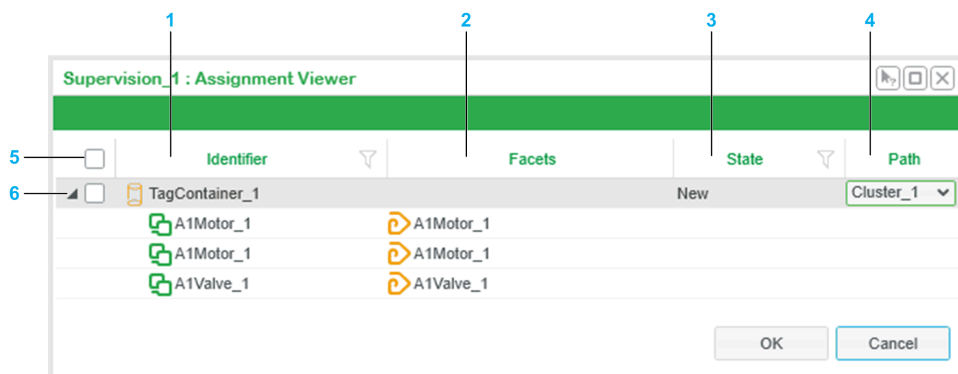
NOTE: For Control projects, you can configure it, page 363 to open only if the facets that you are assigning create a new section.

Assignment Viewer Description

The following figure shows an example of the **Assignment Viewer** when you drag the top level folder of a folder structure containing two instances⁽¹⁾ to a Control project while the default expression, page 365 for a new project is selected.

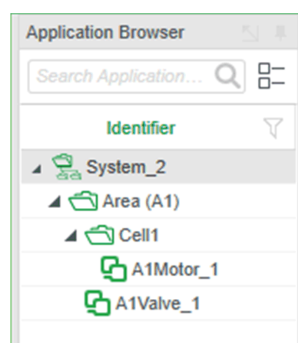


The following figure shows an example of the **Assignment Viewer** when you drag the top level folder of a folder structure containing two instances⁽¹⁾ to a Supervision project.



Item	
1	<p>Shows the identifier of the containers to which the facets of the instances that you have dragged will be assigned. For each container, shows the identifier of the instances whose facets will be assigned.</p> <p>By default, the order of containers and instances corresponds to the hierarchy of the application.</p> <p>For Control projects, the number of sections and their identifier depends on the <i>expression</i> that is selected, page 363 and the object that you have dragged (for example, an instance, a folder, or a folder structure).</p> <p>For Supervision projects, one tag container is created each time you drag an object to the project independently of the number of instances inside the object.</p> <p>You can edit the identifier of new sections.</p>
2	Identifiers of the facets that will be assigned to the container.
3	Indicates whether the container already exists in the project or if it will be created.
4	<p>Indicates or, for new containers, lets you select:</p> <ul style="list-style-type: none"> For Control projects: The task folder of the section. For Supervision projects: The cluster of the container.
5	Check box to select all the containers that appear in the window.
6	<p>Select the check box to create the container (if new) and assign the facets.</p> <p>Containers whose check box is cleared are ignored.</p>
7	For Control projects, indicates the name of the <i>expression</i> , page 363 that is selected.

(1) The following figure shows the application that was used for the two **Assignment Viewer** examples.



Editing Container Identifiers

You can edit the identifier of new containers only.

By editing the identifier of a container so that it is the same as an existing container or another container that is proposed in the **Assignment Viewer**, you can assign the facets of both to a single container.

Container Identifier Validation

Either of the following dots can appear on the container icon:

- A red dot to indicate that the container identifier that is the result of the expression is blank.
- A yellow dot to indicate that the container identifier has been modified to satisfy applicable naming rules, [page 31](#) (for example, maximum length or allowed characters).

Modifying the Application While the Assignment Viewer is Open

If the application is modified while the **Assignment Viewer** is open and the changes impact any result of the expression that is selected, the information in the window is not updated. Also, when you proceed with the assignments, they are performed based on the information that is displayed in the window.

Assigning Control Facets of Instances

Before Starting

To assign Control facets of instances, you must have completed:

- The Instantiation stage, page 157
- The Control Project Definition stage, page 268

You can only assign facets of instances that have a valid **Data** and **Link** status, page 177.

You can assign facets only to FBD sections.

You cannot assign the same facet to more than one FBD section in the same project but you can assign the same facet to several Control Participant projects of the system.

NOTE: You can associate a Control project to a tag container to assign the corresponding Supervision facets automatically, page 383.

Facet Assignment Order

Depending on the method that you use to select instances for assignment, the order of the referenced facets varies. The following applies when you assign all facets to the same FBD section.

Assignment method	Facet order
Instance by instance	The order in which you drag instances to assign them defines the order of the referenced facets. Instances selected first for assignment have facets with the lowest order.
Multiple instances at once	The order in which you select instances defines the order of the referenced facets. Instances selected first have facets with the lowest order.
By folder	The order of instantiation defines the order of the referenced facets. Instances created first have facets with the lowest order.
Multiple folders at once	The order in which you select folders defines the order of the referenced facets. Folders selected first have facets with the lowest order. If the folders contain multiple instances, the order in which instances where created is not relevant anymore.

NOTE: You can change the order of facets, page 394.

Assigning Facets

This procedure describes how to assign facets of instances to sections of the Control project by using an expression, page 363.

Step	Action	Result
1	Verify that the instances are displayed in tree view mode in the Instances pane; otherwise, click the toggle button, page 343. NOTE: Alternatively, you can assign instances using the grid view mode. The assignment principle remains unchanged. You can switch between views at any time during assignment by clicking the toggle button.	The Assignment Editor displays the valid, page 177 application instances in tree view mode.
2	Configure the expression depending on the assignment mechanism that you want to use and select it.	—

Step	Action	Result
3	In the Containers pane, select the Control project.	<ul style="list-style-type: none"> The name of the project appears in the title bar of the Assignments pane. Any existing sections and the facets already assigned to them are displayed.
4	From the Instances pane, drag one or more objects, page 346 to the title bar of the Assignments pane. NOTE: Alternatively, you can drag objects directly to project in the Containers pane.	The Assignment Viewer window, page 372 opens.
5	Review the information that is displayed, edit section identifiers as needed, and select the task folder for each section.	—
6	Click OK .	<ul style="list-style-type: none"> Sections that do not exist yet are created after existing sections. The Control facets that have a status of Candidate are assigned to the respective FBD sections and displayed in the Assignments pane. EcoStruxure Process Expert assigns an order, page 379 to each assigned facet. The status of the assigned facets in the System column of the Facets pane changes to Assigned if the status was Not assigned. The status in the Project column of the Facets pane changes from Candidate to Done.

NOTE: By default, remaining facets will be assigned automatically, page 381 when you generate the project or a section.

Assigning Supervision Facets of Instances

Before Starting

To assign Supervision data facets of instances, you must have completed:

- The Instantiation stage, page 157
- The Supervision Project Definition stage, page 298

You can only assign facets of instances that have a valid Data and Link status, page 177.

You cannot assign the same facet to more than one tag container in the same project but you can assign the same facet to several Supervision projects of the system.

NOTE: To assign Supervision genies facets to a page, page 358, refine the Supervision project.

Automatic Supervision Project Component Creation

Each time you drag an object from the **Instances** pane to the Supervision project, one tag container is created, page 372.

In addition, depending on the facets that you assign to the tag container, EcoStruxure Process Expert may automatically create and configure the following components under the **Services** node, page 301 of the Supervision project cluster:

- One I/O server including:
 - One I/O device
 - One communication board and its port
- One alarm and/or one trend server (if facets that create alarm and/or trend tags have been assigned).

NOTE: The I/O server and its components are created only for the first tag container of a cluster whether you create the tag container manually or it is automatically created during the assignment. The other servers are created also when you assign corresponding facets to additional tag containers.

Assigning Facets

This procedure describes how to assign data facets of instances to a Supervision project.

Step	Action	Result
1	<p>Verify that the instances are displayed in tree view mode in the Instances pane, else click the toggle button, page 343.</p> <p>NOTE: Alternatively, you can assign instances using the grid view mode. The assignment principle remains unchanged. You can switch between views at any time during assignment by clicking the toggle button.</p>	The Assignment Editor displays the valid, page 177 application instances in tree view mode.
2	In the Containers pane, select the Supervision.	<ul style="list-style-type: none"> • The name of the project appears in the title bar of the Assignments pane. • Any existing containers and the facets already assigned to them are displayed.
3	From the Instances pane, drag one or more objects, page 346 to the title bar of the Assignments pane.	The Assignment Viewer window, page 372 opens.

Step	Action	Result
	NOTE: Alternatively, you can drag objects directly to the project in the Containers pane.	
4	Review the information that is displayed, edit the container identifier as needed, and select the cluster.	—
5	Click OK .	<ul style="list-style-type: none">• Containers that do not exist yet are created.• Supervision data facets that have a status of Candidate are assigned to the respective container and displayed in the Assignments pane.• The status of the assigned facets in the System column of the Facets pane changes to Assigned if the status was Not assigned.• The status in the Project column of the Facets pane changes from Candidate to Done.

NOTE: If an instance contains a facet of type *ClientEvent* or *ServerEvent*, the facet is assigned to the **ClientEvents** and **ServerEvents** containers, page 302 respectively.

NOTE: By default, remaining facets will be assigned automatically, page 381 when you generate the project or a section.

Assigned Facet Description

Facet Description

The **Assignments** pane, page 343 of the **Assignment Editor** provides information on the facets that are assigned to an FBD section or tag container.

To see a list of facets that are assigned to a project, select the project in the **Containers** pane.

The following information is displayed in the **Assignments** pane.

Column header	Description
Container	Identifier of the section/container to which the facet is assigned.
Instance	Identifier of the instance that references the facet.
Instance Template	Identifier of the template that the instance is using.
State	State of the facet in the context of the instance. Possible values: <ul style="list-style-type: none"> • Valid • Invalid, page 186 • Deleted: The section is generated, page 396 and you have unselected the corresponding element of the instance in the Instance Editor or deleted the instance itself.
Facet	Unique identifier of the facet.
Facet Template	Identifier of the template that the facet is using.
Path	Path to the facet in the composition of the instance, page 182.
Order	This column applies to Control projects only. Indicates the order in which the software generates constituents of the facets inside the section and as such, the order in which the Control Participant executes the constituents inside the section. You can change the order of facets, page 394 inside a section. The software applies the order only during: <ul style="list-style-type: none"> • A first generation, page 401 • A regeneration, page 434 For more detailed information on how the software manages the facet order, refer to the description of the generation functionality, page 403.
Assignment	Indicates the status of the facet in relation to the instance that references the facet. Possible values: <ul style="list-style-type: none"> • Assigned • Unassigned • Out Of Date • Deleted Refer to <i>Assignment Status of Generated Facets</i> , page 425 for a description of the assignment status.
Generation	Indicates the status of the facet in relation to the generation of its constituents in the logical Participant project. Possible values: <ul style="list-style-type: none"> • Non Generated • Generated • Moved • Inconsistent • Unlinked Refer to <i>Generation Status of Facets</i> , page 424 for a description of the generation status.

Changing the Element Selection of Instances After Assignment

If you change the Control or Supervision element selection of an instance in the **Instance Editor** window after assigning facets of the instance (before generating), the changes are reflected in the respective **Assignment Editor**.

The table describes how the changes to the element selection of an instance are reflected in the **Assignment Editor**.

When you...	Then...
Select a new element of an instance	The corresponding facets are added to the existing facets of the instance and you can display them in the Facets pane of the Assignment Editor . Proceed with the assignment as described in the assignment procedure.
Unselect an element of an instance	The corresponding facets are removed from the list of facets in the Facets pane of the Assignment Editor . If you had already assigned the facets, then they are also removed from the corresponding container.

NOTE: Changes impacting the assignment status of facets that you have generated are described in *Assignment Status of Generated Facets*, page 425.

Automatic Assignment of Remaining Facets

Overview

The **Project Explorer** features a functionality that allows you to assign remaining Control and Supervision candidate facets of instances, which have at least one facet that is already assigned to a Participant project. The assignment is done automatically when you select a [generation command](#), [page 399](#). The container to which these facets are assigned depends on where the other facets of the instance are already assigned.

There can be candidate facets remaining because you have:

- Not assigned all the facets to a Participant project yet.
- Unassigned facets.
- Changed the [element selection](#) of the instance, [page 188](#) after assignment, which has created new facets.

Information on facets that are assigned automatically is displayed in the notification panel.

NOTE: Automatic assignment does not apply to Supervision genie facets.

Using and Disabling the Functionality

The functionality is enabled by default and remains enabled until you disable it in the settings of a [Control project](#), [page 272](#) or [Supervision project](#), [page 300](#).

Assignment is triggered, [page 399](#) when you use any generation command at the project or project container level.

NOTE: You can still assign facets manually while the functionality is enabled.

Automatic Assignment Criteria

The table describes how remaining candidate facets of an instance are assigned, depending on where other facets of the instance are already assigned.

Candidate facet type	Facets of same instance are already assigned to one container	Facets of same instance are already assigned to several containers
Control facet	Candidate facets are assigned to the same FBD section.	Candidate facets are assigned to the FBD section to which other facets of the instance are already assigned and that has the lowest Order number.
Supervision facet	Candidate facets are assigned to the same tag container.	<ul style="list-style-type: none"> • Supervision project with one cluster: Candidate facets are assigned to the tag container that was created first. • Supervision project with several clusters: Candidate facets are assigned to each cluster. In each cluster, candidate facets are assigned to the tag container that was created first.

NOTE: Only facets that are available are assigned. If a facet is locked, for example, because the instance referencing it is open in the **Instance Editor**, it remains unassigned.

Viewing Assignments

To verify to which container facets of an instance are assigned, use the **View Assignments**, [page 392](#) window.

Examples

In this example, three facets of instance *Motor_1* are already assigned to section *FBDSection_1*. A modification of the element selection of *Motor_1* has created facet *Motor_1_DEVMNT*, which is candidate to be assigned. EcoStruxure Process Expert assigns *Motor_1_DEVMNT* automatically to section *FBDSection_1* when you generate the project. (Information has been highlighted in yellow for this example.)

ControlProject_1: Assignment Editor

Motor_1 - Facets									
Instance	Instance Template	Facet	Facet Template	Path	System	Project	Type		
Motor_1	\$Motor	Motor_1_DEVCTL	SDEVCTL_UL	ControlMotor	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM	SCONDSUM_UL	ControlFailures	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM1	SCONDSUM1_UL	ControlInterlocks	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_DEVMNT	SDEVMT_UL	ControlMaintenance Not Assigned	Candidate		Logic		

ControlProject_1 - Containers

FBDSection_1 - Assignments

Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_DEVCTL	SDEVCTL_UL	ControlMotor	0	Assigned	Non Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM	SCONDSUM_UL	ControlFailures	1	Assigned	Non Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM1	SCONDSUM1_UL	ControlInterlocks	2	Assigned	Non Generated

Motor_1 - Facet

Instance	Instance Template	Facet	Facet Template	Path	System	Project	Type		
Motor_1	\$Motor	Motor_1_DEVCTL	SDEVCTL_UL	ControlMotor	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM	SCONDSUM_UL	ControlFailures	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM1	SCONDSUM1_UL	ControlInterlocks	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_DEVMNT	SDEVMT_UL	ControlMaintenance Assigned	Done		Logic		

FBDSection_1 - Assignments

Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_DEVCTL	SDEVCTL_UL	ControlMotor	0	Assigned	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM	SCONDSUM_UL	ControlFailures	1	Assigned	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM1	SCONDSUM1_UL	ControlInterlocks	2	Assigned	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_DEVMNT	SDEVMT_UL	ControlMaintenance 3		Assigned	Generated

In this example, three facets of instance *Motor_1* have been assigned to section *FBDSection_1*. The section was generated and the three facets unassigned. Thereafter, modification of the element selection of *Motor_1* has created facet *Motor_1_DEVMNT*, which is candidate to be assigned. EcoStruxure Process Expert assigns *Motor_1_DEVMNT* automatically to section *FBDSection_1* when you generate the project again while the three unassigned facets are removed from it. (Information has been highlighted in yellow for this example.)

ControlProject_1: Assignment Editor

Motor_1 - Facets									
Instance	Instance Template	Facet	Facet Template	Path	System	Project	Type		
Motor_1	\$Motor	Motor_1_DEVCTL	SDEVCTL_UL	ControlMotor	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM	SCONDSUM_UL	ControlFailures	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM1	SCONDSUM1_UL	ControlInterlocks	Assigned	Done	Logic		
Motor_1	\$Motor	Motor_1_DEVMNT	SDEVMT_UL	ControlMaintenance Not Assigned	Candidate		Logic		

ControlProject_1 - Containers

FBDSection_1 - Assignments

Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_DEVCTL	SDEVCTL_UL	ControlMotor	0	Unassigned	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM	SCONDSUM_UL	ControlFailures	1	Unassigned	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM1	SCONDSUM1_UL	ControlInterlocks	2	Unassigned	Generated

Motor_1 - Facets

Instance	Instance Template	Facet	Facet Template	Path	System	Project	Type		
Motor_1	\$Motor	Motor_1_DEVCTL	SDEVCTL_UL	ControlMotor	Not Assigned	Candidate	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM	SCONDSUM_UL	ControlFailures	Not Assigned	Candidate	Logic		
Motor_1	\$Motor	Motor_1_CONDSUM1	SCONDSUM1_UL	ControlInterlocks	Not Assigned	Candidate	Logic		
Motor_1	\$Motor	Motor_1_DEVMNT	SDEVMT_UL	ControlMaintenance Assigned	Done		Logic		

FBDSection_1 - Assignments

Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generated
FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_DEVMNT	SDEVMT_UL	ControlMaintenance 0		Assigned	Generated

Automatic Assignment of Supervision Facets

Overview

This topic describes the functionality that is enabled when you associate a Control project to a tag container by using the **Associated Control Project** property of the tag container, page 353.

Working Principle

When you associate a tag container to a Control project, whenever you drag instances (or a folder) to the **Containers** pane in the **Assignment Editor** to assign their Control facets to this Control project, the Supervision data facets of these instances are automatically assigned to the associated tag container.

You can see the details of the Supervision facet assignment in the notification panel.

The automatic assignment of Supervision facets is not performed in the following cases:

- When you drag Control facets from the **Facets** pane.
- When remaining candidate Control facets are automatically assigned, page 383.
- When Control facet assignments are created by means of import.
- The instance has no Supervision data facets. In this case, an entry for unsuccessful facet assignment appears in the notification panel.

Visual Indicator

The following partial view of the **Assignment Editor** shows two tag containers where only *TagContainer_1* is associated to a Control project. It is indicated by a blue icon.

Path	Container
\	Supervision_1
\Supervision_1\Cluster_1\Tags	TagContainer_1
\Supervision_1\Cluster_1\Tags	TagContainer_2

Further, when you hover with the pointer over the tag container, a tooltip indicates the identifier of the Control project.

Locked Supervision Projects

If the Supervision project that contains the tag container is locked (for example, because it is open for refinement) when EcoStruxure Process Expert attempts to assign the Supervision facets, a dialog box opens. It gives you the following choices:

- **Yes:** The dialog box closes and EcoStruxure Process Expert will attempt to assign both the Control and the Supervision facets again after waiting for approximately 1 min.
 - If, at that moment, the Supervision project is still locked, the dialog box opens again.
 - If the lock has been released, both facets types are assigned without further notification. You can see the details of the facet assignment in the notification panel.

During the wait time, certain actions that you or other users perform on the Control and the Supervision projects are queued without further notification. EcoStruxure Process Expert performs these actions automatically once the initial facet assignment is completed. You can see the details of these actions in the notification panel once they are completed. Such actions can be, for example, assigning additional facets to existing tag containers.

Other actions may be forbidden and a dialog box opens to inform you. For example, refining either project or creating new containers.

- **No:** Only the Control facets are assigned and you can resume working normally. You need to assign the corresponding Supervision facets manually. If you click **No** the second time the dialog box opens, actions that are queued will be performed in sequence.
- **Cancel:** No assignment is performed.

NOTE: If the instance whose Control facets you are assigning has no Supervision data facets to be assigned and the Supervision project is locked, the dialog box still opens. In this case, click **No**.

NOTE: If the dialog box opens and the Supervision project has already been released when you click **Yes**, the wait time still needs to elapse before EcoStruxure Process Expert performs the assignment of facets.

Unassigning Control Facets

When you unassign Control facets, the corresponding Supervision facets that were assigned automatically remain assigned.

Impact on Supervision Project Export and Import

The value that is selected for the **Associated Control Project** property is exported.

When you import into a system a Supervision project, which has a tag container that is associated to a Control project, the value of the **Associated Control Project** property after import is as follows:

- If the system contains no Control project with the same identifier, the value is reset.
- If the system contains a Control project with the same identifier, the value is kept and automatic assignment of Supervision facets is enabled.

Viewing the Application Structure in Control Projects

Overview

Switching to tree view mode in the **Containers** pane, page 343 of the **Assignment Editor** allows you to view FBD sections based on the location of instances whose Control facets are assigned to these sections.

It shows the relationship between the folder structure of the application and the sections of a Control Participant project. This is useful, for example, when you want to view the logic related to a specific area or subdivision of the application or

verify that the Control facets of a group of instances are assigned to the right sections of a project.

Any FBD section to which a Control facet is assigned appears in the tree view. A section can appear several times depending on the facets it contains.

Independently of where and how many times an FBD section appears, its contents, which is visible in the **Assignments** pane, is the same.

Changing the application folder structure, the location of instances, facet assignments, and/or the status of facets has an impact on the position of sections in the tree view.

Empty FBD sections and non-FBD sections are shown as well.

NOTE: You can view the application structure also when refining Control Participant projects, page 448 and making changes online, page 875.

Root Folder Description

In tree view mode, up to three root folders can appear in the **Containers** pane.

Root folder identifier	Description
<system identifier> (for example, <i>System_1</i>)	<p>The <system identifier> (for example, <i>System_1</i>) root folder is the parent folder for the application structure. This structure contains sections to which facets with the following status are assigned:</p> <ul style="list-style-type: none"> • Assigned • Out Of Date • Unassigned • Non Generated • Generated • Moved • Inconsistent
EmptyContainers	<p>Parent folder for sections to which no facet is assigned because you have performed one or more actions such as:</p> <ul style="list-style-type: none"> • Created the section manually. • Unassigned all non-generated facets from the section. If the facets were already generated, you have unassigned all facets and generated the section. • Disabled a service, which removed the corresponding non-generated facet. If the facet was already generated, you have disabled the service and generated the section. • Deleted any instance referencing a non-generated facet assigned to the section. If the facets were already generated, you have deleted the instances and generated the section. • Moved all facets to another section. • The section is a non-FBD section.
DeletedFacetContainers	<p>Parent folder for sections to which at least one facet is assigned with the status <i>Deleted</i>.</p> <p>If the section contains at least one facet with another status, the section also appears under the <system identifier> root folder.</p> <p>Once you generate the section and the <i>Deleted</i> facet is removed from the section, the section appears only under either of the two other root folders depending on the facets that are assigned to it.</p> <p>No application folders are shown under this root folder.</p>

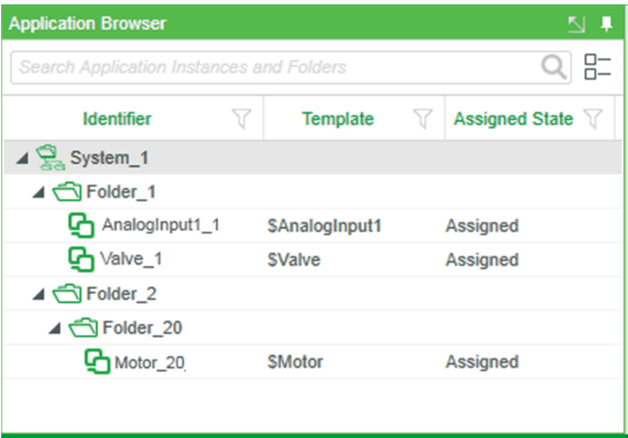
NOTE: A root folder appears only if there is a section in its folder structure. For example, if there is no empty section, the **EmptyContainers** root folder is not shown.

Relation Between Folder Structure and FBD Sections

A folder structure can contain several sections if facets referenced by instances in that folder structure are assigned to multiple sections.

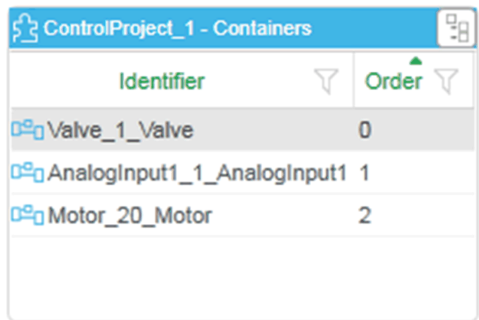
A section can be shown multiple times under different folder structures if the facets assigned to this section are referenced from instances located in various folders.

In this example, the application contains three folders. The Control facets referenced by each instance are assigned to separate FBD sections. *Folder_2* is empty.



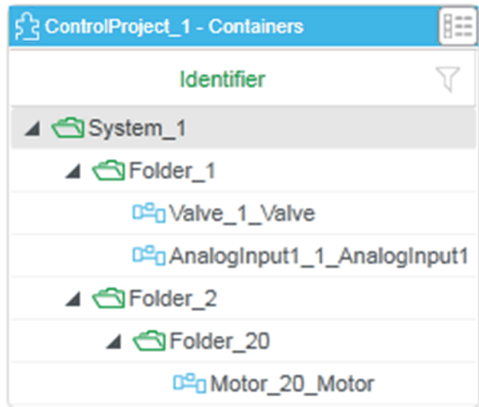
Identifier	Template	Assigned State
System_1		
Folder_1		
AnalogInput1_1	\$AnalogInput1	Assigned
Valve_1	\$Valve	Assigned
Folder_2		
Folder_20		
Motor_20	\$Motor	Assigned

The following is an extract of the **Assignment Editor**, which shows the corresponding FBD sections in the **Containers** pane of the Control project in *grid view* mode. The project node is not shown.



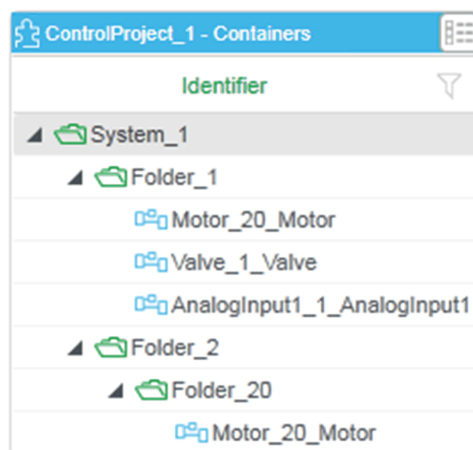
Identifier	Order
Valve_1_Valve	0
AnalogInput1_1_AnalogInput1	1
Motor_20_Motor	2

The following figures show how the FBD sections are displayed in relation to the folder structure of the application in the **Containers** pane in *tree view* mode.



Identifier
System_1
Folder_1
Valve_1_Valve
AnalogInput1_1_AnalogInput1
Folder_2
Folder_20
Motor_20_Motor

If during assignment, you had assigned one facet of instance *Valve_1* also to section *Motor_20_Motor*, this section would appear twice. Under *Folder_20* because of facets of instance *Motor_20* and under *Folder_1* because of the facet of *Valve_1*. The following figure shows this.



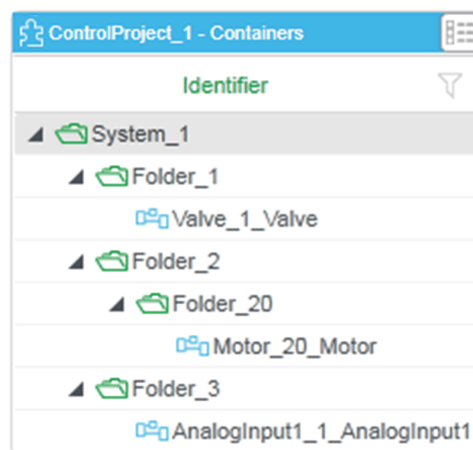
Impact of Changes in the Application on the Tree View

In tree view mode, changes to the application are reflected.

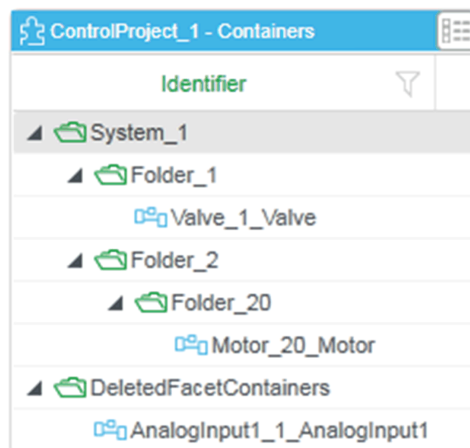
Changes that you make to the folder structure of the application are reflected in folder structure shown in the **Containers** pane.

Changes that you make to the location of instances whose facets are assigned to sections are reflected in the position of these sections in the **Containers** pane.

Based on the application of the previous example, if you create *Folder_3* and move instance *AnalogInput_1* to this folder, the tree view is modified as shown in the figure.



If you delete instance *AnalogInput_1* and its facets were generated, the status of the facets changes to **Deleted**. When this happens, section *AnalogInput_1_AnalogInput* appears under the **DeletedFacetContainers** root folder as shown in the figure.

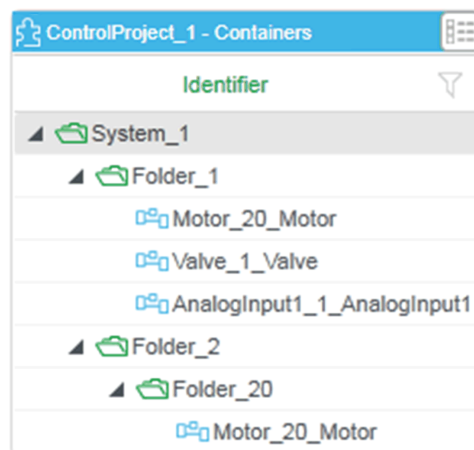


NOTE: *Folder_3* is empty and therefore it does not appear in the **Containers** pane.

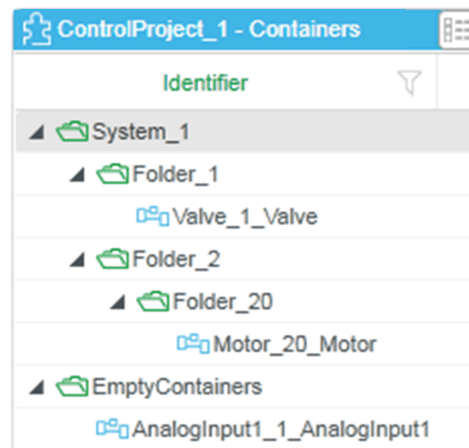
Impact of Changes to Assignments on the Tree View

Changes that you make to the assignment of facets are reflected in the position of sections in the **Containers** pane.

If without changing the application, you move a facet of the *Valve_1* instance to the *Motor_20_Motor* section, the section appears also under *Folder_1* because the instance is located in *Folder_1*.



If you had generated section *AnalogInput_1_AnalogInput*, delete the instance, and generate the section again, the **Deleted** facets are removed from the section and the empty section now appears under the **EmptyContainers** root folder. The result is shown in the following figure.



Managing Assigned Facets

Overview

This section describes the actions that you can perform on assigned facets by using the **Project Explorer** and the **Application Explorer**.

It also describes how to access containers to which facets are assigned and how to open the **Assignment Editor** from facets that you have not yet assigned to a project.

As part of the *simplified workflow*, page 69, it describes how to edit, link, and copy instances by using context menu commands of their assigned facets.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Facet Actions After Assignment

Overview

This topic applies to facets that are assigned but not yet generated.

However, instance-related commands that pertain to the simplified workflow, page 69 are available also for already generated facets.

For information on facets that are generated, refer to the topic describing management of generated facets, page 442.

Commands and Actions

Once you have assigned Control or Supervision data facets of instances to the respective project, you can execute the following commands and actions on these facets from the **Assignments** pane of the **Assignment Editor**.

Command/Action	Description
Unassign	<p>Removes the facet from the section or container.</p> <p>You cannot unassign facets that have the Assignment status of Deleted.</p> <p>Refer to <i>Unassigning Facets</i>, page 391.</p> <p>NOTE: The actions that EcoStruxure Process Expert executes when you select the Unassign command for a non-generated facet is different than if you execute the command on a <i>generated facet</i>, page 443.</p>
Edit Instance	<p>Opens the instance that references the facet in the Instance Editor inside the Project Explorer, which lets you edit the instance.</p> <p>When you save, changes to Control services are automatically reflected in the section. That is, if, for example, you enable/disable a service, the corresponding facet is automatically assigned/unassigned.</p> <p>The command is only available for Control facets.</p>
Edit Links	<p>Opens the instance that references the facet in the Links Editor inside the Application Explorer, which lets you link the instance to other instances.</p> <p>When you close the Links Editor, changes pertaining to links that you have created, modified, or removed are automatically reflected in the section.</p> <p>The command is only available for Control facets.</p>
Go To Instance	<p>Shows the instance that references the facet in the Application Explorer.</p> <p>The navigation feature is not available if the Assignment status of the facet is Deleted.</p> <p>Refer to the topic describing how to <i>navigate from facets to the instance</i>, page 395.</p>
Copy Instances	<p>Adds the instance that references the facet to the Clipboard. This lets you create copies of the instance and assign their facets automatically by using the Paste Instances FBD section context menu command, page 348.</p> <p>If you select several facets belonging to different instances, each instance is added to the Clipboard.</p> <p>The command is only available for Control facets.</p>
Move the facet (action)	<p>Changes the assignment of a facet to another FBD section or tag container. Does not apply to Genie facets, page 509.</p> <p>Refer to <i>Moving Facets</i>, page 391.</p>

NOTE: The **Reassign**, page 444 and **Unlink**, page 443 commands that you can access from the facet context menu become available only once you have completed the generation stage.

Unassigning Facets

To unassign a non-generated facet from a section or container, in the **Assignments** pane of the **Assignment Editor**, right-click the facet and select **Unassign**.

Result: The facet is removed from the **Assignments** pane and:

- The status of the unassigned facet displayed in the **System** column in the **Facets** pane changes from **Assigned** to **Not assigned** if the facet is not assigned to any other project of the system.
- The status of the **Project** column changes from **Done** to **Candidate**.

NOTE: You can select multiple facets to unassign them at once.

Moving Facets

To move a non-generated facet from one FBD section/tag container to another, proceed as follows.

Step	Action	Result
1	In the Assignments pane of the Assignment Editor , right-click any one of the column headers to clear filters and sorting, page 120.	-
2	Drag the facet to another FBD section of the same task folder or tag container that is shown in the Containers pane. NOTE: You can select multiple facets to move them at once.	<ul style="list-style-type: none"> • The facet is removed from its original container and assigned to the target one. • The assignment status of the facet is unchanged.

NOTE: Control facets that you move are assigned to the new section with the highest order value in the **Order** column of the **Assignments** pane, page 343. When you move several facets at once, the facet that you have selected first gets the lowest order out of the ones that are moved.

Viewing Facet Assignments

Overview

By opening the **View Assignments** window for an application instance, you can verify if Control facets and Supervision (data and genie) facets are assigned to a project of the system.

For each facet, you can:

- If the facet is not yet assigned: Open the corresponding **Assignment Editor**, page 342 for existing projects of the system. For genie facets, you can open the **Pages** container of existing Supervision projects.
- If the facet is already assigned: Open each container to which it is assigned and locate the facet.

Opening the View Assignments Window

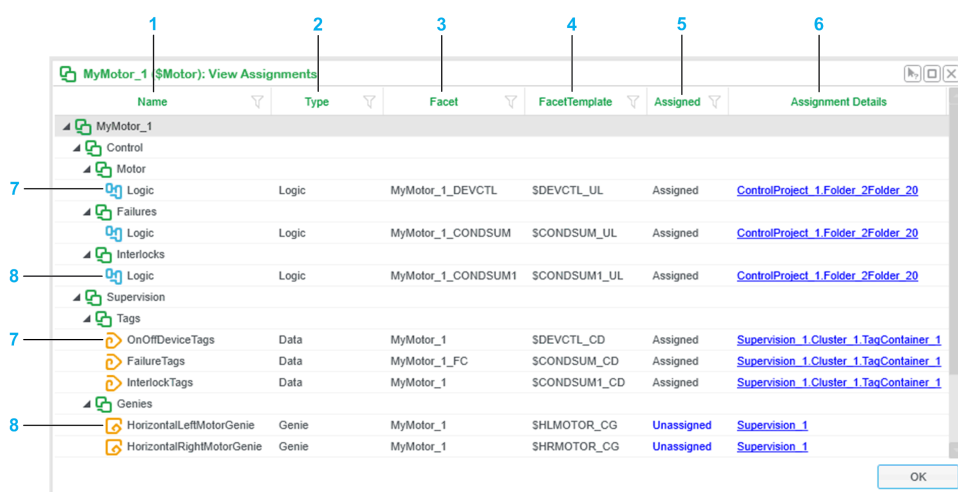
In the **Application Browser** pane, page 160, right-click an instance and select **View Assignments**.

NOTE: You cannot open the **View Assignments** window if both conditions are true for the instance:

- **Assigned** status is **Unassigned**.
- **Data** or **Link** status is **Invalid**.

View Assignments Window

The following figure shows an example of the **View Assignments** window for instance *MyMotor_1*, whose Control and Supervision data facets are assigned but not its genie facets.



Item	Description
1	Displays the elements of the selected instance grouped by Participant. For each Participant, elements are grouped by one or more levels of categories. For example, category Control and subcategory Interlocks contain element Logic .
2	Indicates the subtype of the facet.
3	Identifier of the facet that corresponds to the element.
4	Identifier of the template that the facet is using.
5	Facet assignment status. Possible values: <ul style="list-style-type: none"> • Unassigned: The facet is not yet assigned to any project of the system.

Item	Description
	<ul style="list-style-type: none"> • Assigned: The facet is assigned to, at least, one project of the system.
6	<p>For Control facets, the following information is displayed:</p> <ul style="list-style-type: none"> • For unassigned facets: A list of existing Control projects in the system to which you can assign the facet. Click a project in the list to open its Assignment Editor. • For assigned facets: The list of sections to which the facet is assigned. Section identifiers are shown in the format: <code>ProjectIdentifier.SectionIdentifier</code>. Click a section to show or open the Assignment Editor. The section and facet are highlighted. <p>NOTE: For unassigned facets, if no Control project exists in the system, the column is empty.</p>
6	<p>For Supervision data and genie facets, the following information is displayed:</p> <ul style="list-style-type: none"> • For unassigned facets: A list of existing Supervision projects in the system to which you can assign the facet. Click a project in the list to open its Assignment Editor. • For assigned facets: The list of containers to which the facet is assigned. Container identifiers are shown in the following format format: <ul style="list-style-type: none"> ◦ For data facets: <code>ProjectIdentifier.ClusterIdentifier.ContainerIdentifier</code>. ◦ For genie facets: <code>ProjectIdentifier.PageIdentifier</code>. <p>Click a container to show or open the Assignment Editor. The container and facet are highlighted.</p> <p>NOTE: For unassigned facets, if no Supervision project exists in the system, the column is empty.</p>
7	Elements of the instance, which generate a facet and that are shown when you open the Instance Editor .
8	Elements of the instance, which generate a facet and that are shown when you inspect the instance, page 211.

NOTE: The **View Assignments** window is closed automatically when you navigate to the **Assignment Editor**.

Order of Control Facets

Overview

EcoStruxure Process Expert assigns an order, [page 379](#) to facets that you assign to an FBD section, starting at 0, and displays it in the **Assignments** pane. The values appear in the **Order** column.

The order corresponds to the execution order of the constituents of the facet by the Control Participant inside the section after you generate the section.

You can change the order of facets inside an FBD section.

EcoStruxure Process Expert applies the default order or changes to this order only until the [first generation, page 403](#) of the section. After you have generated the section a first time, you can still change the order of facets inside the section. However, the change has no effect on the order of already generated facet constituents if you generate the section again. You need to [regenerate, page 434](#) the section for EcoStruxure Process Expert to apply the new facet order.

Changing the Facet Order

To change the order of Control facets inside an FBD section, proceed as follows.

Step	Action
1	In the Containers pane, select the FBD section that contains the facets that you want to rank. Result: The facets that are assigned to the selected FBD section appear in the Assignments pane.
2	In the Assignments pane, right-click any one of the column headers and clear filters and sorting, page 120 . Result: The facets are displayed in increasing order from top to bottom.
3	Drag a facet up or down to a new position until you see a tooltip, which indicates the position that the facet will occupy when you release the mouse button. Result: The facet appears in the selected position with a new order value.

NOTE: To change the execution order of the FBD sections, refer to the topic describing how to [change the order of sections, page 275](#).

Navigating from Facets to the Instance

Overview

From a facet that is shown in the **Assignments** pane of the **Assignment Editor**, you can open the **Application Explorer** to show the instance that references this facet.

The navigation feature is available for:

- Control facets.
- Supervision data and genie facets.

NOTE: This functionality complements the commands that let you edit and link instances, page 390 from their Control facets as part of the simplified workflow, page 69.

Conditions Impacting Navigation to the Instance

The table describes the scenarios where navigation to the instance is limited or not possible.

Condition	Result
The Assignment status of its facet is Deleted .	The Go To Instance command is not available.
The instance is not shown in the Application Browser because a filter, page 120 is applied.	The Application Explorer opens or is shown but a dialog box informs you that the instance is not visible.
The Application Browser pane is minimized or in the background (if tabbed).	The Application Browser pane remains minimized or in the background. If the instance is open in the Instance Editor , the corresponding tab is shown.

Navigating to the Instance

To navigate from a facet assigned to a project container to the instance that references this facet, rick-click the facet and select **Go To Instance**.

Result:

- The **Application Explorer** window opens if it is not yet open. If it is, the window is shown.
- The application folder containing the instance expands to select the instance that references the facet.

Generation Stage

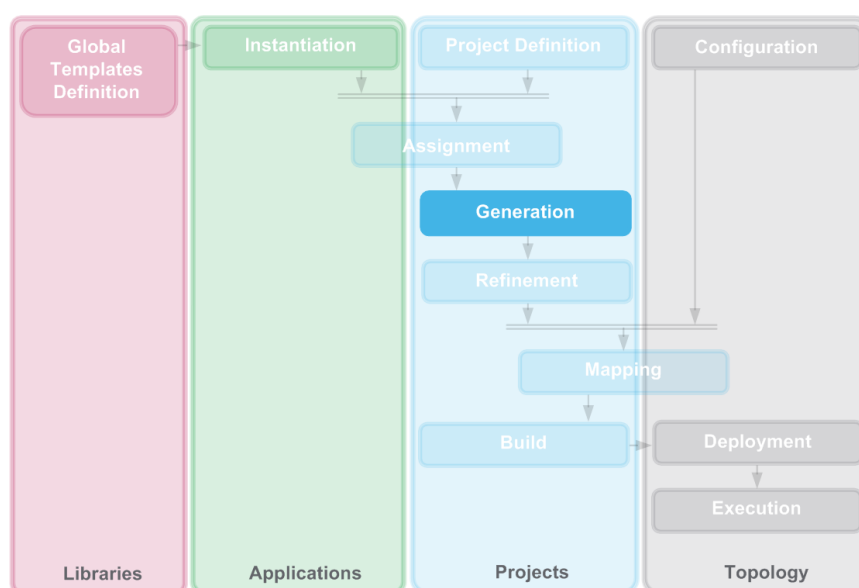
What's in This Chapter

General Description.....	397
Generating a Project for the First Time	426
Generating Projects After Changes.....	429
Generating Equipment in Supervision Projects	437
Generation Diagnostic Messages	440
Managing Generated Facets	442

Overview

This chapter describes how to create the logical Control and Supervision Participant projects of the system.

The following figure shows the position of the **Generation** stage within the system engineering life cycle.



Refer to the Generation stage, page 58 for a description of the purpose of this stage.

General Description

Overview

This section gives a general description of the generation process of Control and Supervision Participant projects.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Generation Process Description

Overview

Once you have assigned Control and/or Supervision facets to the containers of their respective projects, you need to generate these projects.

During the system engineering life cycle, the generation process of Participant projects can encompass up to three steps:

- A first-time generation, page 426.
- A generation after changes, page 429.
- A Control Participant project regeneration, page 434.

You can generate the entire projects or parts of it.

Control projects can be generated automatically, page 407.

The generation process creates the logical Participant project.

For Control projects the generation process:

- Creates the derived function blocks (DFBs) and variables based on the constituents that are encapsulated inside the facets according to the assignment of the facets in the structure of the project and their properties, page 182.
- Creates links between function blocks generated from different facets, which are assigned to the same FBD section, page 402. In addition to links that are defined at the template level, links created in an editor of the **Application Explorer** are processed.

For Supervision projects, the generation process populates the native Supervision Participant database files for:

- Tags from Supervision data facets.
- Report events from Supervision server event facets.
- Events from Supervision client event facets.
- Supervision project components.
- User interface messages.
- Equipment, page 437.

NOTE: You can abort the generation of projects, page 427 and their containers, page 428 as well as the regeneration, page 435.

Data Validation Before Generating

When you select the **Generate** or **ReGenerate** command for a project or a container, EcoStruxure Process Expert first verifies if the constituents of the assigned facets satisfy the applicable naming rules, page 31.

EcoStruxure Process Expert operates as follows when it detects that the applicable naming rules are not satisfied:

- For Control facets, it does not generate containers nor constituents in the logical Participant project even those that satisfy the naming rules.
Refer also to the topic describing the impact of detected errors on generation, page 402.
- For Supervision facets, it generates only the constituents that satisfy the applicable naming rules.
- For both facet types, it displays a diagnostic message, which contains the identifiers of the containers, assigned facets, and their parameter values that do not satisfy a naming rule.

Generating Facets of the Same Instance Assigned to Several Projects

You can generate several projects simultaneously even when facets of the same instance are assigned to these project.

The projects can be Control and/or Supervision projects.

Generating Automatically Assigned Facets

By default, EcoStruxure Process Expert assigns candidate facets automatically, [page 381](#) when you select any generation command at the project or project container level.

If among the instances whose facets are already assigned to the Participant project, at least one facet remains with the status **Candidate**, a dialog box opens to remind you that the functionality is enabled before executing the command. You have the possibility to cancel generation and disable the automatic assignment of these facets.

If EcoStruxure Process Expert is unable to proceed with the automatic assignment of a facet, generation of assigned facets completes but the notification panel displays information on the facets that it could not assign and generate.

If generation does not complete successfully, facets that were assigned automatically are unassigned.

NOTE: Generation takes a little longer to complete when the functionality is enabled.

Generating During Build Stage

To save time or if you have made minor changes to a logical Participant project, instead of using the **Generate** command, you can use the **Generate and Build** command from the [Control executable, page 721](#) or [Supervision executable, page 764](#) context menu. This performs a generation followed by a build. You need to have completed the system engineering life cycle up to the build stage.

If you generate a complete Participant project and generation of any section, [page 402](#) or generation of any container does not complete, build is not started. Information about the sections/containers that were not generated is displayed in the notification panel.

If build is not started or does not complete successfully, the result of the generation process is the same as if you had selected the **Generate** command for the Participant project. This means, for example, that for a Control Participant project, sections that were successfully generated remain generated.

The **Generate and Build** command:

- Cannot be used to perform a regeneration, [page 434](#).
- Can be used together with the automatic facet assignment functionality, [page 381](#).
- Automatically performs a **Build** or **Build All** command, [page 773](#) depending on the type of change that you have made to the Participant project.

NOTE: After selecting the command, the Abort icon is displayed in the notification panel, [page 85](#). Click the icon to cancel the task. Only the user who selected the command is allowed to cancel it.

If you cancel the task:

- Before generation completes, the entire generation task is canceled and no sections are generated. Build does not start.
- After generation completes, only the build task is canceled and a notification is displayed to inform you.

If you are generating and building a small project, the whole task may complete before you are able to click the icon.

Generating Facets with State Invalid

If the **State** of a facet, page 379 has become **Invalid** after you have assigned it, generating the container or Participant project removes the facet from the container.

This can be the case, for example, when you link instances A and B (where instance A needs to be linked to be valid), assign their facets to a Control project, and then delete the link between the two instances. The facet of instance A becomes invalid because the instance is not linked anymore.

Generating the section to which the invalid facet is assigned removes it from the section.

Skipping Facets During Generation

If a facet is locked, the constituents that it encapsulates cannot be generated. In this case, the locked facets are skipped but generation of the section or the Participant project completes for the other facets. This may happen, for example, when an instance is open in the **Instance Editor**.

The generation status of a facet that is skipped does not change.

The notification panel contains information about locked facets that could not be generated.

Control Project Specifics

Overview

This topic describes the default actions when generating Control Participant projects and the settings that you can configure to customize the generation process.

Automatic Generation Setting

The **Generation** setting, page 272 of each Control project lets you enable the automatic generation of sections of the project when the facets that are assigned to them undergo a change.

Identifying Code Generated by Control Facets

Constituents that are generated from facets assigned to Control projects are locked in the Control Participant project. The locked code is color coded in the DFB editor of the Control Participant when you refine the project, page 452.

Validity Indicators

When you use a generation command on a logical Control Participant project or its containers (including the **Generate and Build** command) and project analysis is enabled, page 272, EcoStruxure Process Expert performs an analysis.

If the result of the analysis generates an advisory or detects an error, a validity icon, page 336 is displayed.

In this case, refine, page 446 the Control Participant project, identify, and fix the issue. When you save your refinement changes, EcoStruxure Process Expert performs a new analysis. The result is shown in the **Analyze Project** tab of the Control Participant window.

NOTE: EcoStruxure Process Expert may detect inconsistencies that are linked to the default **configuration** in the Control Participant, which is part of each Control project but that is not relevant in the context of the **Project Explorer** (except for Control projects of M580 safety controllers, page 912). These detected errors do not prevent you from building the project, page 775 if you configure the controller entity in the **Topology Explorer** or the workstation in the **Project Explorer** appropriately.

This can be the case, for example, if the application contains an instance of a communication port template, which requires that you enter a **%MW** holding register starting address. During generation, this address is compared to the default address in the controller module of the default **configuration** of the Control project. If both addresses are not consistent, a validity icon is displayed. When you build the Control Participant project, the **%MW** address is compared to the address in the controller module in the **Topology Explorer** and needs to match.

It can also happen, if the logical Control Participant project is missing IODDT variables (for example, when the configuration contains HART modules, page 606) or Device DDT variables, page 872, which exist only in the configuration project until you perform the build or create them, page 455.

If fixing the inconsistency requires that you change a value in the controller module of the default **configuration**, ensure that the value is configured correctly for the controller module in the **Topological Explorer** by using the **Configure** command for the controller to which the Control executable will be mapped (service mapping).

Generating Graphical Links

Inside the same FBD section, EcoStruxure Process Expert generates graphical links between function blocks that use the same variable and that are encapsulated inside different facets only if the following conditions are satisfied:

- The variable is created by a facet (not created manually by the user).
- The resulting link:
 - Links the output pin and input pin of separate function blocks.
 - Is not a circular link.

EcoStruxure Process Expert breaks any graphical circular link and creates it as variable link.

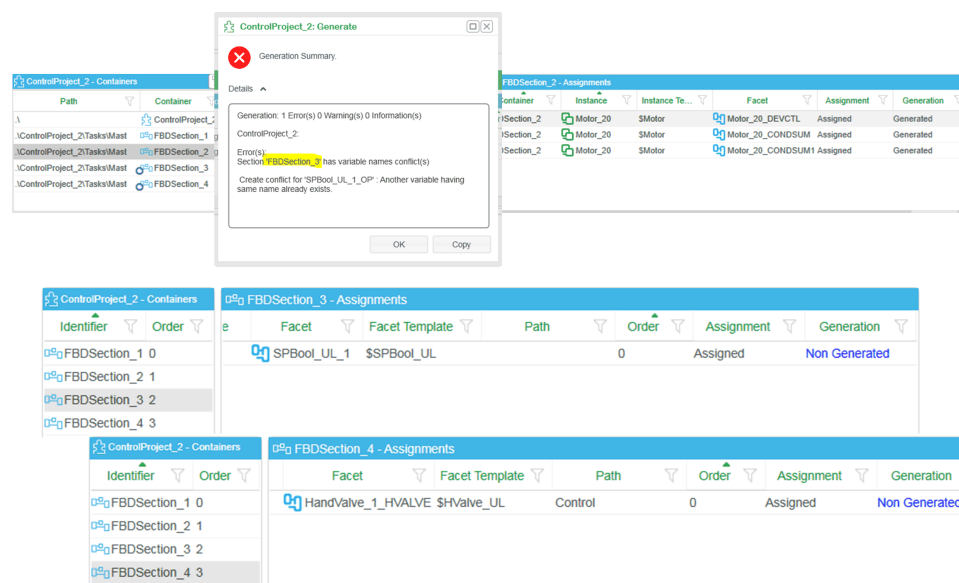
Impact of Detected Errors or User Action During Generation

When you select the **Generate** or **Generate and Build** command for a project containing several sections and an error is detected, the impact on the generation process depends on the detected error or the user action that is performed.

NOTE: The behavior is the same when you regenerate, page 434 a Control Participant project.

Detected error or user action	Impact on the generation process
Variable name conflict ⁽¹⁾	<p>Generation stops. Code of sections that have been successfully generated remains in the logical Control Participant project.</p> <p>The section which contains the detected error and any subsequent sections are not generated.</p> <p>A summary dialog box indicates the identifier of the section where generation stopped as well as the reason.</p>
Type version conflict ⁽¹⁾	
Naming rule conflict, page 31	
Clicking Cancel in the Type Version Conflict dialog box, page 409.	
Operation in the Control Participant did not succeed (for example, injecting data in a section through import).	
.NET exception occurred (for example, an Object reference not set diagnostic message is shown).	<p>Generation completes but any overcrowded section is not generated.</p> <p>Details are provided in the summary dialog box.</p>
Section overcrowded ⁽¹⁾	
(1) Refer to the topic describing diagnostic messages, page 440.	

In the following example, *ControlProject_2*, which consists of four sections is being generated for the first time but EcoStruxure Process Expert stops the generation process because of a variable name conflict in *FBDSection_3* (the section name is highlighted for the purpose of the example). As a result, *FBDSection_1* and *FBDSection_2* are generated in the logical Control Participant project. *FBDSection_3* is not generated because it contains the conflicting variable and *FBDSection_4* is not either because it is generated only once generation of *FBDSection_3* completes



NOTE: If you had selected the **Generate and Build** command, build would not start because at least one section did not generate.

Control Project Section Size

During generation, FBD sections are created in the Control Participant project with a maximum initial size of 360 rows x 240 columns. You can modify this size, page 405.

If additional space is required to fit the function blocks that are generated from the facets assigned to a section, EcoStruxure Process Expert adapts the section size by:

- Deleting empty columns on the outer right of the section to add rows at the bottom of the section.
- Deleting empty rows at the bottom of the section to add columns on the outer right of the section.

A message to inform you of the change in section size is displayed.

If you want to reserve some space in the section, you can insert a comment box on the right-hand side of the section and/or at the bottom of the section so that the rows that are left and/or above the comment box are not deleted when resizing the section.

When the space requirements of the function blocks exceed the maximum section size, a diagnostic message, page 440 is displayed.

NOTE: The section resizing process applies to a first generation, subsequent generations, and regenerations.

Execution Order and Function Block Layout

When you generate a Control project for the first time, EcoStruxure Process Expert processes the facets that you have assigned to the project and that have an **Assignment** status, page 379 of **Assigned**. It takes into consideration the order of sections, page 275 at the project level and the order of facets, page 379 at the section level.

Refer also to the topic giving an example of section layout management (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).

The table describes the process that applies during a first generation to manage the order of sections and facets, and the position of function blocks inside a section.

Stage	Description
1	EcoStruxure Process Expert detects the order, page 343 of the project sections and processes the sections starting with the one ranked 0; then the section ranked 1; and so on, until it has processed the sections of the entire project.
2	For each section, EcoStruxure Process Expert detects the order, page 379 of the first facet of each instance that is assigned to the section.
3	<p>EcoStruxure Process Expert generates the function blocks encapsulated in the facets according to the order that it has detected in stage 2, starting with the facet ranked 0. However, the function blocks are grouped according to the layout, page 992 of the composite referencing the facets.</p> <p>For example, if a section contains two instances, Instance1 and Instance2, referencing three and one facets respectively, and ordered as follows:</p> <p>Facet1_Instance1 - Order 0</p> <p>Facet1_Instance2 - Order 1</p> <p>Facet2_Instance1 - Order 2</p> <p>Facet3_Instance1 - Order 3</p> <p>EcoStruxure Process Expert generates Facet1_Instance1 before Facet1_Instance2 but groups function blocks of facets 1, 2 and 3 of Instance1 at the top of the section to respect the layout of the composite template used by Instance1.</p> <p>Therefore, function blocks of Facet1_Instance2 are placed at the end in spite of the order of the facet, which is 1.</p>
4	<p>If a function block is generated in a position that is already occupied by another function block, creating an overlap, the function block generated last is moved to the end of the section content.</p> <p>The block keeps its horizontal position (X axis) as defined by the layout of them template referencing it.</p> <p>In this case, a message is displayed advising you about the change in position and a possible change in the execution order.</p>

Facet Status After Unsuccessful Generation

The table describes the effect on the status of Control facets when the generation of a section or any facet is not successful.

Generated component	Facet status
Entire Control project	The generation status of facets assigned to the entire project reverts to the value it was before generating.
Control project section	The generation status of facets assigned to the section reverts to the value it was before generating.

Removing Types from the Logical Control Participant Project

Types generated from Control facets remain in the logical Control Participant project even when you unassign generated facets, page 442 encapsulating them, delete, page 425 the facets by deleting the instance, or delete the section, page 348 to which these facets were assigned.

To remove these types, you can either:

- Regenerate, page 434 the Control Participant project. This discards refinements you have made.

- Purge the types from the Control Participant during refinement. For details, refer to *Purge of Unused Instances* in the Control Participant help.

Configurable Generation Settings

By editing the GenerationSettings.xml file, you can customize the behavior of the generation process of Control Participant projects.

The file is located on the computer on which the system server is installed at the path C:\Program Files\Schneider Electric\EcoStruxure\Process Expert\System Server.

You need to have administrator rights to be able to edit this file.

NOTE: Changes to the file are applied only after a system server restart, page 80.

You can configure the following parameters.

Parameter name	Default value	Description
<i>ReGenerateNonFbdSections</i>	<i>True</i>	<p><i>True</i> = The behavior during a regeneration is the following:</p> <ul style="list-style-type: none"> • Non-FBD sections that were created during refinement and their contents are retained. • Refinements that are made in FBD sections to which facets are assigned are discarded. • Refinements made in FBD sections that were created in the Projects Explorer are retained if no facets have been assigned to these sections. <p><i>False</i> = The behavior during a regeneration is the following:</p> <ul style="list-style-type: none"> • Non-FBD sections that were created during refinement and their contents are discarded. • Refinements that are made in FBD sections are discarded whether facets have been assigned to them or not. <p>NOTE: The value of this parameter has no impact on the regeneration of data and logic of the safety program of M580 safety Control projects, page 920.</p>
<i>UnityProIgnoreFbdComments</i>	<i>True</i>	<i>True</i> = Comments of FBD sections are not generated.
<i>UnityProMinSpaceTop</i>	0	Minimum space that is left above function blocks that are positioned at the top of the section.
<i>UnityProMinSpaceLeft</i>	10	Minimum space that is left at the left of any function block that it generates.
<i>UnityProMinDistanceX</i>	5	Minimum space that is left between two consecutive function blocks along the x axis.
<i>UnityProMinDistanceY</i>	2	Minimum space that is left between two consecutive function blocks along the y axis.
<i>UnityProUseEffectiveSize</i>	<i>False</i>	<i>True</i> = The minimum space around function blocks is decreased to help utilize the space of the section more effectively.
<i>UnityProInsertToEnd</i>	<i>True</i>	<ul style="list-style-type: none"> • <i>True</i> = The function blocks that are generated after a first generation are placed at the end of sections independently of the position that they would normally occupy. • <i>False</i> = The function blocks are laid out according to the default behavior when generating Control Participant projects after changes, page 433.
<i>CheckConsistency</i>	<i>True</i>	<p>This is not a user-configurable parameter.</p> <p>Changing the value affects how EcoStruxure Process Expert operates.</p>
<i>SaveAfterNumberSectionGenerated</i>	30	<p>Number of FBD sections that need to generate successfully to trigger a save command in the Control Participant.</p> <p>If you are generating a number of sections lower than the value, a save command is executed once generation completed successfully.</p> <p>NOTE: Reducing this number may increase generation time because the number of save operations is higher. The setting does not alter the impact of detected errors during generation, page 402.</p>
<i>NumberOfUnitySectionRows</i>	36 ⁽¹⁾	<p>Number of rows/columns that the section contains when it is created by generating the Control Participant project or section for the first time or regenerating it.</p> <p><i>NumberOfUnitySectionRows</i> allowed range: 24 to 144</p>
<i>NumberOfUnitySectionColumns</i>	24 ⁽¹⁾	

Parameter name	Default value	Description
		<p><i>NumberOfUnitySectionColumns</i> allowed range: 6 to 36</p> <p>NOTE: If you modify the default section size or if the Control Participant creates a section that is different from your settings, generation or regeneration may not complete successfully because constituents cannot be positioned, page 403 as required.</p>
<p>(1) <i>NumberOfUnitySectionRows</i> x <i>NumberOfUnitySectionColumns</i> =< 864. If this value is exceeded, either 24 or 144 is retained for the number of rows (the value that comes closest to <i>NumberOfUnitySectionRows</i>) and the number of columns adjusted in consequence, independently of the setting of <i>NumberOfUnitySectionColumns</i>. For example, if you enter 200 and 7 respectively, a section of size 1440 x 60 is created because 200 exceeds the allowed number of rows and is closer to 144 than to 24.</p>		

Automatic Generation of Sections

Overview

When the automatic generation setting, page 272 is enabled, certain actions trigger the generation of sections. These are actions that have an impact on facets, which are already assigned to or that will be assigned to the corresponding sections.

The behavior in case of a detected error, page 402 is the same as when you generate manually.

Actions Starting the Automatic Generation of a Section

The following table indicates which actions trigger the generation of a section when the automatic generation of sections is enabled. If you perform several of these actions in parallel, generation may be queued until the trigger point of the last action is reached.

Action	Automatic generation trigger point
Actions on application instances that have facets already assigned to the section	
Actions that modify the identifier of an instance: <ul style="list-style-type: none">Moving the instance to another folder⁽¹⁾.Changing the hierarchy of the instance⁽¹⁾.Changing the alias of a folder in the hierarchy of the instance⁽¹⁾.Renaming the instance in-place.Disabling hierarchical naming of the instance.	The completion of the action (or saving changes when required).
Editing an instance.	Saving changes and closing the editor.
Deleting an instance.	The completion of the action.
Editing links between instances (creating, moving, deleting a link, removing an instance from the editor), which changes the status of assigned facets to <i>Out Of Date</i> . Does not apply to RTNS links, page 208.	Closing the editor.
Moving an instance from a system (source) to another system (target) ⁽²⁾ .	The completion of the action.
Updating or replacing the template of an instance.	The completion of the action.
Import operation in the Application Explorer that changes the status of assigned facets to <i>Out Of Date</i> , <i>Inconsistent</i> , or <i>Deleted</i> .	
Actions in the Control project	
Assigning facets to the section manually.	The completion of the action.
Moving facets from one section to another.	
Unassigning and relinking a facet, which changes the status of the facet to <i>Inconsistent</i> .	
Import operation leading to assigned facets having the status <i>Non Generated</i> .	
Changing the status of a facet to <i>Inconsistent</i> after refining the project offline or online and updating the project.	Closing the Control Participant refinement window or completion of the project update.
Creating an instance from the Control project section.	The completion of the facet assignment. Facets of invalid instances ⁽³⁾ are generated when the instance status becomes <i>Valid</i> .
Copying and pasting an instance in the Control project section.	The completion of the facet assignment.
<p>(1) With hierarchical naming enabled for the instance.</p> <p>(2) The section that belongs to the source system is generated.</p> <p>(3) Facets of an invalid instance can be assigned if you have created the instance in-place, page 351 or if the instance becomes invalid after its facets were assigned.</p>	

Selecting Versions of Types in Control Facets During Generation

Overview

Because in a logical Control Participant project, only one version of a given type can exist, either of the following situations require that you select which version is generated:

- You are assigning to a Control Participant project facets that encapsulate the same type but different versions of it.
- You have updated or replaced the template, page 220 that is used by an instance whose facets are already assigned to a Control Participant project and generated. The new template contains the same type (DFBs and/or DDTs) but of a different version. The assignment status of one or more of these facets has been set to **Out Of Date**.

If this case occurs, when you generate the logical Control Participant project to apply the changes, EcoStruxure Process Expert detects the presence of several versions for a same type.

As a result, the **Global Constituent Type Version Conflicts** dialog box opens where you can select either command to select the version that you want to use in the logical Control Participant project:

- **Update**
- **Skip**
- **Cancel**

EcoStruxure Process Expert maintains a list of the various versions of a type that it detects during each generation cycle and applies your initial selection when it detects a version again. If EcoStruxure Process Expert detects a new version of the same type when it generates the next section, it opens the **Global Constituent Type Version Conflicts** dialog box again.

The detection of the version change is performed at the facet level and it is irrelevant to which instance a facet belongs.

However, if two or more versions of the same type are detected in the same section, generation stops. In this case, you are informed you about the facets, the type, and the versions that create the conflict. You need to resolve the version conflict at the instance or facet level manually.

This topic contains several examples illustrating the main scenarios that you may encounter.

NOTE: Although you select the version at the section level, updating the version of a type impacts all its instances across all sections of the Control Participant project. Also, the last action that you select for a given type is the one that is applied.

NOTE: The **Global Constituent Type Version Conflicts** dialog box also opens if EcoStruxure Process Expert detects the same version of a type but with a different signature, page 412.

Relation Between Type Versions in the Application and in Control Projects

The version of the type that you choose to use in the logical Control Participant project is not necessarily the same as the version that is encapsulated in the template used by instances of the application. Indeed, in the application, instances can use several versions of a same template encapsulating different versions of a type. The version of the type that is used in the logical Control Participant project can only be one of these versions.

This example shows the different versions of a template that are used by three instances of an application. Each template encapsulating a different version of a DFB. It also shows the single type version that is generated in the Control project.

Template version and encapsulated type version in the application	Type version used in the logical Control project
Template_A [1.0.0], type version 1.2	Type version 1.3
Template_A [1.0.1], type version 1.3	
Template_A [1.0.2], type version 2.0	

This second example shows that different versions of a type can be used in different Control projects, each project using only a single version.

Template version and encapsulated type version in the application	Type version used in logical Control project 1	Type version used in logical Control project 2
Template_A [1.0.0], type version 1.2	Type version 1.3	Type version 2.0
Template_A [1.0.1], type version 1.3		
Template_A [1.0.2], type version 2.0		

Impact of Type Version Selection on Links

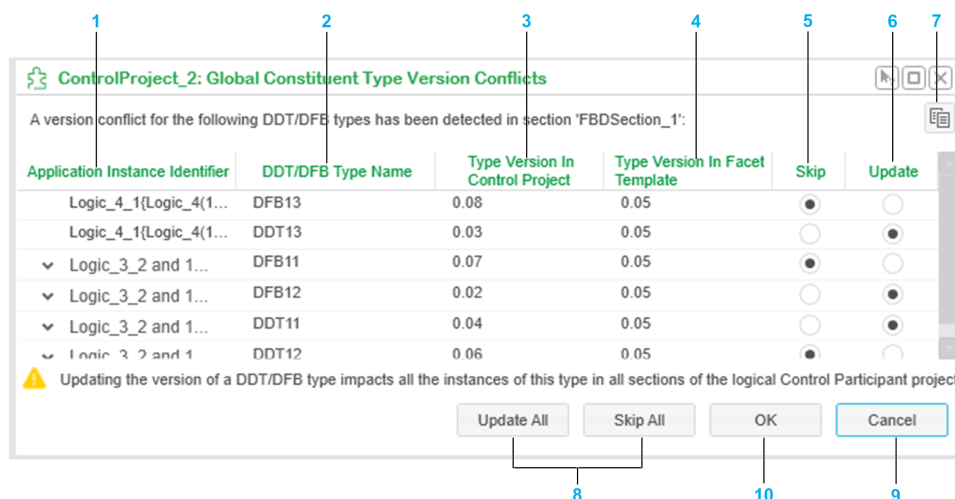
Selecting **Update** or **Skip** for a facet removes links with any instance of the encapsulated type if a pin for which a link exists is impacted in the following ways:

- The pin is removed.
- The pin is renamed.

A message is displayed informing you of the links that have been removed after generation completes.

Global Constituent Type Version Conflicts Dialog Box

The following figure shows an example scenario where several DFB and DDT types create a version conflict in *FBDSection_1*.



Item	Description
1	Identifier of an instance referencing a facet that is assigned to this section and that encapsulates the type. If other facets encapsulating the type and assigned to the section are referenced by different instances, the number of instances is indicated. Expand the row to view a list of these instances with the template identifier and version each one uses. A tooltip displays the same information when the row is not expanded.
2	Name of the type encapsulated in the facets for which a version conflict is detected.
3	Version of the type that is currently used in the logical Control Participant project.
4	Version of the type that is encapsulated in the facets being generated.

Item	Description
5	<p>Button that lets you select the Skip action for individual types.</p> <p>Does not update the type⁽¹⁾ that is currently used in the logical Control project (version X) to version Y, which is the version encapsulated in the facet. Also, it does not generate the facet.</p> <p>For each generation cycle, EcoStruxure Process Expert maintains a list of the various versions of a same type that it has detected. Based on this list, if other facets encapsulate version X or y of the same type, no dialog box opens and version X is maintained automatically.</p> <p>If, in another section, EcoStruxure Process Expert detects a facet that encapsulates the same type but of a version that is different from version X and version Y (for example, version Z), the Global Constituent Type Version Conflicts dialog box opens again. Version Z is added to the list of versions already detected.</p> <p>The generation status of facets may be set to inconsistent. In this case, when generation completes, a summary with the name of the facets that are affected and the name of the section they are assigned to is shown.</p> <p>NOTE: Skip only skips the generation of the facet that encapsulates the type. Other facets belonging to the same instance are generated if their status, page 430 requires it (for example, if it is Out Of Date).</p>
6	<p>Button that lets you select the Update action for individual types.</p> <p>Generates the facet and updates the type⁽¹⁾ currently used in the logical Control project (version X) to the version that is encapsulated in the facet (version Y). The status of the facet becomes Generated.</p> <p>For each generation cycle, EcoStruxure Process Expert maintains a list of the various versions of a same type that it has detected. Based on this list, if other facets encapsulate version X or y of the same type, no dialog box opens and version Y is used automatically.</p> <p>If, in another section, EcoStruxure Process Expert again detects a facet that encapsulates version X of the same type and the facet is not generated, it remains non-generated.</p> <p>If, in another section, EcoStruxure Process Expert detects a facet that encapsulates the same type but of a version that is different from version X and version Y (for example, version Z), the Global Constituent Type Version Conflicts dialog box opens again. This allows you to update the current version of the type (for example, version Y) to version Z. Version Z is added to the list of versions already detected.</p> <p>The generation status of facets may be set to inconsistent. In this case, when generation completes, a summary with the name of the facets that are affected and the name of the section they are assigned to is shown.</p> <p>NOTE: version Y can be a version later or earlier than version X. The same applies to version Z compared to the other versions.</p>
7	Copies the content of the dialog box to the Clipboard.
8	<p>Buttons that let you select an action for the conflicting type versions as a whole.</p> <p>The actions are the same as when you select the action for types individually.</p> <p>NOTE: Clicking these buttons overrides the selection you have made for individual types.</p>
9	<p>Generation of the Control Participant project stops. Code of sections that have been generated successfully remains in the logical Control Participant project. The type⁽¹⁾ version that was used after the generation of the previous section is maintained.</p> <p>The section in which the version conflicts have been detected and any subsequent sections are not generated.</p>
10	Generates the section by applying your selection of actions for conflicting types.
(1) Includes all the types encapsulated in the facet if more than one, for example the DFB and its DDT. You cannot select to update or skip one and not the other.	

NOTE: The status of facets is updated only after the entire generation process is completed.

Setting the Status of Facets to Inconsistent

Updating or skipping the version of a type may set the generation status of a facet to **Inconsistent**. The assignment status does not change.

The table describes under which conditions the generation status is changed.

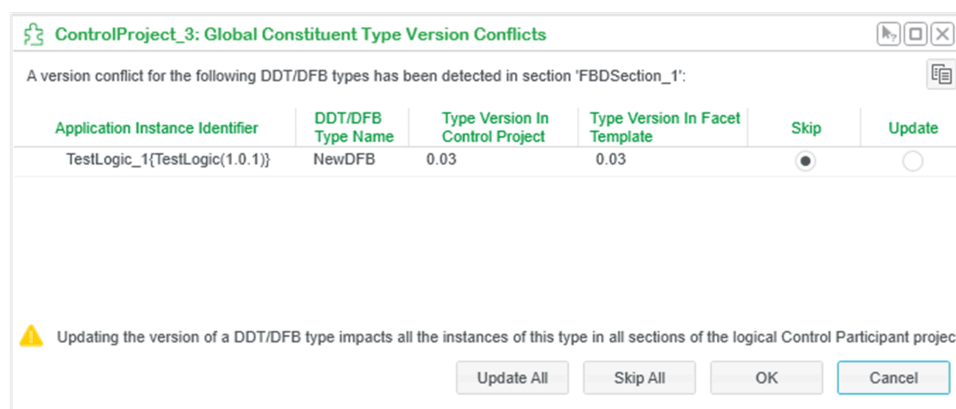
Status of the type version	Required facet status	Required user action
The version has already been detected during this generation cycle but is different from the one currently used in the Control project.	Assignment: Out Of Date	Because the version has already been detected, the Global Constituent Type Version Conflicts dialog box does not open; therefore, no action is required. Refer to example 3, page 417.
The version has not yet been detected during this generation cycle.	Assignment: Out Of Date	You are selecting Skip for the proposed version change after having already selected Update for the same type in another section during the same generation cycle. Refer to example_7, page 421.

NOTE: If the status of the facet is **Non Generated**, it does not change.

Detecting Types of Same Version But With Different Signature

The **Global Constituent Type Version Conflicts** dialog box also opens if EcoStruxure Process Expert detects in a facet assigned to the Control project and which needs to be generated, the same version of a type as it is already used in the logical Control project but with a different signature.

The following figure shows an example of detection of a type of the same version but with a different signature. A tooltip indicates the nature of the conflict when you position the pointer over the version of the type.



- **Update:** The type in the logical Control project is updated with the type and signature that is encapsulated in the facet.
- **Skip:** The type that is used in the logical Control project continues to be used; no update is performed. The generation status of facets may be set to inconsistent, page 411.

Independently of the action that you select, the **Global Constituent Type Version Conflicts** dialog box opens again if in another section of the same Control project another facet is detected, which encapsulates the same version of the same type and with a signature that is the same as in the previous facet or with yet another signature.

If two or more facets encapsulating the same version of a type but each with a different signature are detected in the same section, generation stops, page 402.

Performing a Consistency Check After Version Conflict Detection

It is a good practice to perform a consistency check, page 460 when EcoStruxure Process Expert detects several versions of the same type or the same version with different signature during generation of a Control Participant project.

A consistency check detects which generated facets encapsulate a type whose version or signature is different from the version/signature of the type used in the Control Participant project. The status of such facets is set to **Inconsistent**.

The following example illustrates a scenario where a consistency check allows identifying a facet that contains a version of a type that is different from the version used in the Control project after EcoStruxure Process Expert detected a version conflict. In this example, v1.0 of DFB_A type was generated in the Control project by Facet_2. A second facet (Facet_1) containing v2.0 of the same type is assigned to a different section and the project is generated again.

Section	Facet	DBF_A type version	Facet assignment/ generation status before generating	Facet status after generating and performing a consistency check	Comment
Section_1	Facet_1	v2.0	Out Of Date/ Generated	Assigned/ Generated	EcoStruxure Process Expert detects a version conflict and you update the type used in the Control project to v2.0 by selecting Update .
Section_2	Facet_2	v1.0	Assigned/ Generated	Assigned/ Inconsistent	Because the facet encapsulates v1.0 of the type, the consistency check has set the status of the facet to Inconsistent .

Regenerating a Control Participant Project After Type Version Changes

When you select the **Regenerate** command after generating a Control Participant project or section in which version conflicts were detected, the behavior is different depending on what you regenerate.

If you regenerate the entire Control Participant project, EcoStruxure Process Expert removes the types from the logical Control project and generates the ones that are encapsulated in facets assigned to the Control Participant project sections. If various versions of a same type are detected again, the **Global Constituent Type Version Conflicts** dialog box opens.

If you regenerate a section of a Control Participant project, EcoStruxure Process Expert maintains existing types, page 434 in the logical Control project and starts over the generation process. During regeneration, it compares:

- For the first facet of the section that encapsulates a type, the version of the type.
- The version of this type that exists in the logical Control project.

If they are different, it opens the **Global Constituent Type Version Conflicts** dialog box.

If they are identical, it performs the comparison with the second facet of the section encapsulating a type, and so on.

In case several versions of the same type are detected in the same section, page 413, generation stops.

Examples

The following examples illustrate various type version conflict scenarios when facets are assigned to the same section.

Description	Command used	Refer to
A facet is assigned to an already generated section.	–	Example 1a, page 414
A section contains two facets that are not generated.		Example 1b, page 415
A section contains two facets that are generated and one of them is updated to use a different version of the type.		Example 1c, page 416
A generated facet is updated while the same section contains a non-generated facet. This example illustrates a possible solution to examples 1.	Update and Skip	Example 2, page 416

The following examples illustrate various type version conflict scenarios when facets are assigned to different sections.

Description	Command used	Refer to
Two different versions of the same type in different sections of a Control project.	Update	Example 3, page 417
Several different versions of the same type in different sections of a Control project.		Example 4, page 418
Two different versions of the same type in different sections of a Control project.	Skip	Example 5, page 419
Several different versions of the same type in different sections of a Control project.		Example 6, page 420
Several different versions of the same type in different sections of a Control project.	Update and Skip	Example 7, page 421
Two different versions of the same type in different sections of a Control project.	Cancel and Update The Regenerate command is also described.	Example 8, page 422

Example 1a: Two Different Versions of the Same Type in the Same Section

A facet is assigned to an already generated section.

The starting point is the following:

- One facet is assigned to Section_1 and generated.
- The facet encapsulates DFB_A.

Facet ID	DFB_A version encapsulated in facet	Facet assignment/generation status	DFB_A version used in Control project
Facet_0	v1.0	Assigned/Generated	v1.0

Then, another facet encapsulating DFB_A but of a different version is assigned to the same section. The section is generated again to apply the changes.

The table explains the behavior during generation of Section_1 and which version of DFB_A is used in the end.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Facet assignment/ generation status after generation	DFB_A version used in Control project after generation	Comment
Facet_0	v1.0	Assigned/Generated	v1.0	<p>EcoStruxure Process Expert detects v2.0 while v1.0 is already generated. Because the conflict arises in the same section, generation stops, page 402 and DFB_A version v1.0 is maintained.</p> <p>The status of facets is unchanged.</p> <p>To be able to generate the section, you can either:</p> <ul style="list-style-type: none">• Unassign one of the facets from the section.• Update the template used by one of the instances so that both facets use the same version of DFB_A, page 416.• Unlink Facet_0, generate the section, select Update when the Global Constituent Type Version Conflicts dialog box opens, then relink Facet_0.
Facet_1	v2.0	Assigned/Non Generated		

(1) The order of facets in the section is not relevant.

(Back to example summary table, page 413.)

Example 1b: Two Different Versions of the Same Type in the Same Section

A section contains two facets that are not generated.

The starting point is the following:

- Two facets of two different instances are assigned to Section_1.
- Both facets encapsulate DFB_A but of a different version.
- DFB_A does not exist in the Control project.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Facet assignment/generation status
Facet_0	v1.0	Assigned/Non Generated
Facet_1	v2.0	
(1) The order of facets in the section is not relevant.		

Then, the section is generated.

The table explains the behavior during generation of Section_1 and which version of DFB_A is used in the end.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Facet assignment/ generation status after generation	DFB_A version used in Control project after generation	Comment
Facet_0	v1.0	Assigned/Non Generated	None	<p>EcoStruxure Process Expert detects v1.0 and v2.0 in the same section. Because of this, generation stops, page 402 and no version of DFB_A is generated.</p> <p>The status of facets is unchanged.</p> <p>To be able to generate the section, you can either:</p> <ul style="list-style-type: none">• Unassign one of the facets from the section.• Update the template used by one of the instances so that both facets use the same version of DFB_A, page 416.
Facet_1	v2.0			

(1) The order of facets in the section is not relevant.

(Back to example summary table, page 413.)

Example 1c: Two Different Versions of the Same Type in the Same Section

A section contains two facets that are generated and one of them is updated to use a different version of the type.

The starting point is the following:

- Two facets of two different instances are assigned to Section_1.
- Both facets encapsulate the same version of DFB_A.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Facet assignment/ generation status	DFB_A version used in Control project
Facet_0	v1.0	Assigned/Generated	v1.0
Facet_1	v1.0		
(1) The order of facets in the section is not relevant.			

Then, the template used by instance referencing Facet_0 is updated. This results in Facet_0 encapsulating a different version of DFB_A. The section is generated again to apply the change.

The table explains the behavior during generation of Section_1 and which version of DFB_A is used in the end.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Facet assignment/ generation status after generation	DFB_A version used in Control project after generation	Comment
Facet_0	v2.0	Out Of Date/Generated	v1.0	<p>EcoStruxure Process Expert detects v2.0 while v1.0 is already generated. Because the conflict arises in the same section, generation stops, page 402 and DFB_A version v1.0 is maintained.</p> <p>The status of facets is unchanged.</p> <p>To be able to generate the section, you can either:</p> <ul style="list-style-type: none">• Unassign one of the facets from the section.• Update the template used by the instance referencing Facet_1 so that both facets use the same version of DFB_A, page 416.• Unlink the already assigned and generated facet (Facet_1), generate the section, select Update when the Global Constituent Type Version Conflicts dialog box opens, then relink Facet_1.
Facet_1	v1.0	Assigned/Generated		

(1) The order of facets in the section is not relevant.

(Back to example summary table, page 413.)

Example 2: Two Different Versions of the Same Type in the Same Section

A generated facet is updated while the same section contains a non-generated facet.

The starting point is the following:

- Two facets are assigned to Section_1.
- One facet is generated, the other not.
- Both facets encapsulate DFB_A but of a different version.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Facet assignment/generation status	DFB_A version used in Control project
Facet_0	v1.0	Assigned/Generated	v1.0
Facet_1	v2.0	Assigned/Non Generated	
(1) The order of facets in the section is not relevant.			

Then, the template used by instance referencing Facet_0 is updated. This results in Facet_0 encapsulating the same version of DFB_A as Facet_1 (v2.0). The assignment status of Facet_0 is set to **Out Of Date**. The section is generated again to apply the change.

The table explains the behavior during generation of Section_1 and which version of DFB_A is used in the end.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Facet assignment/ generation status after generation	DFB_A version used in Control project after generation	Comment
Facet_0	v2.0	Assigned/Generated	v2.0	EcoStruxure Process Expert detects v2.0 and opens the Global Constituent Type Version Conflicts dialog box. Update is selected. The facet is generated and the version of DFB_A used in the logical Control project is updated from v1.0 to v2.0.
Facet_1	v2.0	Assigned/Generated		
(1) The order of facets in the section is not relevant.				

NOTE: If you select **Skip** instead, none of the facets in Section_1 are generated and the version of DFB_A in the logical Control project remains v1.0.

(Back to example summary table, page 413.)

Example 3: Two Different Versions of the Same Type in Different Sections of a Control Project

This example illustrates the behavior during generation when facets encapsulating two different versions of the same type are assigned to various sections.

The starting point is the following:

- Four facets of four different instances are assigned to separate sections.
- Each facet encapsulates DFB_A.
- All sections are part of the same Control project and generated.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project
Facet_0	v1.0	Section_0	v1.0
Facet_1	v1.0	Section_1	
Facet_2	v1.0	Section_2	
Facet_3	v1.0	Section_3	
No facet assigned	—	Section_4	
(1) The order of the sections corresponds to the section number.			

Then, the following changes are performed:

- The template used by one instance is updated. As a result, one facet encapsulates DFB_A of a different version and its status is set to **Out Of Date**.
- A new facet, Facet_4, is assigned to Section_4. It encapsulates version v1.0 of DFB_A and its statuses are **Assigned/Non Generated**.

The Control project is generated again to apply the changes.

The table explains the behavior during generation of the Control project and which version of DFB_A is used in the end.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_0	v1.0	Section_0	v1.0	No change to be generated for the facet.
Facet_1	v2.0	Section_1	v2.0 ⁽²⁾	EcoStruxure Process Expert detects v2.0 and opens the Global Constituent Type Version Conflicts dialog box. Update is selected. The facet is generated and the version of DFB_A used in the logical Control project is updated from v1.0 to v2.0.
Facet_2	v1.0	Section_2		Because v1.0 has already been detected, the previously selected Update command is applied automatically and DFB_A version v2.0 is maintained.
Facet_3	v1.0	Section_3		
Facet_4	v1.0	Section_4		Facet_4 is not generated; its statuses remain Assigned/ Non Generated . NOTE: If the assignment status of Facet_2 or Facet_3 is Out Of Date , its generation status is set to Inconsistent .
(1) The order of the sections corresponds to the section number.				
(2) Version of DFB_A that is used in the end after successful generation of the Control project.				

(Back to example summary table, page 413.)

Example 4: Several Different Versions of the Same Type in Different Sections of a Control Project

This example illustrates the behavior during generation when facets encapsulating different versions of the same type are assigned to the various sections.

The starting point is the following:

- Four facets of four different instances are assigned to separate sections.
- Each facet encapsulates DFB_A.
- All sections are part of the same Control project and generated.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project
Facet_0	v1.0	Section_0	v1.0
Facet_1	v1.0	Section_1	
Facet_2	v1.0	Section_2	
Facet_3	v1.0	Section_3	
(1) The order of the sections corresponds to the section number.			

Then, the template used by some instances is updated with various versions. As a result, the facets encapsulate various versions of DFB_A and their status is set to **Out Of Date**. The Control project is generated again to apply the changes.

The table explains the behavior during generation of the Control project and which version of DFB_A is used in the end.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_0	v1.0	Section_0	v1.0	No change to be generated for the facet.
Facet_1	v3.0	Section_1	v3.0	EcoStruxure Process Expert detects v3.0 and opens the Global Constituent Type Version Conflicts dialog box. Update is selected. The facet is generated and the version of DFB_A used in the logical Control project is updated from v1.0 to v3.0.
Facet_2	v1.0	Section_2		Because v1.0 has already been detected, the previously selected Update command is applied automatically and DFB_A version v3.0 is maintained. NOTE: If the assignment status of the facet is Out Of Date , its generation status is set to Inconsistent .
Facet_3	v4.0	Section_3	v4.0 ⁽²⁾	EcoStruxure Process Expert detects v4.0 and opens the Global Constituent Type Version Conflicts dialog box. Update is selected. The facet is generated and the version of DFB_A used in the logical Control project is updated from v3.0 to v4.0.

(1) The order of the sections corresponds to the section number.

(2) Version of DFB_A that is used in the end after successful generation of the Control project.

NOTE: It is possible to update the version of a type to an earlier one. For example, if Facet_3 would encapsulate DFB_A of version v2.0, the version used in the end would be v2.0.

(Back to example summary table, page 413.)

Example 5: Two Different Versions of the Same Type in Different Sections of a Control Project

This example illustrates the behavior during generation when facets encapsulating two different versions of the same type are assigned to various sections and the **Skip** command is used.

The starting point is the following:

- Four facets of four different instances are assigned to separate sections.
- Each facet encapsulates DFB_A.
- All sections are part of the same Control project and generated.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project
Facet_0	v1.0	Section_0	v1.0
Facet_1	v1.0	Section_1	
Facet_2	v1.0	Section_2	
Facet_3	v1.0	Section_3	

(1) The order of the sections corresponds to the section number.

Then, the template used by two instances is updated. As a result, two facets encapsulate DFB_A of a different version and their status is set to **Out Of Date**. The Control project is generated again to apply the changes.

The table explains the behavior during generation of the Control project and which version of DFB_A is used in the end.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_0	v1.0	Section_0	v1.0 ⁽²⁾	No change to be generated for the facet.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_1	v2.0	Section_1		EcoStruxure Process Expert detects v2.0 and opens the Global Constituent Type Version Conflicts dialog box. Skip is selected. The facet is not generated and the version of DFB_A used in the logical Control project remains v1.0.
Facet_2	v1.0	Section_2		No change to be generated for the facet.
Facet_3	v2.0	Section_3		Because v2.0 has already been detected, the previously selected Skip command is applied automatically. The facet is not generated and DFB_A version v1.0 is maintained.
(1) The order of the sections corresponds to the section number.				
(2) Version of DFB_A that is used in the end after successful generation of the Control project.				

NOTE: The assignment status of Facet_1 and Facet_3 remains **Out Of Date**.

(Back to example summary table, page 413.)

Example 6: Several Different Versions of the Same Type in Different Sections of a Control Project

This example illustrates the behavior during generation when facets encapsulating different versions of the same type are assigned to the various sections and the **Skip** command is used.

The starting point is the following:

- Four facets of four different instances are assigned to separate sections.
- Each facet encapsulates DFB_A.
- All sections are part of the same Control project and generated.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project
Facet_0	v1.0	Section_0	v1.0
Facet_1	v1.0	Section_1	
Facet_2	v1.0	Section_2	
Facet_3	v1.0	Section_3	
(1) The order of the sections corresponds to the section number.			

Then, the template used by some instances is updated with various versions. As a result, the facets encapsulate various versions of DFB_A and their status is set to **Out Of Date**. The Control project is generated again to apply the changes.

The table explains the behavior during generation of the Control project and which version of DFB_A is used in the end.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_0	v1.0	Section_0	v1.0 ⁽²⁾	No change to be generated for the facet.
Facet_1	v2.0	Section_1		EcoStruxure Process Expert detects v2.0 and opens the Global Constituent Type Version Conflicts dialog box. Skip is selected. The facet is not generated and the version of DFB_A used in the logical Control project remains v1.0.
Facet_2	v4.0	Section_2		EcoStruxure Process Expert detects v4.0 and opens the Global Constituent Type Version Conflicts dialog box. Skip is selected. The facet is not generated and the version of DFB_A used in the logical Control project remains v1.0.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_3	v2.0	Section_3		Because version v2.0 has already been detected (for Facet_1), the previously selected Skip command is applied automatically. The facet is not generated and the version of DFB_A used in the logical Control project remains v1.0.
(1) The order of the sections corresponds to the section number.				
(2) Version of DFB_A that is used in the end after successful generation of the Control project.				

NOTE: The assignment status of Facet_1, Facet_2, and Facet_3 remains **Out Of Date**.

(Back to example summary table, page 413.)

Example 7: Several Different Versions of the Same Type in Different Sections of a Control Project

This example illustrates the behavior during generation when facets encapsulating different versions of the same type are assigned to the various sections and the **Update** and **Skip** commands are used.

The starting point is the following:

- Six facets of six different instances are assigned to separate sections.
- Each facet encapsulates DFB_A.
- All sections are part of the same Control project and generated.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project
Facet_0	v1.0	Section_0	v1.0
Facet_1	v1.0	Section_1	
Facet_2	v1.0	Section_2	
Facet_3	v1.0	Section_3	
Facet_4	v1.0	Section_4	
Facet_5	v1.0	Section_5	
(1) The order of the sections corresponds to the section number.			

Then, the template used by the instances is updated with various versions. As a result, the facets encapsulate various versions of DFB_A and their status is set to **Out Of Date**. The Control project is generated again to apply the changes.

The table explains the behavior during generation of the Control project and which version of DFB_A is used in the end.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_0	v4.0	Section_0	v4.0	EcoStruxure Process Expert detects v4.0 and opens the Global Constituent Type Version Conflicts dialog box. Update is selected. The facet is generated and the version of DFB_A used in the logical Control project is updated from v1.0 to v4.0.
Facet_1	v2.0	Section_1		EcoStruxure Process Expert detects v2.0 and opens the Global Constituent Type Version Conflicts dialog box. Skip is selected. The facet is not generated and version v4.0 of DFB_A is maintained. NOTE: The generation status of the facet is set to Inconsistent because the version of DFB_A has been previously updated when generating Facet_0.

Facet ID	DFB_A version encapsulated in facet	Section ID ⁽¹⁾	DFB_A version used in Control project after facet generation	Comment
Facet_2	v1.0	Section_2		Because version v1.0 has already been detected when generating Facet_0, the previously selected Update command is applied automatically and DFB_A version v4.0 is maintained. NOTE: If the assignment status of the facet is Out Of Date , its generation status is set to Inconsistent .
Facet_3	v4.0	Section_3		The facet is generated but without the detection of a version change because the version of DFB_A used in the Control project is already v4.0 due to the update when generating Facet_0.
Facet_4	v2.0	Section_4	v4.0 ⁽²⁾	The previously selected Skip command (for Facet_1) is applied automatically. The facet is not generated and DFB_A version v4.0 is maintained. NOTE: The generation status of the facet is set to Inconsistent because the version of DFB_A has been previously updated when generating Facet_0.
Facet_5	v3.0	Section_5		EcoStruxure Process Expert detects v3.0 for the first time and therefore opens the Global Constituent Type Version Conflicts dialog box. Skip is selected. The facet is not generated and the version of DFB_A used in the logical Control project remains v4.0. NOTE: The generation status of the facet is set to Inconsistent because the version of DFB_A has been previously updated when generating Facet_0.
(1) The order of the sections corresponds to the section number.				
(2) Version of DFB_A that is used in the end after successful generation of the Control project.				

(Back to example summary table, page 413.)

Example 8: Two Different Versions of the Same Type in Different Sections

This example illustrates the behavior during generation when facets encapsulating different versions of the same type are assigned to the different sections and the **Cancel** and **Update** commands are used.

The starting point is the following:

- Two facets of two different instances are assigned to one section each.
- Each facet encapsulates DFB_A.
- Facets are not yet generated.

Facet ID ⁽¹⁾	DFB_A version encapsulated in facet	Section ID	DFB_A version used in Control project
Facet_0	v1.0	Section_1	None
Facet_1	v2.0	Section_2	
(1) The order of the facet in the section corresponds to the facet number.			

Then, the Control project is generated.

The table explains the behavior during generation of the Control project and which version of DFB_A is used in the end.

Facet ID	DFB_A version encapsulated in facet	Section ID	DFB_A version used in Control project after generation	Comment
Facet_0	v1.0	Section_1	1.0	Version v1.0 of DFB_A is generated in the logical Control project.
Facet_1	v2.0	Section_2	1.0 ⁽¹⁾	EcoStruxure Process Expert detects v2.0 and opens the Global Constituent Type Version Conflicts dialog box. Cancel is selected. Generation stops. The facet is not generated and the version of DFB_A used in the logical Control project remains v1.0.
(1) Version of DFB_A that is used in the end after successful generation of the section.				

Then, the Control project is generated a second time.

Facet ID	DFB_A version encapsulated in facet	Section ID	DFB_A version used in Control project after generation	Comment
Facet_0	v1.0	Section_1	1.0	No change to be generated for the facet.
Facet_1	v2.0	Section_2	2.0 ⁽¹⁾	EcoStruxure Process Expert detects v2.0 and opens the Global Constituent Type Version Conflicts dialog box. Update is selected. The facet is generated and the version of DFB_A used in the logical Control project is updated from v1.0 to v2.0.
(1) Version of DFB_A that is used in the end after successful generation of the section.				

NOTE: After the second generation, if you regenerate:

- The entire Control project: A version conflict is detected in Section_2 like during a first generation because type DFB_A is removed from the Control before regenerating.
- Section_1: A version conflict is detected because the version of DFB_A in the Control project (v2.0) is different from the version encapsulated in Facet_0 (v1.0).
- Section_2: No version conflict is detected.

(Back to example summary table, page 413.)

Statutes of Facets

Overview

After the generation process, the generation and assignment statuses of facets provide information on the facet state.

NOTE: For information on the behavior when generating such facets again, refer to the topic describing the generation of facets after changes, page 430.

Generation Status of Facets

The generation status indicates the status of the facet in relation to the generation of its constituents in the logical Participant project.

You can see the generation status in the **Generation** column of the **Assignments** pane, page 343 in the **Assignment Editor**.

The table describes the possible generation statuses of facets following a generation.

Status	Description
Non Generated	<p>You have assigned the facet to the FBD section or container after generating that section/container. Generate, page 430 the section/container again to create the constituents of the facet in the logical Participant project.</p> <p>It can also indicate that the generation process of a facet or its container was unsuccessful.</p>
Generated	<p>The constituents of the facet are successfully generated into the logical participant project.</p> <p>NOTE: The status of Supervision genie facets that you assign to a page during page edit is automatically set to Generated.</p>
Moved	<p>You have moved, page 391 the facet to another FBD section of the Control project or another container of the Supervision.</p> <p>NOTE:</p> <ul style="list-style-type: none"> If you move the facet back to its original FBD section or container, it keeps the status Moved. The status does not apply to genie facets.
Inconsistent	<p>You have refined, updated, or imported the logical Control Participant project, or relinked a facet, which was unlinked and an inconsistency in the facet was detected during the consistency check, page 460:</p> <ul style="list-style-type: none"> You have not yet generated the section containing the facet again in order to resolve the inconsistency, page 432 and change the status, or EcoStruxure Process Expert could not resolve the inconsistency during a subsequent generation and the status of the facet remains Inconsistent. <p>NOTE: By default, the status of Supervision genie and data facets changes to Inconsistent when you reassign or relink, page 442 them.</p> <p>NOTE: This status can also be the result of a type version conflict that was detected during generation of Control facets, page 409.</p>
Unlinked	<p>You have unlinked the facet from its already generated constituents in the logical Participant project:</p> <ul style="list-style-type: none"> By using the Unlink, page 443 command. By unlinking the facet through the Check Consistency dialog box: <ul style="list-style-type: none"> After refining a Control Participant project, page 446. Before updating a Control Participant project, page 903. <p>By unlinking a facet, EcoStruxure Process Expert does not change already generated constituents when you generate it again if the status the facet is such that a generation would change its status (for example, if the status of the facet is out of date or inconsistent, generating it would change its status to generated). This allows you to preserve changes that you have made in the logical Participant project to constituents of the facet through refinement or refinement online.</p>

Assignment Status of Generated Facets

The assignment status indicates the status of the facet in relation to the instance that references the facet.

EcoStruxure Process Expert may update the **Assignment** status of the facet when you change the properties of the instance that references the facet.

You can see the assignment status in the **Assignment** column of the **Assignments** pane, page 343 in the **Assignment Editor**.

The table describes the possible assignment statuses of a facet following a generation.

Status	Description
Assigned	The facet is assigned to the container.
Unassigned	You have unassigned, page 442 a generated facet from the container to which you had assigned it. NOTE: The facet can still be assigned to other projects of the system.
Out Of Date	<p>You have changed at least 1 of the following parameter configurations of the instance that references the facet:</p> <ul style="list-style-type: none"> • <i>\$System</i> parameters (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) that affect the facet. • Element selections and their parameters. • Instance identifier (including changes made by EcoStruxure Process Expert related to the hierarchical naming function). • Links. • Updated or replaced the template that is used by the instance. • Other changes made within the application to an instance and that affect its facet. <p>Furthermore, for Supervision data facets only, you have changed any of the following in the Supervision project:</p> <ul style="list-style-type: none"> • The identifier of the cluster or I/O device. • The identifier of the tag container if Protocol is set to <i>OPCUA</i>. • The relationship between the tag container and the I/O device. • The protocol of the tag container. • The application structure, page 213 (in the Application Explorer).
Deleted	You have unselected the corresponding element in the Instance Editor or deleted the instance that references the facet.

Generating a Project for the First Time

Overview

This section describes how to use the **Project Explorer** to create a logical Control or Supervision Participant project or part of a project for the first time.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Generating Entire Projects

Overview

The generation of the entire project for the first time encompasses:

- For a Control project: The FBD sections of the project and the facets that are assigned to these sections.
- For a Supervision project: The various containers of the clusters of the project to which facets are assigned and any associated components of the **Services** node.

The status of facets is updated only after the entire generation process is completed.

NOTE: You can generate a project while the generation of another project is still ongoing.

Aborting Project Generation Tasks

After selecting the **Generate** command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The project is not generated; what was already generated at the time you canceled the task is reverted.

Refer to the notification panel for details.

Only the user who selected the command is allowed to cancel it.

Generating an Entire Project

To generate an entire Participant project, in the project browser or the **Containers** pane of the **Assignment Editor**, right-click the project that you want to generate and select **Generate**.

Result:

- The corresponding logical Participant project is created. Generation is complete when **Completed** is displayed for the process in the notification panel.
- The status of the **Generation** column in the **Assignments** pane changes to **Generated** for facets that are assigned to the project.

NOTE: If you are generating the project again, the generation status, page 424 of facets that were **Unlinked** remains unchanged and those that were **Inconsistent** may remain unchanged.

NOTE: For Control projects, you can configure certain generation settings, page 405.

Generating Sections, Containers, and Nodes

Overview

The generation of parts of a project for the first time encompasses:

- For an FBD section of a Control project: The facets that are assigned to the section and the order of the section.
- For a Supervision project:
 - For the **Containers** node: The containers of all the clusters of the project.
 - For tag containers, pages, client and sever event containers: The facets that are assigned to the respective container.
 - For the **Tags** containers: The facets that are assigned to the containers of the corresponding cluster.
 - For a cluster: The containers of the **Tags**, **ClientEvents**, and **ServerEvents** nodes of the cluster and any components of the **Services** node.
 - For the **Services** node: The components of the node.

NOTE: You can generate several sections or containers of the same project at the same time but the generation process will occur in sequence. If the sections/containers belong to different projects, they are generated in parallel.

Aborting Generation Tasks

After selecting the **Generate** command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The container of the Participant project is not generated; what was already generated at the time you canceled the task is reverted.

Refer to the notification panel for details.

Only the user who selected the command is allowed to cancel it.

Generating Part of a Project

To generate part of a project, in the project browser or in the **Containers** pane of the **Assignment Editor**, right-click the item that you want to generate and select **Generate**.

Result:

- The contents of the selected item is generated. Generation is complete when **Completed** is displayed for the process in the notification panel.
- The status of the **Generation** column in the **Assignments** pane changes to **Generated** for the facets that are assigned to the selected containers.

NOTE: If you are generating this part of the project again, the generation status, page 424 of facets that were **Unlinked** remains unchanged and those that were **Inconsistent** may remain unchanged.

NOTE: For Control projects, you can configure certain generation settings, page 405.

Generating Projects After Changes

Overview

This section describes how to use the **Project Explorer** to perform a subsequent generation or to regenerate entire logical Control or Supervision Participant projects, or parts of such projects.

You may be able to import into the deployed Control Participant project, page 885 sections that you have generated or regenerated after modifying them.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Generating Participant Projects After Changes

Overview

You need to generate the logical Participant project that you have created with the first-time generation when you make changes that result in facets having one of the following statuses:

- **Assignment status:**
 - **Unassigned**
 - **Out Of Date**
 - **Deleted**
- **Generation status:**
 - **Non Generated**
 - **Moved**
 - **Inconsistent**

You also need to generate it when you perform the following actions on sections of a Control project in the **Project Explorer**:

- Rename a section.
- Create an FBD section.
- Change the order of sections.

To update the logical Participant project, use one of the following commands:

Generate	Only changes that you made to the application and the project since the last generation are generated incrementally into the logical Participant project. Already generated data remains unchanged. However, certain changes that you made during refinement may be discarded.
Regenerate	Applies to <i>Control projects only</i> , page 434. Re-creates the logical Participant project from the beginning based on the application and Control project settings. Certain changes that you made during refinement are discarded. You can configure certain generation settings, page 405. EcoStruxure Process Expert validates data, page 398 like it does when you generate.

NOTE: To apply to the logical Control project changes that you have made to sections, you can also select the **Refine** command, page 446 at the project level.

Generation Function Description

The table indicates the actions performed on facets when you perform a subsequent generation of a project or parts of it, based on the **Assignment** and **Generation** status of the facet.

	Facet generation status				
Facet assignment status	<i>Non Generated</i>	<i>Generated</i>	<i>Moved</i>	<i>Inconsistent</i>	<i>Unlinked</i>
<i>Assigned</i>	X	O ⁽¹⁾	X	X	O
<i>Out Of Date</i>	–	X	X	X	O
<i>Unassigned</i>	–	Del	–	Del	–

	Facet generation status				
<i>Deleted</i>	–	Del	–	Del	–
<p>X Creates the constituent of the facet in the logical Participant project</p> <p>O No action</p> <p>Del Removes the facet and its already generated constituents (for example, variables and FB instances) but not derived types (DFBs and DDTs)</p> <p>– Combination not applicable</p> <p>(1) Some exceptions apply. Refer to the topic describing the generation of Control projects after refining</p>					

NOTE: When you change the order of a facet inside a section that you had generated a first time and generate it again, EcoStruxure Process Expert does not propagate the new order of facets, page 394 to the logical Control Participant project.

Generating Control Projects After Changing the Order of Sections

When you change the order, page 275 of already generated sections and generate again either the entire Control project or a section whose order you have changed, EcoStruxure Process Expert propagates the new order to the logical Control Participant project.

NOTE: Generating at the project level may take slightly longer to complete compared to generating at the section level.

Generating Control Projects After Refining

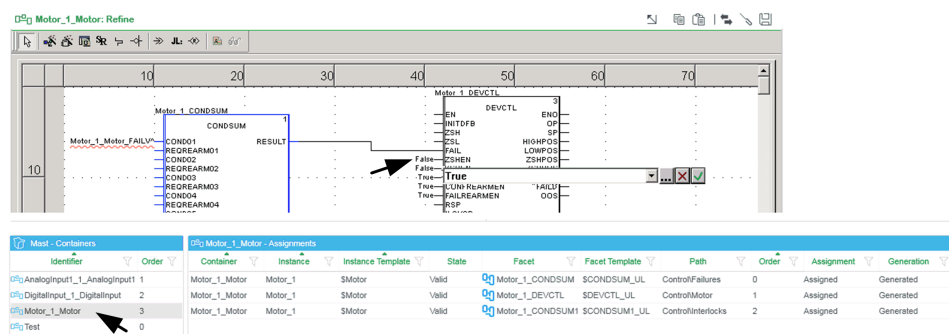
If during refinement you have modified code in a section that was already generated from a facet (for example, you have modified the variable or immediate value on a pin of a DFB), EcoStruxure Process Expert reverts your change when you generate the section again if any of the following conditions is true; otherwise your modification is preserved:

- You assign a facet to this section.
- You unassign a facet in this section.
- You move a facet to or from this section.
- You make a modification, which sets the status of any facet assigned to this section to **Out Of Date**.
- You perform a consistency check, which sets the status of any facet assigned to this section to **Inconsistent**.

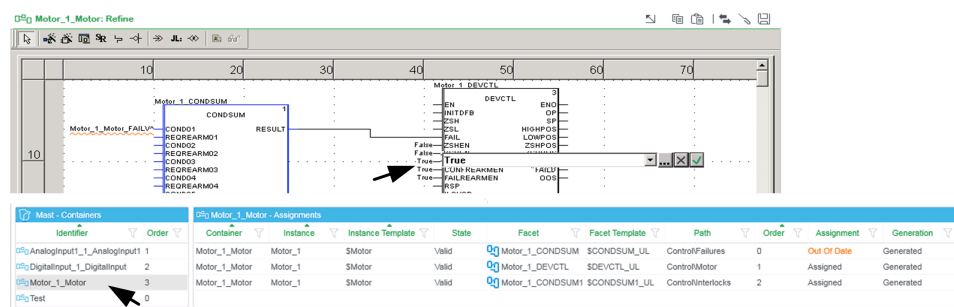
Your modification is reverted to the code that was previously generated from the facet.

If your modification consists in replacing a variable or immediate value on the pin of an FFB by a graphical link, such modification is not reverted by a subsequent generation. If the change is detected as an inconsistency, the generation status of the corresponding facet is set to **Inconsistent**.

In this example, the immediate value of the *ZSHEN* input parameter of the *DEVCTL* DFB is changed to *True* during refinement and saved. Section *Motor_1_Motor* that contains the facet, which generates the DFB (*Motor_1_DEVCTL*) contains only facets with the **Assigned** and **Generated** status. No other changes were made in the Control project. Generating section *Motor_1_Motor* does not undo your refinement; the immediate value of the *ZSHEN* input parameter remains *True*.



However, if section *Motor_1_Motor* contains, for example, a facet with status **Out Of Date**, generating the section undoes the refinement change, setting the *ZSHEN* input parameter again to *False*.



Generating Supervision Projects After Refining

Generating a Supervision project overrides any changes that you have made in refinement to elements pertaining to the **Services**, page 306 of the project.

Generating Inconsistent Facets

During generation, EcoStruxure Process Expert attempts to restore the generated constituents of facets in the logical Control Participant project that have the **Generation** status **Inconsistent** to be consistent with the related facets of the Control project.

The resulting **Generation** status of an inconsistent facet can be:

- **Generated:** EcoStruxure Process Expert completely restored the generated constituents while preserving the refinement that you have made.
- **Inconsistent:** EcoStruxure Process Expert was unable to restore the generated constituents completely because it would require altering the refinement that you have made. EcoStruxure Process Expert may have partially restored the constituents and the refinement that you have made was preserved.

This is the case, for example, if during refinement of an FBD section, you replace a variable by a graphical link and such a change is detected as an inconsistency. Your refinement is retained if you generate the section and it is discarded if you regenerate the section.

Generating Facets with State Invalid

If the **State** of a facet, page 379 has become **Invalid** after you generated it, generating the container or Participant project again does not remove the facet from the container.

This can be the case, for example, when you link instances A and B (where instance A needs to be linked to be valid), assign their facets to a Control project, generate the section, and then delete the link between the two instances. The facet of instance A becomes invalid because the instance is not linked anymore.

You need to either [regenerate](#), page 434 the section to which the invalid facet is assigned or unassign the facet and generate to remove it from the section.

Function Block Layout

When you generate Control projects or sections again, function blocks that are encapsulated in facets are positioned depending on the status of the facet:

- Facets that you had already generated and that have the following status:
 - **Out Of Date:** Facets creating a new function block after a change in their properties. The function block is positioned according to the layout defined by the template referencing the facet.
 - **Inconsistent:** Function blocks encapsulated in the facet are recreated and positioned in their previously generated position.

For both facets, if the position of the function block is already occupied, the block is moved to the end of the section content (Y axis), keeping its horizontal position (X axis) as defined by the layout of the template referencing the facet.

- **Non Generated** facets:
 - Single facets created through an element selection and that you have assigned to the section after generating it: Function blocks encapsulated in the facet are positioned at the end of the section content, on the outer left.
 - Facets referenced by an instance of which no facets were previously assigned: Function blocks encapsulated in the facets are positioned at the end of the section content while respecting the layout as defined by the template that the instance uses.

This requires that the generation of the corresponding facets completes successfully and that the resulting facet status is **Generated**.

The section size, page 403 is managed automatically.

For more information, refer to Section Layout Management Example (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

NOTE: To keep the constituents that are generated for an instance grouped, assign all its facets at once.

NOTE: Certain [configurable generation settings](#), page 405 can affect the position of function blocks.

Generating a Project Again

Refer to the procedure describing how to generate:

- The entire project, page 427
- Part of a project, page 428

Regenerating a Control Project

Control Project Regeneration Description

This process recreates the logical Control Participant project from the beginning based on these settings:

- Section execution order.
- Facet execution order inside the section.
- Properties of assigned facets.

The following changes made during refinement, page 446 are discarded by the regeneration process:

- Changes to the contents of sections to which facets are assigned.
- Contents added to such sections.
- If you set the *ReGenerateNonFbdSections* parameter to *false*:
 - Non-FBD sections that you added during refinement and their contents.
 - Name changes of FBD sections that are created by the generation process.
 - FBD sections that you added during refinement, their contents, and facets assigned to these sections.
- **Firmware** and **Web Diagnostic/Data Storage** passwords, page 447 set in the project **Properties**.

You can regenerate the entire Control project or FBD sections individually.

If EcoStruxure Process Expert could not regenerate a Control project or FBD sections of it, the status of facets, page 436 may vary.

Types Removed When Regenerating

The following derived types (DFBs and DDTs) are removed from the logical Control project when you regenerate it:

- Types that were generated from facets that you have unassigned from a project section or whose instance has been deleted.
- Types that remained after deleting the section that contained the facet that generated them.

NOTE: In both cases, such types are not removed when you:

- Regenerate only a section of a Control project.
- Generate the Control project.

Actions on Facets

The table indicates the actions performed on facets when you regenerate a Control project or FBD section, based on the **Assignment** and **Generation** status of the facet.

	Facet generation status				
Facet assignment status	<i>Non Generated</i>	<i>Generated</i>	<i>Moved</i>	<i>Inconsistent</i>	<i>Unlinked</i>
<i>Assigned</i>	X	X	X	X	X
<i>Out Of Date</i>	–	X	X	X	X
<i>Unassigned</i>	–	Del	–	Del	–
<i>Deleted</i>	–	Del	–	Del	–
<p>X Creates the constituent of the facet in the Participant project</p> <p>Del Deletes the facet and its already generated constituents</p> <p>– Combination not applicable</p>					

Control Project Section Size

EcoStruxure Process Expert manages the section size, page 403.

Execution Order and Function Block Layout

The process to regenerate the section order and section layout is the same as for a first-time generation, page 403, independently of the facet status.

Regenerating Control Projects Containing Non-FBD Sections

When you regenerate a Control project that contains non-FBD sections, a message appears in the notification panel for each one of them informing you that the section was generated because of an order change. This message, which is related to an internal process, appears even if the position of these sections has not changed.

An exception applies to non-FBD sections that have rank 0 (**Order** column) when the **Regenerate** command is used. For these sections, the message does not appear whether their position has changed or not.

Aborting Regeneration of Projects And Sections

After confirming the **ReGenerate** command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The Control project or section is not regenerated; what was already regenerated at the moment you canceled the task is reverted.

Refer to the notification panel for details.

Only the user who selected the command is allowed to cancel it.

NOTE: If you are regenerating a small project or section, the task may complete before you are able to click the icon.

Regenerating the Entire Control Project

To recreate the entire logical Control Participant project, proceed as follows.

Step	Action
1	Configure generation settings, page 405 as needed.
2	<p>In the project browser or the Containers pane of the Assignment Editor, right-click the Control project that you want to regenerate and select ReGenerate.</p> <p>Result:</p> <ul style="list-style-type: none"> The the logical Control Participant project is recreated. Regeneration is complete when Completed is displayed for the process in the notification panel. The status of the Generation column in the Assignments pane changes to Generated for the facets that are assigned to the project.

Regenerating an FBD Section Only

The regeneration at the FBD section level allows you to recreate the Control Participant logic for the facets contained in the selected FBD section only.

To regenerate the logic of an FBD section, in the **Containers** pane of the **Assignment Editor**, right-click the section that you want to regenerate and select **ReGenerate**.

Result:

- Only the Control Participant logic of the selected section is recreated. Regeneration is complete when **Completed** is displayed for the process in the notification panel.
- The status of the **Generation** column in the **Assignments** pane changes to **Generated** for the facets that are assigned to the selected FBD section.

Facet Status After Unsuccessful Regeneration

The table describes the effect on the status of Control facets when the regeneration of a section is not successful.

Regenerated project component	Facet status
Entire Control project	The generation status of facets assigned to sections that are not regenerated, page 402 reverts to the value it was before regenerating.
Control project section	The generation status of facets assigned to this section reverts to the value it was before regenerating.

Generating Equipment in Supervision Projects

Generating Equipment Based on the Application

Overview

By default, generating a Supervision project populates the equipment table of the Supervision project based on the structure of the application, page 159. Each instance of the application creates one equipment instance. This allows you to view alarms during runtime in a structure identical to the hierarchy of the application.

To create equipment names, EcoStruxure Process Expert may truncate the identifier of folders and instances to comply with the format of the name field of the equipment database file.

Changes to the structure, page 213 of the application are reflected on generated facets of the project allowing you to update the equipment database file by generating the project again.

NOTE: To view the equipment name that is generated, refine the Supervision project and open the equipment table.

Properties of the Equipment Table

The generation process populates the following properties of the equipment table of the Supervision project.

Field	Description
<i>Name</i>	Instance identifier with its path in the application, satisfying the equipment naming rules for the name field ⁽¹⁾ . The folder and instance identifiers are dot-separated. NOTE: The system root folder identifier is not included in the name.
<i>Cluster</i>	Identifier of the cluster of the Supervision project to which facets of the instance are associated.
<i>Comment</i>	<i>\$Description</i> parameter of the instance.
<i>Area</i>	<i>\$Area</i> parameter of the instance.
<p>(1) The Supervision Participant does not accept the use of certain reserved words for item names (for example, <i>DO</i>). While you can use these reserved words as identifiers of instances in an application, doing so prevents the Supervision project from generating successfully because the reserved word becomes the equipment name when the instance is created in the system root folder. For details on the reserved words, look up <i>Reserved Words</i> in the help of the Supervision Participant, page 95.</p>	

NOTE: You can use the **Equipment** element (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) to populate additional properties of equipment instances without impacting the way the equipment name is generated.

Rules Applying to the Creation of Equipment Names

EcoStruxure Process Expert may modify and/or truncate the identifier of folders and instances according to the following rules to populate the *Name* property of the equipment table of the Supervision project.

Item	Rule	Comment
Folder levels	7 folders maximum not including the system root folder	–
Instance path	125 characters maximum including the instance identifier and dot separators	–
Folder identifier length	12 characters maximum ⁽¹⁾	–
Instance identifier length	18 characters maximum ⁽¹⁾	By default, the identifiers of instances of Schneider Electric templates are limited to 18 characters.
Identifiers of folders	<p>Can only begin with a letter.</p> <p>Can only contain alphanumeric characters and underscores (_).</p> <p>No spaces.</p>	<p>Examples:</p> <p><i>\$Folder1</i> and <i>Folder\$1</i> are modified to <i>Folder1</i> (or <i>Folder100</i> if <i>Folder1</i> already exists)</p> <p><i>Folder 2</i> is modified to <i>Folder2</i></p>
<p>(1) EcoStruxure Process Expert may add an indexer consisting of two digits at the end of identifiers so that the name of each folder having the same parent folder and the name of each instance inside a folder remain unique.</p>		

For more information on not allowed characters, refer to the topic describing naming rules, page 31.

NOTE: If the instance identifier or path is truncated to satisfy the equipment name creation rules, a notification is displayed once the generation process is completed.

NOTE: The identifier of folders and instances in an application does not change.

Example 1 - Folder Identifier Exceeds Length

If in the **Application Explorer**, the following instance exists:

- *Conveyor_Motor_1*
- At path (folder identifiers): *System_1\PlantNorthFace\Unit 20*

The generated equipment name is: *PlantNorthFa.Unit20.Conveyor_Motor_1*

Example 2 - Indexing the Truncated Folder Identifier

If in the **Application Explorer**, the following two instances exist in two distinct folders:

- *Instance_1* at the path *System_1\Folder_12345_1* (the application folder name has 14 characters)
- *Instance_2* at the path *System_1\Folder_12345*

The generated equipment names are:

- *Folder_1234500.Instance_1*
- *Folder_12345.Instance_2*

Example 3 - Number of Folders Exceeds Limit

If in the **Application Explorer**, the following instance exists:

- *Motor_1*
- At path (folder identifiers): *System_1\F1\F2\F3\F4\F5\F6\F7\F8\F9*

The generated equipment name is: *F1.F2.F3.F4.F5.F6.F7.Motor_1*

Generation Diagnostic Messages

Message Description

Overview

The following tables describe some of the messages that are displayed when applicable rules are not satisfied during the generation of Control and Supervision projects and propose corrective actions.

In these messages and others, [notifications](#), [page 339](#) that are generated by the Control Participant may be displayed. These notifications provide more detailed information about the reason for which the generation process did not complete successfully.

For Control Projects

Message details	Cause	Corrective action
Section is overcrowded	<p>The section cannot accept as many function blocks as EcoStruxure Process Expert is trying to generate inside it, page 403.</p> <p>Also, you may have changed the position of DFBs during refinement so that EcoStruxure Process Expert cannot adapt the size of the section to accommodate the function blocks.</p>	<p>Do the following to free up space:</p> <ul style="list-style-type: none"> • Move or unassign facets from the FBD section. • Change the layout of the function blocks inside the section. • Verify whether a comment box is preventing EcoStruxure Process Expert from adapting the size of the section, page 403. <p>Then, generate the project/section again</p> <p>NOTE: When you move DFBs inside a section, use either method:</p> <ul style="list-style-type: none"> • Position DFBs tightly together to use the full width of the grid effectively, leaving empty rows at the bottom of the section. • Position DFBs to use the entire height of the grid, leaving empty columns on the right side of the section. • You can change the section size, page 405 by regenerating it.
Variable name conflict	A facet is generating a variable that has the same name as an already existing variable in the project.	<p>You can do either of the following:</p> <ul style="list-style-type: none"> • Change the instance \$Name parameter, page 182 so that the facet creates a variable with a different name. Then, generate the project again. • If you had already generated the project, refine the project and rename or remove the existing variable. Then, generate the project again. • Regenerate the Control project, page 434 in order to remove the existing variable (for example, if you had created the variable during refinement).
Version conflict in the same section	<p>EcoStruxure Process Expert is trying to generate, within the same section, a version of a DDT and/or DFB that is different from the version that already exists in the Control project.</p> <p>You may have updated the template used by an instance, page 224.</p>	<p>You can do either of the following:</p> <ul style="list-style-type: none"> • Update the template used by instances whose facets are assigned to the section so that only one version of the same type is generated. • Remove the type from the project if it is not used anymore. • Assign the facet that is generating the new version of the type to another section. <p>Generate the project again.</p> <p>For details, refer to the topic describing the management of type versions in Control facets during generation, page 409.</p>
Invalid characters	The name of a generated constituent contains a character that is invalid for use with the Control Participant.	Edit the properties of the instance referencing the facet that is mentioned in the message details so that resulting variable names and section identifiers satisfy the naming rules, page 32 , and generate the project again.

For Supervision Projects

Detected error	Cause	Corrective action
Unique equipment name cannot be generated. Indexer used for uniqueness has reached 99 (maximum limit).	The number of non-unique truncated folder and instance identifiers during generation exceeds 99.	Change the folder structure of the application or instance identifiers to reduce the number of non-unique identifiers that EcoStruxure Process Expert creates through truncation, page 438.

Managing Generated Facets

Working With Generated Facets

Overview

Following the generation of the Control or Supervision project, or parts of it, you can execute commands and actions on generated facets from the **Assignments** pane of the **Assignment Editor**.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Generated Facets Actions

The table outlines the commands and actions that you can execute on generated facets.

Command or action	Description
Unassign	<p>You can unassign facets that have the following Assignment status:</p> <ul style="list-style-type: none"> • Assigned • Out Of Date <p>If you unassign a Control facet that has the generation status of Unlinked, the facet is removed from the section but the generated constituents pertaining to the facet are not deleted from the logical Control Participant project. If you assign this facet again to the same project, you may receive a notification about a variable name conflict, page 440.</p> <p>The status UnAssigned is described in Assignment Status of Generated Facets, page 425.</p> <p>Refer to Unassigning Generated Facets, page 443.</p> <p>NOTE: The actions that are performed when you select the Unassign command for a generated facet or a facet that is not yet generated, page 391 are different.</p>
Unlink	<p>Refer to Unlinking Generated Facets, page 443.</p> <p>NOTE: The status Unlinked is described in Generation Status of Facets, page 424.</p>
Reassign	<p>You can reassign generated facets that have the status UnAssigned.</p> <p>The Reassign command assigns a facet to the container in which it is located, allowing you to:</p> <ul style="list-style-type: none"> • Apply to the facet changes to the instance that references it. • Generate its constituents into the logical Participant project. <p>EcoStruxure Process Expert performs a consistency check, page 460 when available.</p> <p>When the consistency check is not available, the generation status of facets changes to Inconsistent. This is the case, for example, for Supervision genie and data facets.</p> <p>Reassigning a facet changes its status to Assigned.</p> <p>Refer to Reassigning Facets, page 444.</p>
Relink	<p>You can relink generated facets that have the status Unlinked.</p> <p>The Relink command links a facet again to its already generated constituents, allowing you to:</p> <ul style="list-style-type: none"> • Apply to the logical Participant project changes of the facet using the Generate command. • Detect inconsistencies by performing a consistency check. <p>When you relink a facet, EcoStruxure Process Expert performs a consistency check if the functionality is available for the facet. Depending on the result, it changes the generation status of the facet to either:</p>

Command or action	Description
	<ul style="list-style-type: none"> • Generated if it detects no inconsistency. • Inconsistent if it detects an inconsistency. <p>When the consistency check is not available, the generation status of facets changes to Inconsistent. This is the case, for example, for Supervision genie and data facets.</p> <p>Refer to Relinking Facets, page 444.</p>
Go To Instance	<p>Allows you to go to the instance that references the facet.</p> <p>The navigation feature is not available if the Assignment status of the facet is Deleted.</p> <p>Refer to Navigating From Facets to Instance, page 395.</p>
Move	<p>Refer to Moving Generated Facets, page 444.</p> <p>NOTE: The status Moved is described in Generation Status of Facets, page 424.</p>

NOTE: For a description of the effect of a subsequent generation process on the facets that you have acted upon, refer to *Generating Projects After Changes*, page 429.

Unassigning Generated Facets

To unassign a generated facet from the container to which it is assigned, in the **Assignments** pane, right-click the facet and select **Unassign**.

Result:

- The facet is unassigned from the selected container.
- Its already generated constituents (for example, variables and FB instances for Control facets and database entries for Supervision facets) are removed from the logical Participant project (except derived data types, page 434 for Control facets).
- The status of the facet displayed in the **Assignment** column changes to **UnAssigned**.

NOTE: You can select several facets in the same container.

Unlinking Generated Facets

To unlink a facet, in the **Assignments** pane, right-click the facet and select **Unlink**.

Result:

- The facet is unlinked from its already generated constituents.
- The status of the facet displayed in the **Generation** column changes to **Unlinked**.

NOTE: You can select several facets in the same container.

Reassigning Facets

To reassign a facet, in the **Assignments** pane, right-click the facet and select **Reassign**.

Result:

- The facet is reassigned to the same container and a consistency check is performed.
- The status of the facet displayed in the **Assignment** column changes to **Assigned** or **Out Of Date**.
- The status of the facet displayed in the **Generation** column changes to **Generated** or **Inconsistent**, page 424.

NOTE: You can select several facets in the same container.

Relinking Facets

To relink a facet, in the **Assignments** pane, right-click the facet and select **Relink**.

Result:

- The facet is linked again to its generated constituents in the logical Participant project.
- The status of the facet displayed in the **Generation** column changes to **Generated** or **Inconsistent**, page 424.

NOTE: You can select several facets in the same container.

Moving Generated Facets

To move a facet from one container to another, proceed as follows.

Step	Action
1	In the Assignments pane, right-click any one of the column headers and select the appropriate command to clear filters and sorting, page 120.
2	<p>In the Assignments pane, drag the facet to another container in the Containers pane.</p> <p>Result:</p> <ul style="list-style-type: none"> • The facet is removed from its original location and assigned it to the target container. • The status of the facet displayed in the Generation column changes to Moved. <p>NOTE: You can select several facets in the same container.</p>

NOTE: Control facets that you move are assigned to the new section with the highest order value in the **Order** column of the **Assignments** pane, page 379. If you move several facets at once, the facet that you have selected first gets the lowest order out of the ones that are moved.

NOTE: After you have moved a generated facet from one container to another, *generate*, page 430 the target container again to update the associated logical Participant project.

Refinement Stage

What's in This Chapter

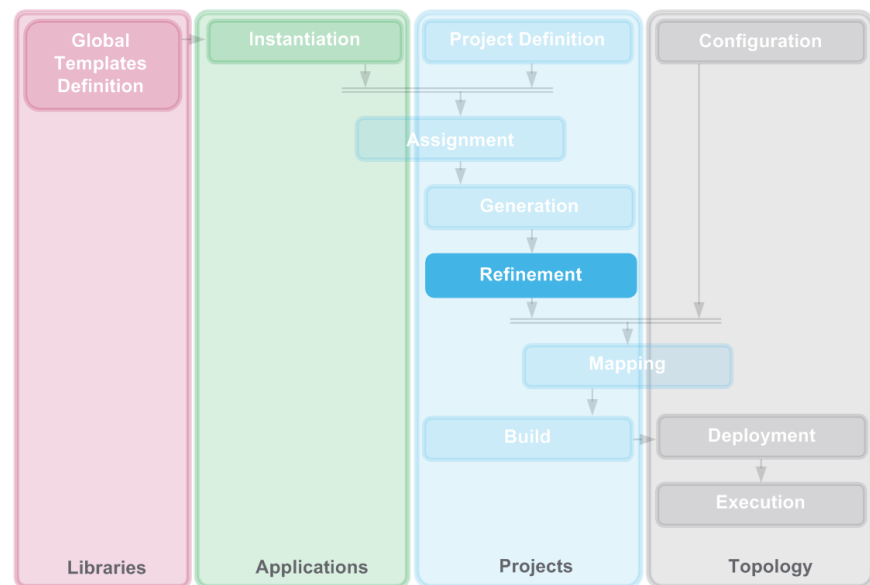
Control Project Refinement Stage	446
Managing Peer to Peer Communication	475
Supervision Project Refinement Stage	508

Overview

This chapter describes how to modify the logical Participant projects, page 54 that the EcoStruxure Process Expert created during the generation stage to fulfill the requirements of the system.

The refinement stage is not mandatory to be able to proceed to the next stage.

The following figure shows the position of the **Refinement** stage within the system engineering life cycle.



Refer to Refinement Stage, page 59 for a description of the purpose of this stage.

Control Project Refinement Stage

Refining the Logical Control Project

Overview

You can refine at the project level, which allows you to view and modify the entire logical Control project with the help of the Control Participant.

You can also refine a section by launching the refinement process from the section (except for LL984 sections). The section does not need to be generated to be refined.

You can import LL984 sections (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

FBD and non-FBD sections that you add during refinement are displayed in the Control project (**Assignment Editor**) in the order they appear in the Control Participant. The contents of these sections is not displayed there however.

Once refinement is complete, you can perform a consistency check, page 460.

At the refinement stage, you can also create and manage network variables, page 463, which are used for peer to peer communication.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Refining Offline With CCOTF Enabled

For M580 and M580 safety-related Control projects created in the **Project Explorer**, the Configuration Change On The Fly (CCOTF) capability of the Control Participant is enabled by default (the **Online modification in RUN or STOP** check box of the configuration of the Control project is selected).

To complement CCOTF for offline controller configuration, page 585, when you refine a Control project offline, EcoStruxure Process Expert informs you ahead of time if a modification that you are about to make will require stopping the controller when you deploy changes, page 830 to the controller thereafter. This information is available again in the **Impact on Engine** column of the **Deploy Changes / Undo Online Changes** window where modifications that can be deployed are listed.

With CCOTF enabled, a project analysis is automatically performed, page 272 when you save offline refinement changes.

The working principle is the same as for CCOTF when configuring a controller except for the following:

- If you have performed a CCOTF-incompatible modification, save changes, and close the Control Participant window, the next time you refine the same project offline, you continue receiving notifications about CCOTF-incompatible modifications. This, even if you have not yet deployed the last CCOTF-incompatible modification.
- Performing a CCOTF-incompatible modification does not disable the **Build** nor the **Deploy Changes / Undo Online Changes** command.

Before performing a CCOTF-compatible modification, refer to the topics describing the CCOTF functionality in the help of the Control Participant for your specific controller platform and follow the instructions.

NOTE: CCOTF is not supported for Control projects the M340 and Quantum controller families.

NOTE: CCOTF notifications are shown even if you have not yet deployed the Control project.

Firmware and Web Diagnostic/Data Storage Passwords

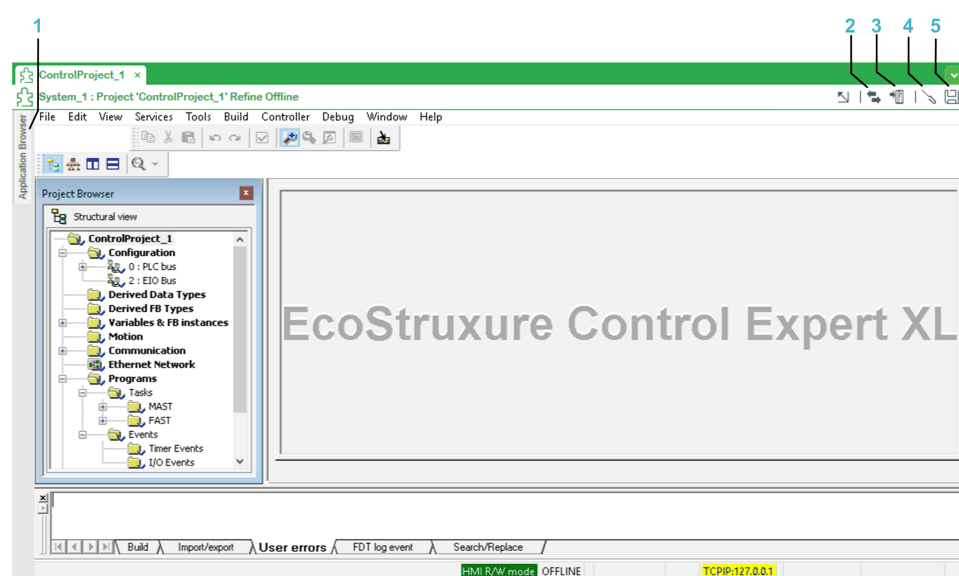
When you create an **M580** Control project, EcoStruxure Process Expert changes the default **Firmware** and **Web Diagnostic/Data Storage** passwords so that you do not need to set them to refine the project and build the executables, page 775. To change them, contact your local Schneider Electric representative.

If you map an executable, page 723 of this project to an M580 controller, the passwords that you set in the controller configuration (**Topology Explorer**) are used instead (for controllers with firmware version 4.00 and later).

NOTE: The default **Firmware** and **Web Diagnostic/Data Storage** passwords of **M580 Safety** Control projects is not changed.

Project Refinement Window

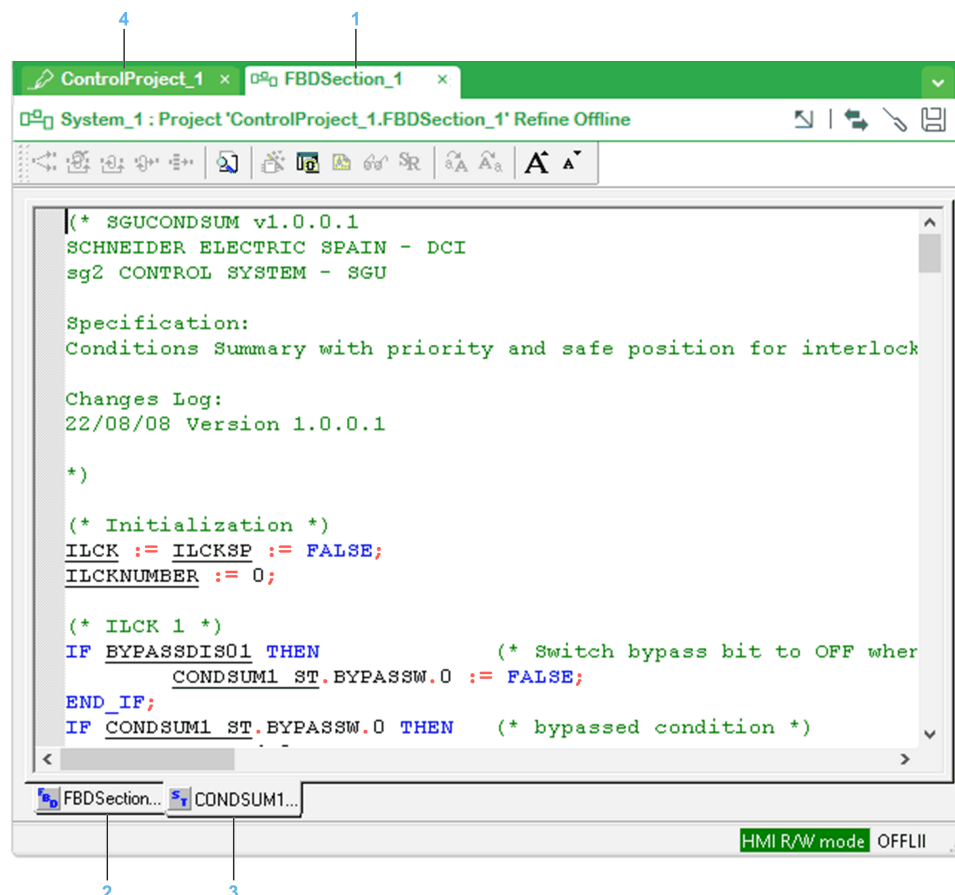
The figure shows the **Refine Offline** window embedding the Control Participant, which opens when you select the **Refine** command at the Control Participant project level. The window is shown restored.



Item	Description
1	Tab of the pane in which you can see the project sections in relation to the structure of the application
2	Manage network variables button
3	Access device DDTs of unmapped hardware button
4	Check consistency button
5	Save button

Section Refinement Window

The figure shows the **Refine Offline** window embedding the Control Participant, which opens when you select the **Refine** command at the section level. The window is shown restored.



Item	Description
1	Tab of the Control Participant window.
2	Tab of the section refinement window.
3	Tab of another refinement window (for example, a DFB refinement window), which you have opened from the section that is being refined.
4	Tab of another window that is open in the Project Explorer .

NOTE:

- To switch between open refinement windows, use the tabs at the bottom of the Participant window.
- To close a refinement window, right-click its tab and select **Close Window**.

Viewing the Application Structure When Refining Control Participant Projects

While you refine a Control Participant project, the **Application Browser** pane displays FBD sections based on the location of instances whose Control facets are assigned to these sections.

It shows the relationship between the folder structure of the application and the sections of the Control Participant project. This is useful, for example, when you want to view the logic related to a specific area or subdivision of the application. You can open sections from the pane.

The pane only shows FBD and non-FBD sections that exist in the logical Control Participant project at the moment you select the **Refine** command.

For a general description of the contents of the pane, refer to the topic describing the application structure view, page 384.

For information on how to use the pane, refer to the topic describing the refinement of Control Participant projects (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).

For general information on how to work with panes, refer to the topic describing the engineering client workspace, page 113.

Applying Changes to Generated Control Participant Projects

Selecting the **Refine** command applies to the logical Control Participant project the following changes that you have made in the **Project Explorer** to an already generated Control project:

- Renaming a section.
- Creating an FBD section.
- Changing the order of sections.

If the status of the executable of the Control project is **Built**, it is set to **Out Of Date**.

NOTE: You do not need to click the save button in the Control Participant window to apply the above mentioned changes.

Validity Indicators

When you save the logical Control Participant project during refinement and project analysis is enabled, page 272, the project is analyzed, page 401. The status of the project is indicated by the validity icons, page 336.

Details of the analysis are shown in the **Analyze Project** tab of the Control Participant window while it is still open.

If the logical Control project is not open in the Participant and the validity icon is displayed, you need to refine the Control project and analyze it. This allows you to identify the cause of the invalid status. The Control Participant features tools that help you identify validity issues, which result in an unsuccessful build, page 777.

For information on troubleshooting Control projects and/or analyzing them, refer to the help of the Control Participant.

NOTE: This verification is performed also when you generate a logical Control project, page 401.

Deleting Sections During Project Refinement

The table describes the result when you delete sections during refinement and the action that you need to perform to complete the deletion process depending on the type of section.

Type of section	Result after saving changes and closing the Control Participant window	Required action
Non-generated FBD section that was created in the Project Explorer .	The section still appears in the Containers pane but below other existing sections (higher order value) ⁽¹⁾ .	Right-click the section in the Containers pane and select Delete .
Non-generated FBD section that was created during refinement.		
Any non-FBD section.		
(1) LL984 segments deleted during refinement appear in last position (highest order value)		


Refining Non-Generated Sections

When you refine an FBD section to which facets are assigned but that is not generated yet, you do not see the constituents of these facets in the Control Participant.

If you add logic to this section during refinement, when you generate the section or project, page 401, DFBs that you have added are not moved, and DFBs coming from assigned facets are laid out around them.


Refining at the Control Project Level

To refine the logical Control project, proceed as follows.

Step	Action
1	In the Project Explorer , right-click the Control project you want to refine and select Refine . Result: The Control Participant opens extracted, page 118 and maximized.
2	Proceed with the refinement. For more information on the refinement process, refer to Refining the Logical Control Participant Project (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).
3	Click the save button in the toolbar of the Participant window to save the changes in the logical Control Participant project. 
4	Perform a consistency check, page 462 if required.
5	Click the close button in the Control Participant window. NOTE: If you have unsaved changes and click the close button in the Participant window, the Save Refinement dialog box opens where you have the following choices: <ul style="list-style-type: none"> • Yes: Saves changes in the logical Control Participant project and closes the Participant window. • No: Discards changes and closes the Participant window. • Cancel: Does not save changes and leaves the Participant window open.

Refining at the Section Level

To refine a section of the Control project (except for LL984 sections), proceed as follows.

Step	Action
1	In the Containers pane of the Assignment Editor , right-click the section that you want to refine and select Refine . Result: The Control Participant opens extracted, page 118 and maximized and displays only the contents of the selected section.
2	Proceed with the refinement of the section. For more information on the refinement process, refer to Refining the Logical Control Participant Project (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).
3	Click the save button in the toolbar of the Participant window to save the changes in the logical Control project. 

Step	Action
4	Perform a consistency check, page 462 if required.
5	<p>Click the close button in the tab of the Control Participant window.</p> <p>NOTE: If you have unsaved changes and click the close button in the Participant window, the Save Refinement dialog box opens where you have the following choices:</p> <ul style="list-style-type: none">• Yes: Saves changes in the logical Control Participant project and closes the Participant window.• No: Discards changes and closes the Participant window.• Cancel: Does not save changes and leaves the Participant window open.

Identifying Code Generated by EcoStruxure Process Expert

Overview

Constituents that are generated automatically from facets assigned to Control projects are locked in the Control Participant project. The locked code is color coded in the DFB editor of the Control Participant.

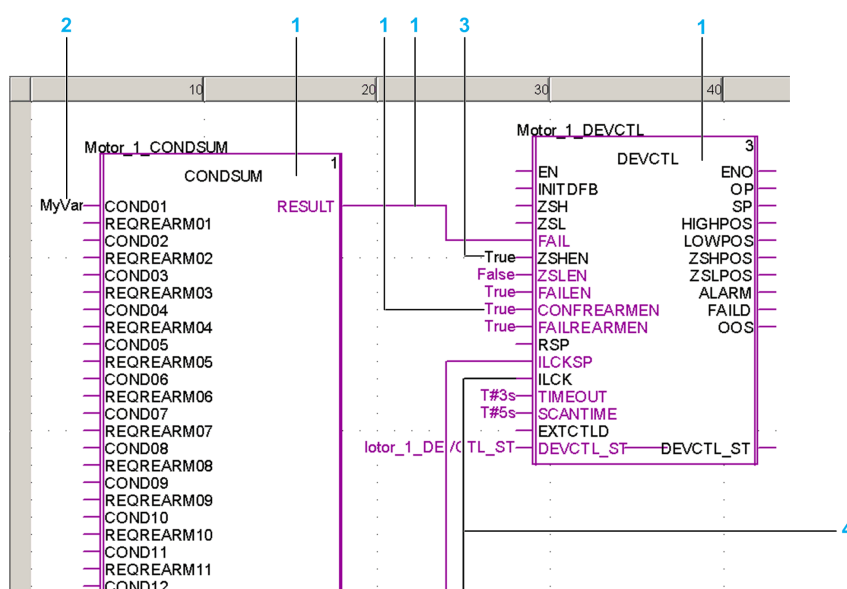
This feature helps you with the following:

- Distinguish code generated automatically from code that you modify and/or add during refinement.
- Avoid modifying inadvertently code generated automatically and maintain consistency. This is achieved by requiring a confirmation before unlocking code and providing the possibility to lock code again.

The locking of code is managed automatically throughout the system engineering life cycle.

Viewing Locked Code

The following figure shows an example of a Control Participant refinement window showing an FBD section, which contains code that was generated by some sample facets assigned to the section.



Item	Description
1	Code that is generated from the facet is locked and shown in purple.
2	Variable that was added by the user during refinement to code generated automatically. The code is not locked and shown in black.
3	Literal, previously shown in purple that was unlocked and changed by the user and now shown in black.
4	Link, previously shown in purple that was unlocked by the user and now shown in black.

Code that Is Locked

The following data is locked when generated from facets assigned to a Control project:

- DFBs and the pins to which an effective parameter is assigned.
- Effective parameters.

- Links (including links created by using an editor of the **Application Explorer**).
- EFB and DFB instances.

NOTE: FBD sections that you have generated are also locked but without visual indication.

Code that Is not Locked

The following data is not locked:

- Non-FBD sections.
- Pins of DFBs to which no effective parameter is assigned or that are not connected.
- Formal parameters of DFBs.
- Variables shown in the **Data Editor** of the Control Participant.
- Any code added by the user during refinement offline and online.
- FBD sections that are generated but to which no facets are assigned.

Modifying Locked Code During Refinement

To modify locked code during refinement offline and online, you need to unlock it first (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).

For example, changing the parameter that is assigned to the pin of a generated DFB.

However, adding code that does not modify generated code can be done without unlocking. For example, assigning a parameter to an empty pin of a generated DFB or adding an EFB in a section containing generated code.

NOTE: You cannot delete FBD sections that are locked.

Management of Locked Code in the System Engineering Life Cycle

The fact that code is locked is part of its attributes and managed automatically during the system engineering life cycle.

As such, unlocking code is detected as an inconsistency during a consistency check, [page 460](#) even if the code itself is unchanged. The status of the facet that generated the code is set to **Inconsistent**.

Code that was originally locked and that you have unlocked is:

- Locked again when the facet is generated and either condition applies:
 - The status of the facet is **Inconsistent**.
 - No consistency check was performed but the contents of the section was modified, [page 431](#).
- Detected when you use the following commands:
 - **Deploy Changes / Undo Online Changes**
 - **Update Project**

Considerations when Migrating Databases

When you refine a Control project of a database that was migrated from a previous version, code generated automatically may not be locked.

However, you can enable the code locking feature for such code in a migrated database if you perform a consistency check and generate the Control project.

Code generated from facets that are assigned to Control projects after the migration is locked.

The table describes the impact on code that is generated automatically and contained in a migrated database depending on the action that you perform and the status of facets.

Actions performed after migration	Status of facets assigned to Control projects in the migrated database	Result after generating the Control projects
No consistency check was performed.	Generated or Unlinked .	No change.
	Non Generated, Out Of Date , or Inconsistent .	The code of facets that were successfully generated is locked. FBD sections containing at least one such facet are locked.
A consistency check is performed.	All facets are set to Inconsistent .	The code of facets that were successfully generated is locked. FBD sections containing at least one such facet are locked.

NOTE: If you use the **Regenerate** command on the Control project after migrating the database, code that was generated automatically before the migration is locked even if you have not performed a consistency check.

Creating and Updating Device DDT Variables in Logical Control Projects

Overview

In the system engineering life cycle, you can engineer Control projects independently of the hardware that exists in the topology up to the build stage. As a result, device DDT variables, which are associated to hardware modules and managed by the Control Participant are not readily available when you refine a logical Control project offline.

This functionality lets you create in the logical Control project device DDT variables that are associated to the hardware present in the topology of the system. This makes it possible to use device DDT variables in the program during refinement offline, for example, to manage special modules or access hardware diagnostic data.

Using device DDT variables does not replace hardware mapping. Device DDT variables do not appear in the **Hardware Mapping Editor**.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Device DDT Variables Present in the Control Project by Default

When you create a Control project of the **M580** platform in the **Project Explorer**, it contains a default configuration with only a controller (although it is not to be used from the **Project Explorer** at this stage of the system engineering life cycle). Therefore, by default, the logical Control Participant project already contains the device DDT variable that is mapped to the M580 CPU module (a lock icon appears next to the DDDT variable). You can use it by refining the logical Control project.

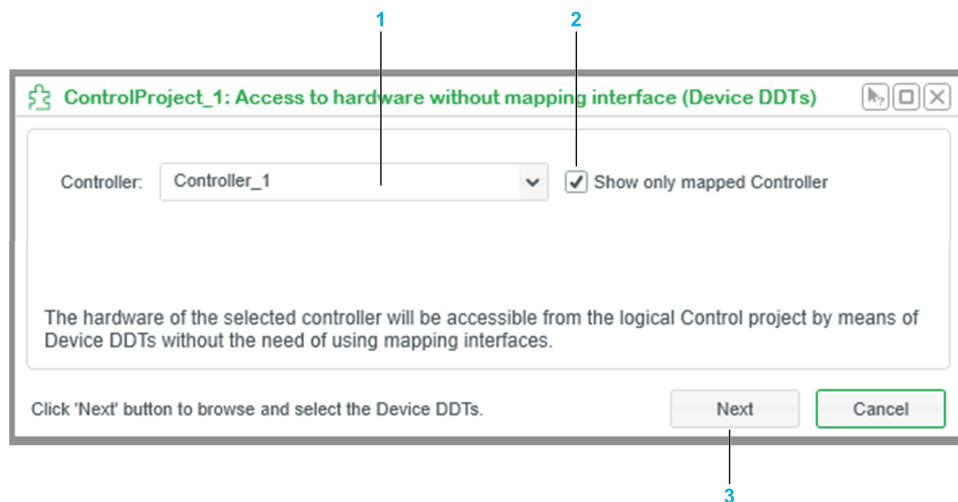
Prerequisites

The topological entities whose device DDT variables you want to create must exist in the topology of the system.

Accessing the Device DDT Variable Wizard

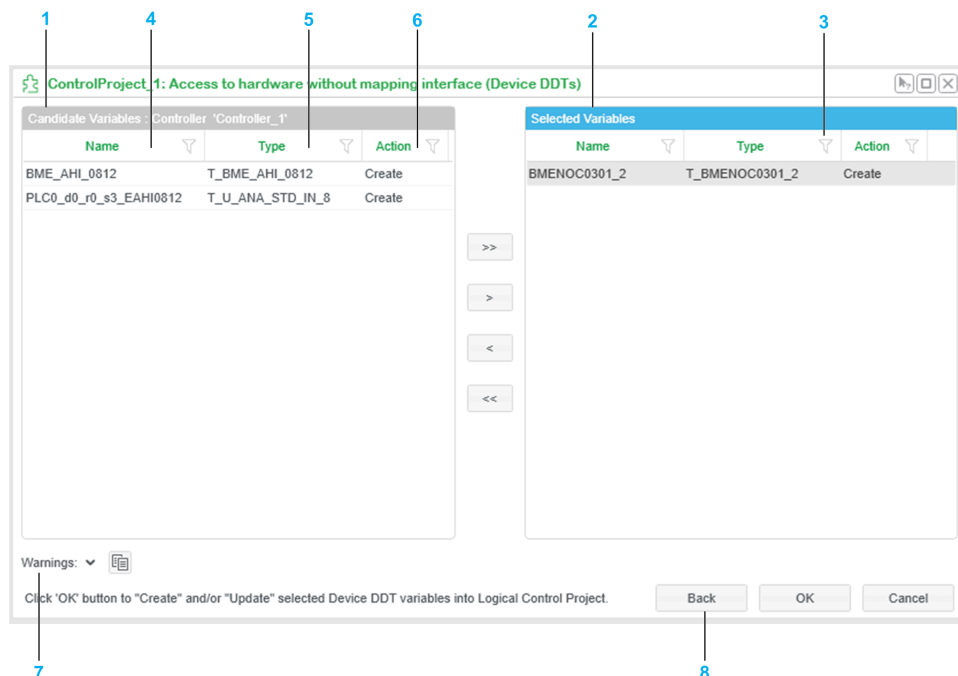
To open the wizard, refine the Control project and click the device DDT wizard button, page 447.

Device DDT Variable Wizard - Controller Selection



Item	Description
1	Box to select the controller whose device DDT variables you want to create in the logical Control project.
2	When the check box is selected, the menu shows only the controllers that are service mapped to an executable of the Control project. Otherwise, it shows the controllers that exist in the topology of the system.
3	Proceeds to the next screen and locks the controller in the Topology Explorer so that no other user can change its configuration or delete it.

Device DDT Variable Wizard - Device DDT Variable Selection



Item	Description
1	<p>Candidate pane that shows device DDT variables of the controller that you have selected and that you can create in the logical Control project. The following can be shown:</p> <ul style="list-style-type: none"> Variables that do not exist in the logical Control project yet. Variables and types that already exist but which have been modified in the controller. <p>NOTE: The following variables do not appear in the candidate pane:</p> <ul style="list-style-type: none"> Device DDT variables of the M580 CPU, page 455. Device DDT variables that already exist in the logical Control project with the same name but with a different type. This can be the case, for example, if after having created the device DDT variable for hardware module 1, you remove this module in the topology, add hardware module 2, which creates a device DDT variable of a different type, and rename this device DDT variable to have the same name as the variable that was associated to hardware module 1.
2	Selected variables panes that shows the device DDT variables that you have selected.
3	Filter icon, page 120.
4	Name of the device DDT variable.
5	Data type of the device DDT variable.
6	<p>Either action appears for each device DDT variable:</p> <ul style="list-style-type: none"> Create: Creates, page 457 the data type and the variable in the logical Control project. Update: Updates, page 457 the data type in the logical Control project.
7	<p>Notification area.</p> <p>Expand the area to view alerts on device DDT variables of the selected controller.</p>
8	<p>Reverts to the previous screen.</p> <p>To select device DDT variables from another controller, complete the process for your ongoing selection first. Any selection that you have not applied is discarded.</p>

Impact of the Create Action on the Logical Control Project

The table describes the impact when you create in the Logical Control Project a device DDT variable that appears in the candidate pane with the **Create** action. The impact varies depending on the scenario.

Scenario	Impact of the Create Action
The variable and the type are not yet present in the logical Control project.	Both are created in the logical Control project.
A type of the same name already exists in the logical Control project but not the variable.	<p>The variable is created.</p> <p>If the structure of the type of the selected variable is different, it overwrites the type that is present in the Control project. This may impact other existing variables of that type.</p>

Impact of the Update Action on the Logical Control Project

The table describes the impact when you update in the Logical Control Project a device DDT variable that appears in the candidate pane with the **Update** action. The impact varies depending on the scenario.

Scenario	Impact of the Update Action
The variable and the type already exist in the logical Control project but the structure of the selected type is different.	The structure of the type is updated in the logical Control project. This may impact other existing variables of that type.
The variable and the type already exist in the logical Control project but the version of the selected type is different.	The version of the type is updated in the logical Control project. This may impact other existing variables of that type.

Creating or Updating Device DDT Variables

To create or update device DDT variables in the logical Control project, proceed as follows.

Step	Action
1	In the device DDT variable wizard, select the controller whose device DDT variables you want to create or update and click Next .
2	Expand the notification area to view alert messages.
3	In the candidate pane, select one or more device DDT variables that you want to create/update and click OK . Result: The selected device DDT variables are processed by applying the corresponding action for each and the wizard is closed.
4	Save changes by clicking the save button, page 447 in the toolbar of the Control Participant window frame.

Modifying Device DDT Variables

Device DDT variables that you create in the logical Control project are not locked and you must not modify their structure or name. If you do, EcoStruxure Process Expert cannot treat them anymore as device DDT variables associated to the device from which they were created originally.

Deleting Device DDT Variables

You cannot remove a device DDT variable and/or its data type from the logical Control project by using the device DDT variable wizard.

To remove the variable, select it in the Control Participant window while refining the logical Control project and select **Delete**. Then, purge the type by using the **Types Library manager**

Considerations When Removing Modules in the Topology Explorer and Deploying Changes

When the topology of the system contains entities and you had created the corresponding device DDT variables in the logical Control project, built the project, and deployed it, if you delete the source of the device DDT variable from the topology (for example, a module or DTM), verify that the corresponding device DDT variable is removed in the configuration of the **Topology Explorer**. If not (for example, when you delete an analog input module), you need to purge the corresponding device DDT variable.

Otherwise, if you build the logical Control project after removing the source of the device DDT variable from the topology, it remains in the built Control project as unmapped variable (while it is present in the deployed Participant project as mapped variable). If you deploy changes, EcoStruxure Process Expert proposes to add the unmapped device DDT variable.

Integrating Device DDT Variables in the System Engineering Life Cycle

To use device DDT variables that you have created logical Control project, follow the stages of the system engineering life cycle described in this document (build, page 777, build changes, page 782, deployment, deployment of changes, page 836, refinement online, and so on).

Using Device DDT Variables in the Supervision Runtime

Device DDT variables do not have the HMI attribute enabled. To use such a variable in Supervision, you need to connect it to the input pin of a DFB encapsulated in a template that features Supervision services.

Managing the Consistency of Control Logic

Overview

Consistency management is a process that encompasses the following steps:

- Launching the consistency check of a logical Control Participant project or FBD section of such project.
- Indicating if the constituents of facets that were automatically generated in such project or FBD section are not consistent with the related facets that are assigned to the Control project or section. In this case, a message informs you of the detected inconsistencies and the **Generation** status of the related facets is changed to **Inconsistent**. Otherwise, a message informs you that no inconsistency was detected.
- Attempting to fix the detected inconsistencies by restoring the constituents inside the logical Control Participant project or FBD section to a state that is consistent with the related facets of the Control project or section.

The purpose of consistency management is to provide means for you to know whether changes that you have made during refinement affect the logical Control Participant project that was created during generation and, if desired, to fix inconsistencies when it is possible without modifying the refinements you made.

EcoStruxure Process Expert performs a consistency check in the following way when you:

- Generate again an already generated Control project or FBD section: Depending on the changes you have made, page 431.
- Reassign, page 442 a facet with the status **UnAssigned**: Automatically (if available).
- Relink, page 442 a facet with the status **Unlinked**: Automatically (if available).
- Update the logical Control Participant project, page 897: On request.

NOTE: No consistency check is performed on facets that have the **Unlinked** status, page 424.

Rules for Consistency Checks

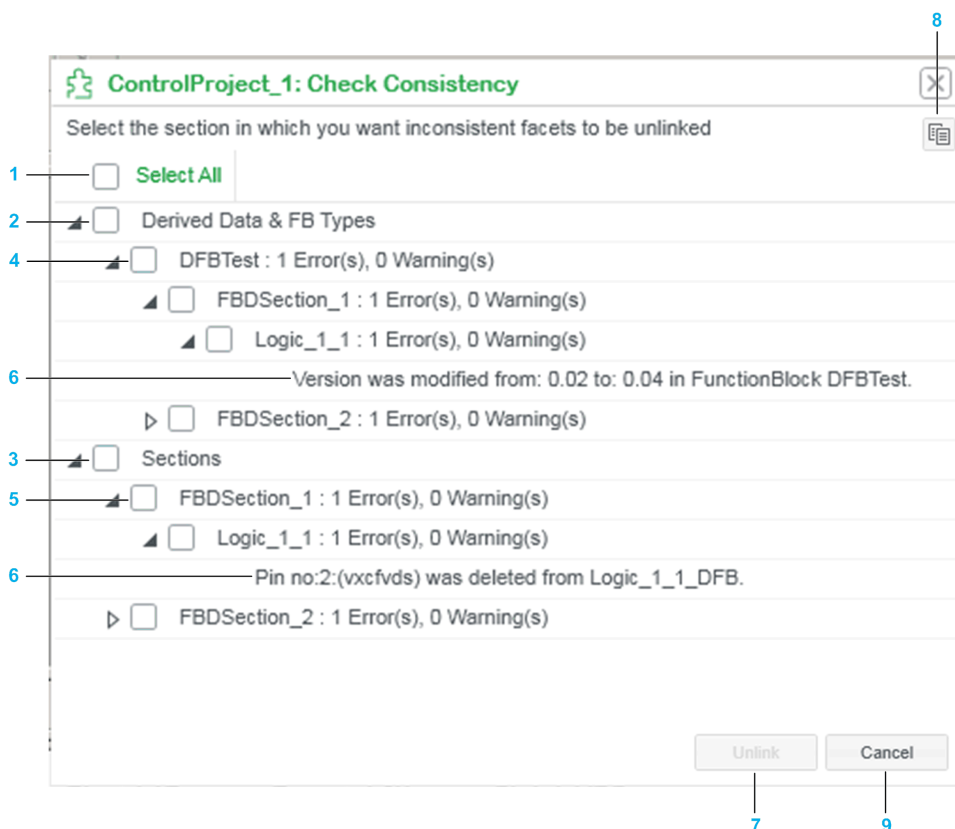
The table indicates in which cases EcoStruxure Process Expert detects and reports an inconsistency when performing a consistency check on constituents that are generated in a logical Control Participant project. It also indicates the resulting status of the facet encapsulating the constituent.

Type	Change made through refinement	Generation status of the facet
Derived data (DDT) and function block (DFB) types	Any change resulting in a new version of the type ⁽¹⁾ .	Changes to Inconsistent .
	Change causing no version change but resulting in a new type or code signature ⁽¹⁾ .	
Variables	Changing an attribute of a variable.	
	Removing markers created automatically (for example <i>P2P</i>).	No change.
FBD sections	Changing function calls (EFs, EFBs, or DFBs) in an FBD section.	
	Changing the relative execution order of function blocks (EFs, EFBs, or DFBs).	
	Changing links. Includes changing immediate values and links between variables.	Changes to Inconsistent .

Type	Change made through refinement	Generation status of the facet
–	Unlocking locked data, page 452.	Changes to Inconsistent .
(1) The change may also have been made by encapsulating a modified type in a Control facet and updating the template with this facet. However, starting with Process Expert 4.2 SP2, you cannot save modifications to the type encapsulated in a Control facet if the version of the type is still the same but its type or code signature different.		

Check Consistency Dialog Box

The figure shows an example of the **Check Consistency** dialog box.




Item	Description
1	Check box to select all FBD sections and facets that appear in the two categories of the dialog box. By default no section or facet is selected.
2	Category listing FBD sections containing facets for which inconsistencies related to their DDT and DFB types have been detected. Select the check box to select all sections and facets that appear in this category.
3	Category listing FBD sections in which inconsistent facets have been detected. Select the check box to select all sections and facets that appear in this category.
4	Name of a type for which an inconsistency has been detected. Expand the node to view the facets generating this type and the sections to which these facets are assigned.
5	Identifier of a section to which facets are assigned, which are inconsistent. Expand the node to view the facets.
6	Description of the detected inconsistency.
7	Unlinks the selected facets so that no consistency check is performed on them.
8	Copies the content of the dialog box to the Clipboard in text format. The status of check boxes is not copied.
9	Cancels your selection and closes the dialog box.

NOTE: When a section or facet appears in both categories, selecting it in one category automatically selects it in the other, and the other way around, clearing its selection in one category automatically clears it in the other.

Performing a Consistency Check

To perform a consistency check on generated constituents, proceed as follows.

Step	Action
1	In the Project Explorer , right-click the Control project or FBD section that you want to check for consistency and select refine . Result: The Control Participant opens in a reduced window.
2	In the toolbar of the window, click the check consistency button  . Result: <ul style="list-style-type: none"> • A consistency check is performed. • The Check Consistency dialog box opens . • The Generation status of facets referencing inconsistent constituents changes to Inconsistent. NOTE: The check consistency button is not available when you refine a non-FBD section.
3	In the Check Consistency dialog box, select inconsistent sections and/or facets that you want to unlink.
4	Click Unlink to unlink the selected facets. NOTE: Click Cancel to close the Check Consistency dialog box without unlinking facets. The status of inconsistent facets remains Inconsistent .

Managing Inconsistencies

Even though a logical Control Participant project contains inconsistent facets, you can proceed further in the system engineering life cycle.

Regenerate the FBD section or project containing inconsistent facets to revert to a consistent state, which is indicated by the generation status **Generated** of the facet but certain changes that you made through refinement are discarded, page 434.

For information on managing inconsistencies without affecting changes made through refinement, refer to [Generating Inconsistent Facets](#), page 432.

About Variables for Peer to Peer Communication

Overview

EcoStruxure Process Expert facilitates the creation of variables to exchange scattered data or sets of data between two or more Control Participant projects by using peer to peer communication. Such variables are called *network variables*.

You can create network variables in the following ways:

- By assigning facets of instances to different projects.
- By refining both logical Control Participant projects. Refer to Refining the Logical Control Participant project (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).
- By using peer to peer communication templates of the EcoStruxure Process Expert Foundation Library.
- By using a mix of the above.

Depending on the way that you use to create them, network variables have the *candidate* status until they are explicitly created for use as network variables by using the **Manage Network Variables** dialog box, page 471.

You can create network variables before proceeding with the service mapping and the creation of communication channels, page 725.

Variables used for peer to peer communication are allocated during build, page 778 based on the network variable mapping, page 738.

NOTE: For more information on the steps to implement peer to peer communication by using the I/O scanner function, refer to the topic describing peer to peer communication by Modbus TCP Ethernet implicit messaging, page 935.

Using the Control Participant Peer to Peer Communication Functionality

EcoStruxure Process Expert also automates the implementation of peer to peer communication by using the specialized *READ_REMOTE* and *WRITE_REMOTE* function blocks of the Control Participant instead of or in addition to using the peer to peer communication templates of the EcoStruxure Process Expert Foundation Library.

For details, refer to the topic the describing the management of peer to peer communication, page 475.

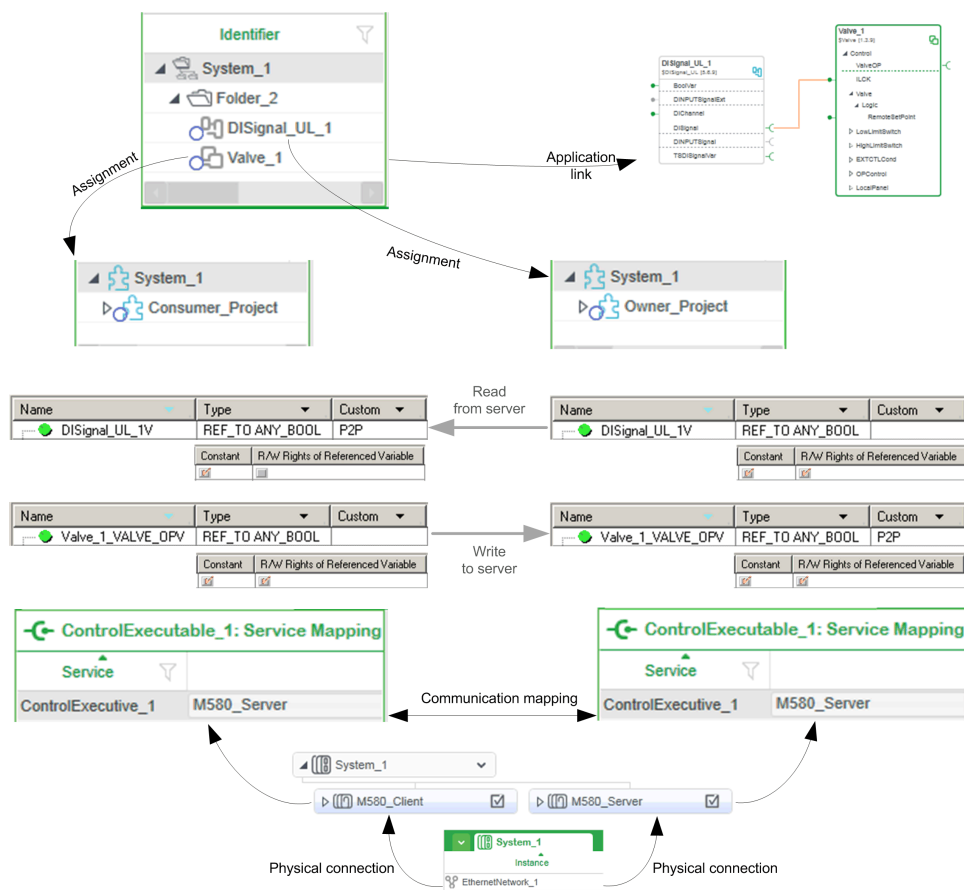
Owner/Consumer Project Principle

To help you understand the way peer to peer communication is implemented, the following naming conventions are used to refer to the Control Participant projects that exchange data:

- The logical Control Participant project producing the variables to be shared is the *owner* project, and the one receiving the variables is the *consumer* project.
- The controller that is mapped to the owner project acts as the server. The one mapped to the consumer project acts as the client.
- A network variable allows you to read data from the server by mapping it to a communication channel that exists client-side.

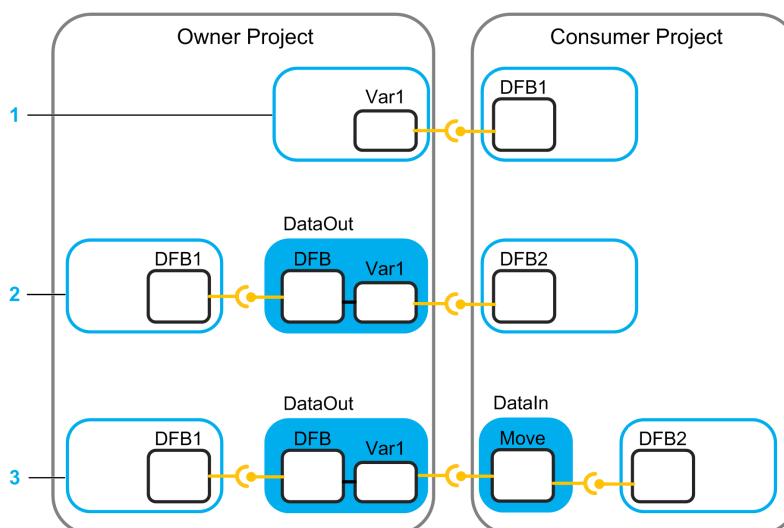
NOTE: You can write data to the server by using a network variable that exists in the owner project and the corresponding variable of same name and type in the consumer project.

The following example illustrates the owner/consumer principle in a system where sample variables *DISignal_UL_1V* and *Valve_1_VALVE_OPV* are used for peer to peer communication to read from and write to the server respectively. The variables are generated by facets of the *DISignal_UL_1* and *Valve_1* sample instances respectively. The specific attributes of the variables of reference data type are shown. The two instances are linked and their facets are assigned to the *Owner_Project* and *Consumer_Project* Control Participant projects. The executable of each project is mapped to the corresponding topological entity. The read from server and write to server actions are performed from the consumer (client) project.



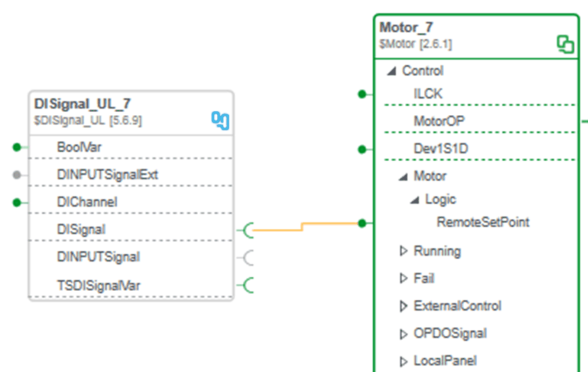
Ways to Create Network Variables to Share Scattered Data One-to-One

The first two scenarios that are shown in the following figure illustrate two common ways to create a network variable to share scattered data between an owner and a consumer project. Other ways to create network variables exist, page 468.

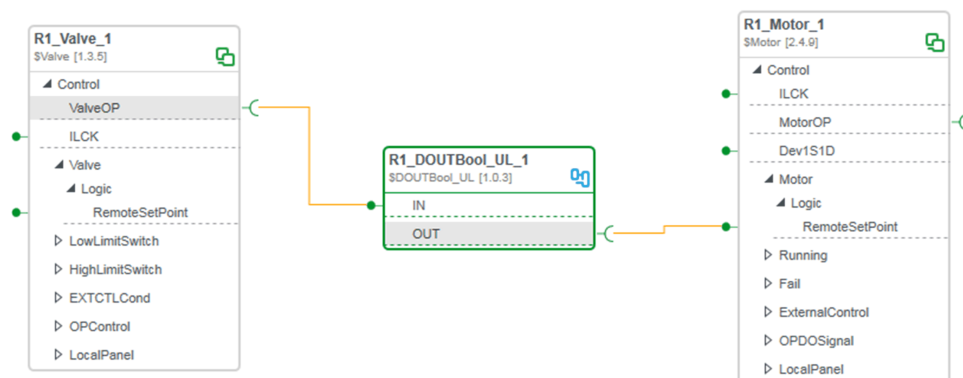


Scenario	Description
1	When the instance in the owner project creates a variable (for example, an instance of a hardware abstraction layer (HAL) template), it can be linked directly to the input pin of a DFB (created, for example, by an instance of an application template) in the consumer project.
2	When the instance in the owner project does not create a variable (for example, an instance of an application template), the output pin needs to be linked first to an instance of a peer to peer communication template (for example, Data Out) in the owner project. The Data Out instance creates a variable, which can be linked directly to the input pin of a DFB (created, for example, by an instance of an application template) in the consumer project.
3	In the first two scenarios, only a network variable candidate is created. You need to create the network variable manually by using the Manage Network Variables command, page 471. This third scenario shows how to create the network variable automatically by using two peer to peer communication templates (Data Out and Data In).

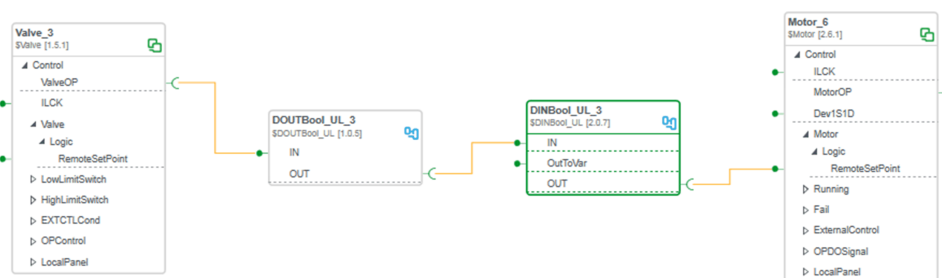
The following example shows how to implement the first scenario by using sample instances and linking them in the **Asset Workspace Editor**. Facets of each instance are then assigned to the owner and consumer projects.



The following example shows how to implement the second scenario by using sample instances and linking them in the **Asset Workspace Editor**. Facets of each instance are then assigned to the owner and consumer projects.

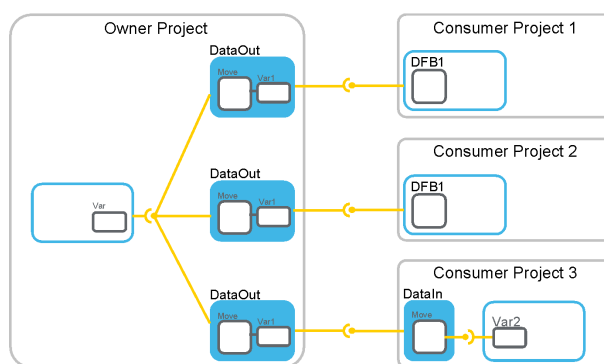


The following example shows how to implement the third scenario by using two peer to peer communication templates to create the network variable automatically. Instances are linked in the **Asset Workspace Editor**.

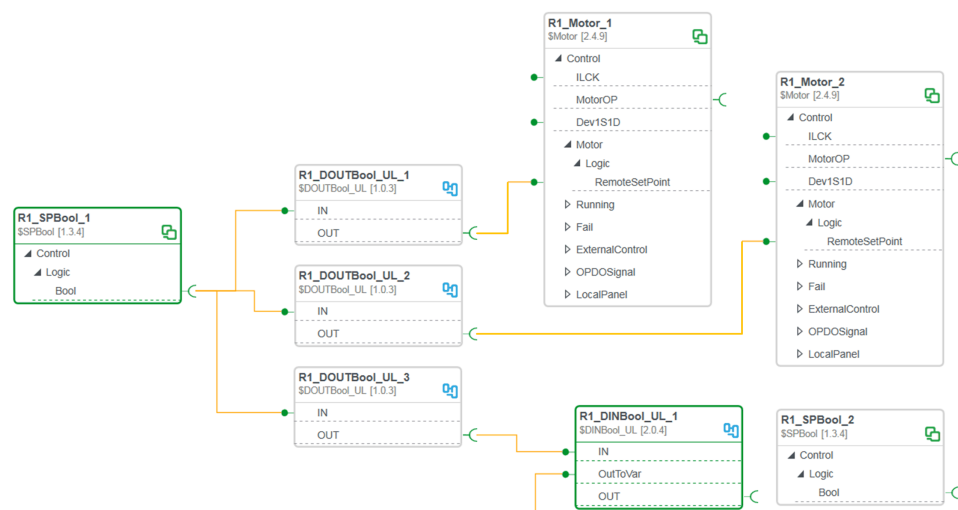


Ways to Create Network Variables to Share Scattered Data One-to-Many

The following figure illustrates the use of Data In and/or Data Out application templates to share one variable exposed by one owner project with three different consumer projects of the same system.



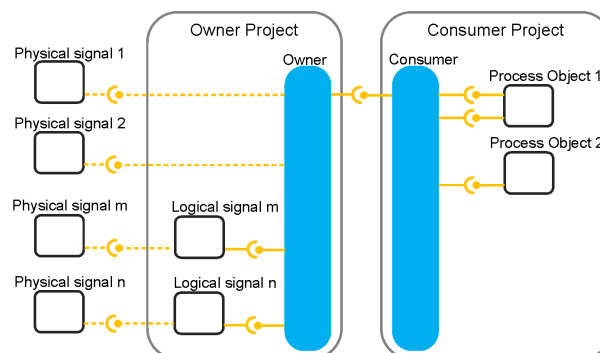
This example illustrates the scenario by using sample instances and linking them in the **Asset Workspace Editor**. Facets of each instance are then assigned to the owner and the three consumer projects.



Ways to Create Network Variables to Share Sets of Data

The following figure illustrates the use of owner and consumer application templates to share physical signals:

- Without signal conditioning.
- With signal conditioning by using a HAL template.



NOTE: In both cases, the physical signals are linked to the owner template and the HAL templates respectively by using mapping interfaces (dotted lines) during the hardware mapping.

For more information on owner/consumer templates and a usage example, refer to the topic describing these templates (see EcoStruxure™ Process Expert , Foundation Application Templates, User Guide).

Properties of Variables for Peer to Peer Communication

Certain attributes of variables that you create through refinement to be mapped to peer to peer communication channels require a specific configuration.

The configuration varies depending on:

- Whether the variable is used to read from the server or write to the server, page 463.
- The platform or the communication module that is used by the topological entity acting as client.

The table describes required variable attribute values in the Control Participant when the topological entity acting as client communicates by using a Quantum CPU module or an NOE communication module (with Quantum or M340 platform).

Usage	Variable attribute	Owner project/Server	Consumer project/Client
Reading data from the server	Designation of the variable in the context of EcoStruxure Process Expert	Variable	Network variable
	Name	Any valid name	Same name
	Type	Any, including the Reference data type (REF_TO). Some restrictions apply, page 471.	Same type
	Custom	Blank	<i>P2P</i>
	Constant⁽¹⁾	True (selected)	True (selected)
	R/W Rights of Referenced Variable⁽¹⁾	True (selected)	False (cleared)
Writing data to the server	Designation of the variable in the context of EcoStruxure Process Expert	Network variable	Variable
	Name	Any valid name	Same name
	Type	Any, including the Reference data type (REF_TO). Some restrictions apply, page 471.	Same type
	Custom	<i>P2P</i>	Blank

Usage	Variable attribute	Owner project/Server	Consumer project/Client
	Constant ⁽¹⁾	True (selected)	True (selected)
	R/W Rights of Referenced Variable ⁽¹⁾	True (selected)	True (selected)
(1) For variables of the Reference data type (REF_TO) only			

The table describes required variable attribute values in the Control Participant when the topological entity acting as client communicates by using an M580 CPU module with or without NOC communication module, or a Quantum controller with NOC communication module.

Usage	Variable attribute	Owner project/Server	Consumer project/Client
Reading data from the server	Designation of the variable in the context of EcoStruxure Process Expert	Variable	Network variable
	Name	Any valid name	Same name
	Type	Reference data type (REF_TO) only	Same type
	Custom	Blank	P2P
	Constant	True (selected)	True (selected)
	R/W Rights of Referenced Variable	True (selected)	False (cleared)
Writing data to the server	Designation of the variable in the context of EcoStruxure Process Expert	Network variable	Variable
	Name	Any valid name	Same name
	Type	Reference data type (REF_TO) only	Same type
	Custom	P2P	Blank
	Constant	True (selected)	True (selected)
	R/W Rights of Referenced Variable	True (selected)	True (selected)

NOTE: For more information on the REF_TO data type, refer to the topic describing the refinement of Control projects (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).

Overview of Ways to Create Network Variables

The table provides an overview of the different ways to create network variables in the consumer project to read data from the controller of the owner project, acting as server. It also indicates the status of the variable for each method.

Action in owner project	Action in consumer project	Status of variable
Instantiate a Data Out template corresponding to the data type to be shared (for example, \$DOUTBool_UL to share a boolean variable of the REF_TO ANY_BOOL data type).	-	Once you generate both projects, the Manage Network Variables dialog box detects the variable as a network variable candidate in the consumer project, with the name of the Data Out template instance. Create the network variable, page 471.
	Instantiate a Data In template corresponding to the data type of the Data Out template (for example, \$DINBool_UL).	The generation process creates a network variable in the consumer project with the name of the Data Out template instance. NOTE: The Manage Network Variables dialog box shows the variable in the Existing Network Variables pane.

Action in owner project	Action in consumer project	Status of variable
Instantiate a peer to peer owner template corresponding to the data type to be shared (for example, <i>\$Bool16/NO</i> to share up to 16 boolean input signals of the REF_TO data type).	Instantiate a peer to peer consumer template matching the peer to peer owner template (for example, <i>\$Bool16/INC</i> to receive up to 16 boolean input signals).	The generation process creates a network variable in the consumer project with the name of the owner template instance. NOTE: The Manage Network Variables dialog box shows the variable in the Existing Network Variables pane.
Instantiate a template, which creates a variable during the generation process (for example, a variable of type REF_TO ANY_BOOL ⁽¹⁾).	Reference a variable of the same name and data type as the one created by the instance in the owner project without declaring it by using the Refine command.	The Manage Network Variables dialog box detects the variable as a network variable candidate in the consumer project. Create the network variable, page 471.
	Instantiate a template that receives the variable exposed by the instance in the owner project by creating a link between both templates in the Asset Workspace Editor .	
	Create a variable of the same name and data type as the one created by the instance in the owner project, and enter <i>P2P</i> as Custom attribute by using the Refine command.	The Manage Network Variables dialog box shows the variable in the Existing Network Variables pane.
Create a variable ⁽¹⁾ to be shared by using the Refine command.	Reference a variable of the same name and data type without declaring it, using the Refine command.	The Manage Network Variables dialog box detects the variable as a network variable candidate in the consumer project. Create the network variable, page 471.
	Create a variable of the same name and data type, and enter <i>P2P</i> as Custom attribute by using the Refine command.	The Manage Network Variables dialog box shows the variable in the Existing Network Variables pane.

(1) Verify that the data type of the variable that is created meets the requirements to be used for peer to peer communication, page 467.

NOTE:

- The table does not detail each step required to implement peer to peer communication. For more information, refer to *Peer to Peer Communication Through Modbus TCP Implicit Messaging*, page 943.
- To be detected and/or created as network variables by the **Manage Network Variables** dialog box, variables need to satisfy the network variable management rules, page 471.

Overview of Peer to Peer Templates

The following templates are available from the Global Templates Library to create network variables:

- Data IN/Data OUT: Application facet templates that create a network variable in the consumer project. The Library contains templates for sharing various variable sizes and data types. Use these templates to share scattered data.

For a detailed description of the templates and how to use them, refer to Data In/Data Out Templates (see *EcoStruxure™ Process Expert*, Foundation Application Templates, User Guide).

NOTE: To find these templates by using the template browser of the **Application Explorer**, select the **Facets** filter, page 165, and enter either `din` or `dout` in the search field.

- Owner/Consumer: Application composite templates that create a network variable in the consumer project. The Library contains templates for sharing up to 16 digital and 8 analog inputs/outputs. Use these templates to share sets of data. Typically, such templates allow you to manage I/O objects exchanged between a Modicon BMX PRA module and a master controller.

For a detailed description of the templates and how to use them, refer to Owner/Consumer Templates (see *EcoStruxure Process Expert*, Foundation Application Templates, User Guide).

NOTE: Network variables created by using peer to peer templates have the **Custom** attribute set to *P2P* by default in the variable properties window of the Control Participant.

Managing Network Variables

Overview

This topic describes how to use the **Manage Network Variables** dialog box to manage network variables, which exist or have the candidate status.

Network variable management includes the creation, modification, and deletion of network variables.

General Network Variable Management Rules

EcoStruxure Process Expert applies either of the following rules to identify candidate network variables when you open the **Manage Network Variables** dialog box:

- The variable needs to be referenced from, at least, 1 FBD section but not be declared.
- The variable is created by using a peer to peer template, page 464.

In addition, the following are not considered network variable candidates:

- Variables referencing an element of a data structure (for example, *DigitalInput_DINPUT_ST.STW* where *STW* is a word of the *DINPUT_ST* structure).
- Function block instances (for example, *DigitalInput_DINPUT.PV* where *PV* is a formal parameter of the *DINPUT* DFB).
- Types *ANY_X* where *X* represents the type (except for the *ANY_BOOL* data type).
- Inconsistent types (for example, *a.b* where *a* and *b* are variables and one of them is undefined).
- 32-bit data types.

NOTE: In the **Manage Network Variables** dialog box, it is not taken into account whether the data type of the variable is compatible with the communication module client-side, page 467.

Network Variable Properties

Network variables created by using the **Manage Network Variables** dialog box have the following properties:


- They are of the same type as shown in the **Type** field of the **Network Variables Candidates** section.
- They have *P2P* as **Custom** attribute in the variable properties window of the Control Participant.

NOTE: EcoStruxure Process Expert does not set the **R/W Rights of Referenced Variable** attribute for REF_TO type network variables that it creates. If it needs to be enabled, you need to refine the consumer project and set the attribute manually (see EcoStruxure™ Process Expert, Control Participant Services, User Guide) in the variable properties window of the Control Participant.

Opening the Manage Network Variables Dialog Box

You can open the **Manage Network Variables** dialog box either:

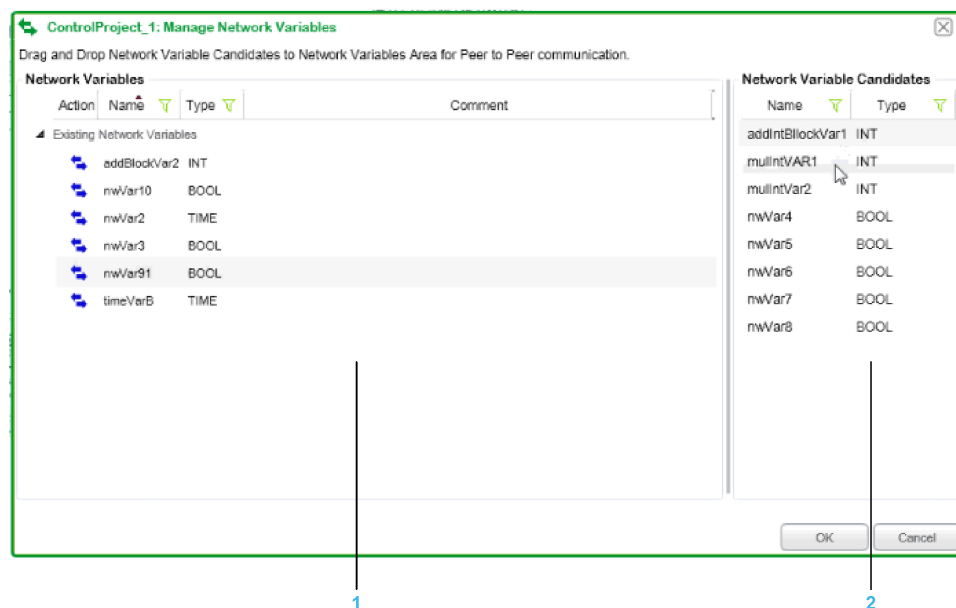
- From the context menu of the Control Participant project in the **Project Explorer** by selecting **Manage Network Variables**.

- From within the Control Participant by clicking the  button (see *EcoStruxure Process Expert, Control Participant Services, User Guide*). When you refine at the section level, the button is available only for FBD sections.

Both commands open the **Manage Network Variables** dialog box. The difference lies in the way you save changes.

Manage Network Variables Dialog Box

The following figure shows an example of the **Manage Network Variables** dialog box in which several sample variables appear.



Item	Description
1	Network Variables section showing existing network variables.
2	Network Variable Candidates section showing the candidate variables were detected in the selected project. These variables satisfy the applicable rules and you can them use as network variables. Certain restrictions apply depending on the communication module used by the topological entity acting as client.

Fields of the **Network Variables** section.

Header	Description
Action	<ul style="list-style-type: none"> Left/right arrow icon: Existing network variable. Minus icon: The variable will be removed from the project. Plus icon: The variable will be added to the project. Pencil icon: The variable will be modified in the project. Lock icon: The variable is already mapped to a peer to peer communication channel in the communication mapping, page 725.
Name	Identifier of the variable. The field is editable, page 474.
Type	Data type of the variable
Comment	You can enter a comment for the variable using free form text.

Fields of the **Network Variable Candidates** section.

Header	Description
Name	Identifier of the variable. When you use HAL or peer to peer templates, the name of the variable is the identifier of the template instance.
Type	Data type of the variable.

NOTE: You can select multiple variables in either sections by selecting the first variable, pressing and holding **Ctrl**, and selecting the other variables by using the pointer.

Saving Changes in the Manage Network Variables Dialog Box

Depending from where you have opened the **Manage Network Variables** dialog box, the procedure to apply and save your changes varies.

If you have opened the **Manage Network Variables** dialog box from the context menu of the Control project, proceed as follows to apply and save your changes.

Step	Action
1	In the Manage Network Variables dialog box, click OK . Result: A confirmation dialog box opens.
2	Click Yes to save your changes. NOTE: <ul style="list-style-type: none"> Click No to close the Manage Network Variables dialog box without saving changes. Click Cancel to leave the Manage Network Variables dialog box open without saving changes.

If you have opened the **Manage Network Variables** dialog box from within the Control Participant during refinement, proceed as follows to apply and save your changes.

Step	Action
1	In the Manage Network Variables dialog box, click OK to apply the changes to the logical Control project. Result: The Manage Network Variables dialog box closes. NOTE: If you open the Manage Network Variables dialog box again before closing the Control Participant, your changes are preserved but are not saved yet.
2	Click the save button in the toolbar of the Control Participant window to save your changes. NOTE: Closing the Control Participant without saving does not apply your changes.

Creating Network Variables

To create a network variable by using the **Manage Network Variables** command from the context menu of the Control project, proceed as follows.

Step	Action
1	<p>In the Projects Explorer, right-click the logical Control project where you want to identify network variable candidates and select Manage Network Variables.</p> <p>Result: The Manage Network Variables dialog box opens. If EcoStruxure Process Expert did not detect any candidate or network variables, the corresponding dialog box fields are empty.</p>
2	<p>In the Network Variable Candidates section, select the candidate variables that you want to create as network variables and drag them to the Network Variables section.</p> <p>Result: The network variables are shown in the New Network Variables section.</p> <p>NOTE: Right-click a network variable to open a context menu, which allows you to make changes, page 474.</p>
3	<p>Save your changes, page 473.</p> <p>Result: The dialog box closes, the network variables are created, and a summary of the actions appears in the notification panel.</p> <p>NOTE: You can see the variable by refining the Control Participant project and double-clicking Elementary Variables in the Variables & FB Instances section of the Project Browser.</p>

Modifying Network Variables

NOTE: Depending from where you have opened the **Manage Network Variables** dialog box, there are different ways to save your changes, page 473.

You can perform the following modifications on network variables displayed in the **Network Variables** section:

Action	Description
Adding a comment	Double-click the Comment field of the variable and enter a description with free form text.
Changing the name of the variable	<p>Double-click the Name of the variable and enter a new name with free form text.</p> <p>NOTE: You can only change the name of existing network variables that are not yet mapped to a communication channel.</p>
Undoing changes	<p>Right-click a variable and select Undo to discard the changes that you have made.</p> <p>You can also press Ctrl+Z.</p> <p>NOTE: You cannot undo changes that you have applied by clicking OK if you have opened the Manage Network Variables dialog box from within the Control Participant.</p>
Deleting a network variable	<p>Right-click a variable and select Remove to:</p> <ul style="list-style-type: none"> • Display the variable again in the Network Variable Candidates section if it has not been created yet. • Mark the network variable for deletion in the Control Participant project. <p>If the variable is mapped to a communication channel, mapping is removed, page 738.</p> <p>You can also select a variable and press Del.</p>

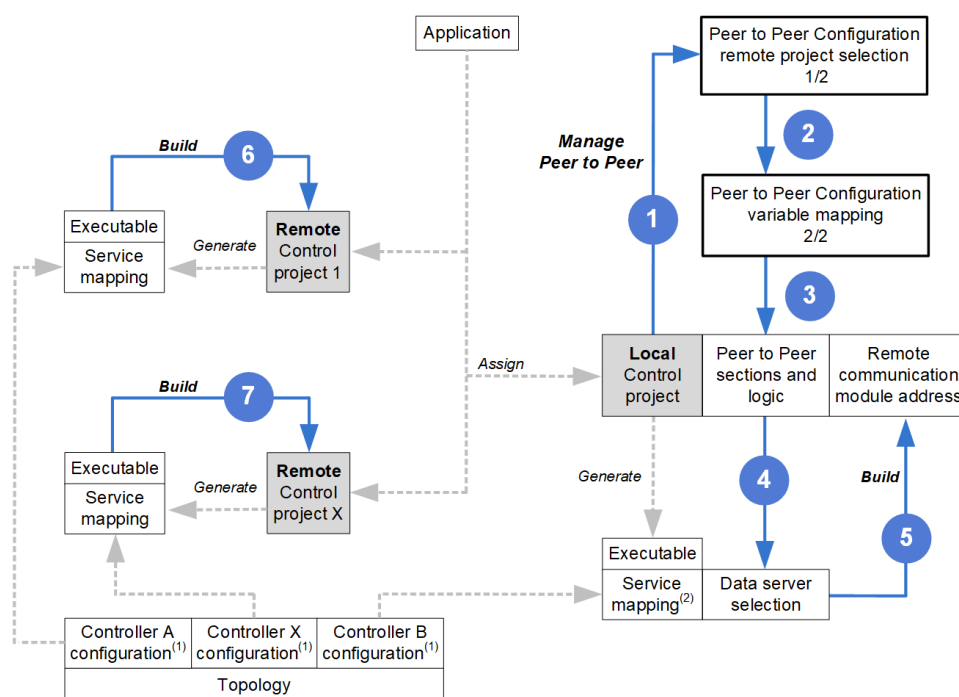
Managing Peer to Peer Communication

Overview

You can implement peer to peer communication by using the specialized *READ_REMOTE* and *WRITE_REMOTE* function blocks of the Control Participant to exchange data between two or more Control projects of the same system. These function blocks as well as the necessary variables and logic are created by EcoStruxure Process Expert in the local Control project.

For more information on the *READ_REMOTE* and *WRITE_REMOTE* function blocks including controller module firmware requirements, refer to the help of the Communication Library of the Control Participant.

The following diagram shows the workflow to configure peer to peer communication between a local Control project (client) and several remote Control projects (servers) in which variables are read and/or written. Refinement of either Control project may be required to perform additional steps. Refer to the next topics for details.



----	System engineering workflow. Completing these steps is a prerequisite. You can start with the local or the remote Control project.
—	Peer to peer communication configuration steps, page 484.
(1)	Using the Controller simulator, page 597 as topological engine is supported.
(2)	Creating the executable in the local Control project and performing the service mapping can also be done after generating the peer to peer communication mapping (step 3).

NOTE:

You can use this method instead of or in addition to using the peer to peer communication templates, page 463 of the Foundation Library.

Peer to Peer Communication Configuration – Remote Project Selection 1/2

Overview

The **Manage Peer to Peer** context menu command of a Control project (**Project Explorer**) opens the **Peer to Peer Communication Configuration** window, which lets you start the peer to peer communication configuration process. The prerequisites, page 484 must be fulfilled to enable the command.

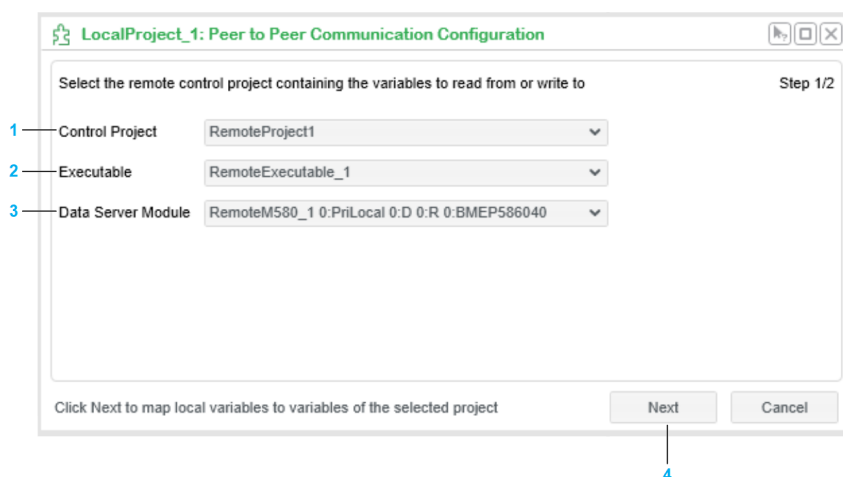
The Control project from which you open this window is considered the *local* project.

Only one user at a time can open this window for the same Control project and while the window is open, the local and the selected remote Control projects are locked.

NOTE: The configuration windows described in this topic do not identify variables that have already been created as *network variables*, page 463 by using the various methods available. They also do not identify network variables that have been mapped by using the **Variable Mapping Editor**, page 738.

Window Description

The following figure shows an example of the window that opens when you select the **Manage Peer to Peer** command for a Control project.



Item	Description
1	Lists the other Control projects of the system, which have at least one executable with existing service mapping. These are the projects (remote) from which you can select variables to read from or write to.
2	Executables of the selected Control project with existing service mapping.
3	Communication modules that exist in the configuration of the controller that is mapped to the selected executable. NOTE: You can also select an executable that is mapped to a workstation/Control Expert service instance (Controller simulator). In this case, <i>NIC_X</i> or a controller module part number appears as data server. NOTE: For peer to peer communication to work, controllers must fulfill the prerequisites, page 484.
4	Opens the variable mapping window (step 2/2).

Configuring Peer to Peer Communication With Controller Simulator Instances

When you select an executable that is mapped to a **Control Expert** service, page 597 of a workstation, the following information appears in the **Data Server Module** text box depending on the port that is used by the Control Expert service:

- Default port 502: *NIC_X* of the workstation.
- Other ports: Part number of the controller module in the configuration of the project file.

EcoStruxure Process Expert creates and/or updates the *IPSimPortAssign.xml* port simulation file in the *Temp* folder at the path *C:\Users\<username>\AppData\Local*.

Ensure that the same path is configured in the **IP Address <-> Port Simulation File** text box of the **Simulator Panel Options** of the Controller simulator.

The file is created the first time you generate peer to peer communication mappings.

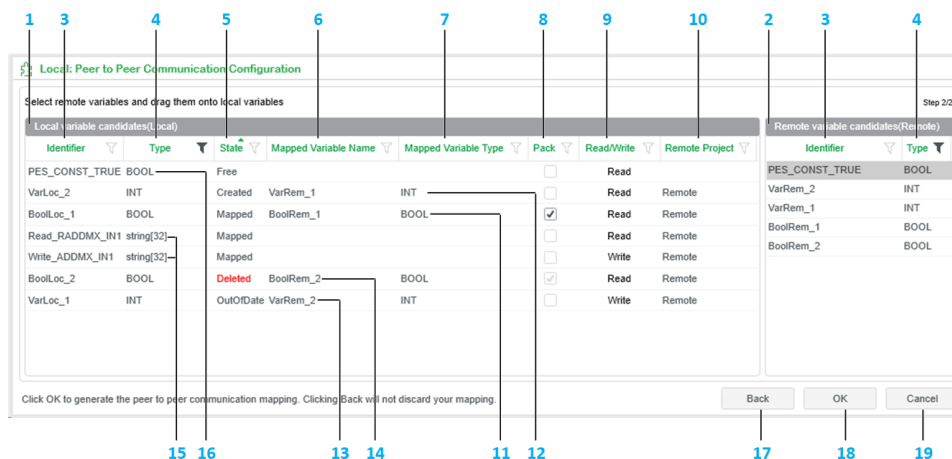
For details on the content and purpose of this file, start the Controller simulator, page 800 and refer to the topic *IP Address and Communication Port Simulation* in the help of the Controller simulator.

NOTE: With Windows Server operating systems, the port simulation file is created in a subfolder of the *Temp* folder. For example, subfolder 2.

Peer to Peer Communication Configuration – Variable Mapping 2/2

Window Description

The following figure shows an example of step 2 of the **Peer to Peer Communication Configuration** window that opens when you click **Next** in step 1. In this example, mappings have been generated a first time then, the window was reopened by using the **Manage Peer to Peer** command and changes were made.



Item	Description
1	<p>Pane showing the variables of the local and the selected remote Control project. It also shows the peer to peer communication mappings that have been generated previously for this pair of Control projects.</p> <p>NOTE:</p> <p>When you reopen the window after generating mappings, the following data is not shown:</p> <ul style="list-style-type: none"> Variables linked to parameters of the READ_REMOTE/WRITE_REMOTE function blocks and that are used for the management of peer to peer communication. Variables of the local Control project that have already been mapped to variables of another remote Control project for peer to peer communication. <p>To view the variables and/or existing mappings, refine the local Control project and open the corresponding sections, page 481.</p>
2	Pane showing the variables of the remote Control project selected in step 1, which you can map to variables of the local project.
3	Name of the variable.
4	<p>Data type of the variable.</p> <p>You can map variables that are of different type but the analysis of the Control project, page 272 during generation of the peer to peer communication mapping or during build of the executable of the local Control project may detect errors.</p> <p>NOTE: By default, a filter is applied to exclude some data types, such as structures. You can clear and/or modify it.</p>
5	<p>State of the local variable with regard to peer to peer communication.</p> <p>Possible values:</p> <ul style="list-style-type: none"> Free: The variable is not yet used for peer to peer communication with any remote project. Created: Variable is already used for peer to peer communication. Mapped: The mapping with the remote variable has been generated in the local Control project.

Item	Description
	<ul style="list-style-type: none"> • OutOfDate: Applies when a mapping with a remote variable was already generated (state = Mapped) and the following action is performed: <ul style="list-style-type: none"> ◦ A mapping with another remote variable, page 487 was performed on the same local variable. ◦ The Pack property was changed. <p>For variables assigned to the ADDMX/ADDM function blocks, page 481, this state can also indicate the following in case mappings were already generated:</p> <ul style="list-style-type: none"> ◦ The executable of the remote control project has been deleted, page 489. ◦ The local Control project, which contains peer to peer communication mapping data related to an existing remote Control project was imported. <p>Once you generate changes, the state changes to Mapped again.</p> • Deleted: The mapping has been unmapped. You cannot reuse the variable until you generate changes, after which the state changes to Free. This state can also identify the following: <ul style="list-style-type: none"> ◦ Refinement was performed in the local and/or remote Control projects, which results in the deletion of an existing mapping. ◦ The instance or the facet generating a mapped variable has been deleted or unassigned. ◦ The mapping (state = Mapped) pertains to a remote Control project, which has been deleted, page 489. In this case, the identifier of the deleted remote Control project is indicated in the Remote Project column and the mapping will be removed once you generate changes.
6	Name of the variable of the remote Control project that is mapped.
7	Type of the variable of the remote Control project that is mapped.
8	<p>Lets you pack variables, page 483 of boolean data type so that you can group into one variable more than a <i>READ_REMOTE</i> or <i>WRITE_REMOTE</i> function block supports.</p> <p>NOTE: Once you change the Pack property for a mapping that is already generated (state = Mapped), you cannot change it anymore unless you generate changes and reopen the window.</p>
9	<p>Direction of the communication.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Read: The value is read from the remote variable and stored in the local variable. • Write: The value stored in the local variable is written to the remote variable. <p>To change the value, right-click a row in the local variables pane and select a direction. You can set a direction when several mapped variables are selected.</p> <p>NOTE: Once the mapping for a variable has been generated, you cannot change the direction, page 488.</p>
10	Identifier of the remote Control project that the mapped variables belongs to.
11	Variables of one or more remote Control projects that have been mapped to local variables and whose value will be read/written (state is Mapped).
12	New mapping that was created after the variable mapping window was reopened (state is Created). This mapping needs to be generated.
13	Existing mapping that was modified after the variable mapping window was reopened by dragging a different remote variable onto it (state is OutOfDate). This mapping needs to be generated for the change to be applied.
14	Existing mapping that was deleted after the variable mapping window was reopened by using the Unmap context menu command (state is Deleted). This mapping needs to be generated for the change to be applied.
15	<p>Variables of the ADDMX/ADDM function blocks that are linked to a <i>READ_REMOTE</i> or <i>WRITE_REMOTE</i> function block. They hold the topological address and IP address of the selected communication module of the remote Control project.</p> <p>The variables are shown and automatically mapped (and cannot be unmapped) after you generate peer to peer communication. Their value is set after you build the executable of the local Control project.</p>
16	<p><i>PES_CONST_TRUE</i> elementary read-only variable (see <i>EcoStruxure Process Expert, Control Participant Services User Guide</i>) that exists in any EcoStruxure Process Expert Control project.</p> <p>NOTE: Mapping onto read-only variables of the local project with direction <i>Read</i> or with read-only variables from the remote project with direction <i>Write</i> is allowed but the analysis of the Control project, page 272 during generation of the peer to peer communication mapping or during build of the executable of the local Control project detects errors.</p>

Item	Description
17	<p>Lets you go back to the Control project selection window where you can select another remote Control project, executable, and/or communication module.</p> <p>Thereafter, when you click Next, you can see already generated mappings and the remote variables that correspond to your new selection.</p> <p>NOTE: Clicking Back does not discard any mappings. However, selecting another remote Control project discards all changes and mappings that have not been generated yet. That is, all mappings with State other than <i>Mapped</i>.</p>
18	<p>Closes the window and generates the peer to peer communication mapping.</p> <p>This creates in the local Control project one or more <i>FBD</i> sections, page 481 containing logic that is used for peer to peer communication.</p>
19	<p>If changes have not been generated yet, opens the Cancel Peer to Peer Communication Configuration dialog box, which lets you perform either action:</p> <ul style="list-style-type: none">• No, closes the configuration window and discard changes that you have not applied yet.• Yes, applies changes before closing the window.• Cancel, reverts to the variable mapping window. <p>Otherwise, closes the window.</p>

Sections and Logic Created for Peer to Peer Communication

Overview

Generating peer to peer communication mapping for the first time or generating changes creates in the local and/or remote Control projects one or more FBD sections.

The sections appear in the **Assignment Editor** but you cannot assign facets to them. To view and edit their logic, refine the local Control project.

This topic describes the sections and their content.

Refer also to the [example scenarios](#), page 492.

Sections for *READ_REMOTE* and *WRITE_REMOTE* Function Blocks

Section characteristics	Description
Name	<p>Default value:</p> <ul style="list-style-type: none"> <i>Read_<Remote project identifier>_P2P_X</i> <i>Write_<Remote project identifier>_P2P_X</i> <p>You can edit the name of the FBD sections in the dialog box that opens when you drag a remote variable onto a local one to create the mapping.</p> <p>Where <i>_X</i> is integer that is incremented if more than one section for a given communication direction is required.</p> <p>The name is discarded and the corresponding section not created if no mapping with this read or write direction has been generated.</p> <p>The next time you drag a remote variable onto a local one, if a section for the <i>READ_REMOTE</i> or <i>WRITE_REMOTE</i> block has not been generated yet in the local Control project, the dialog opens again.</p> <p>For information on applicable naming rules, refer to <i>Variable Names and Section Identifiers</i> in the topic describing naming rules, page 31.</p> <p>NOTE: Once a section has been created with the name that you have entered, it appears as read-only in the dialog box if it opens again.</p>
Location	In the local Control project only.
Count	<p>At least one FBD section for each direction (read/write) per remote Control project of which you have mapped variables.</p> <p>If one section cannot hold the required number of function blocks, additional sections are created.</p>
Content	<ul style="list-style-type: none"> In each section, as many <i>READ_REMOTE</i> or <i>WRITE_REMOTE</i> function blocks as needed to support the number of mapped variables. One <i>ADDMX</i> or <i>ADDM</i> function block for each <i>READ_REMOTE</i> or <i>WRITE_REMOTE</i> function block. The necessary variables and logic. <p>NOTE: Sections and variables are locked, page 452.</p>
Function block configuration	<p>The variables linked to the following parameters of each <i>READ_REMOTE</i> and <i>WRITE_REMOTE</i> function block are initialized with these values:</p> <ul style="list-style-type: none"> <i>CTRL</i>: <ul style="list-style-type: none"> <i>INT [0] (TIMEOUT)</i>: 1000 <i>INT [1]</i>: 0 <i>ENABLE</i>: 1

Section characteristics	Description	
Variable properties	Packed variables (independently of the data type)	Refer to the description of variable properties in sections for mapped variables that are packed.
	Variables of <i>REF_TO</i> data type	Refer to the description of variable properties in sections for mapped variables of <i>REF_TO</i> data type.
	Intermediate variables	<ul style="list-style-type: none"> Name: <ul style="list-style-type: none"> For variables linked to the <i>READ_REMOTE</i> function block: <i>Read_<First letter of the remote Control project>_RVAR_X</i> where <i>_X</i> is an integer incremented for name uniqueness. For variables linked to the <i>WRITE_REMOTE</i> function block: <i>Write__RVAR_X</i> where <i>_X</i> is an integer incremented for name uniqueness (the remote Control project is not identified in the name). Data type: <i>String[128]</i> Value: Name of the variable of the remote Control project that is mapped or, if the mapped variable is packed, name of the variable that is assigned to the input of the function block unpacking the data, page 483 (for example, <i>Pack_L_UPG_P2P_1</i>).
	Peer to peer management variables	Name: Uses the <i>Read_</i> or <i>Write_</i> prefix and an incremental integer suffix <i>_X</i> , which corresponds to the suffix that is used for the section (for example, <i>Read_ENABLE_1</i> or <i>Write_STATUS_1</i>).
	Variables for topological and IP address of remote data server	<p>Variables linked to the function blocks of type <i>ADDMX/ADDM</i>.</p> <ul style="list-style-type: none"> Name: <ul style="list-style-type: none"> For variables in a READ section: <i>Read_<First letter of the remote Control project><Function block type>_INX</i> where <i>_X</i> is an integer incremented for name uniqueness. For variables in a WRITE section: <i>Write_<Function block type>_INX</i> where <i>_X</i> is an integer incremented for name uniqueness (the remote Control project is not identified in the name). Data type: <i>String[32]</i> Value: Topological address and IP address of the selected communication module of the remote Control project. The value is set after you build the executable of the local Control project. Default value: Blank.

Sections for Mapped Variables that are Packed

The following logic is generated when the **Pack** property is selected for at least one boolean variable in step 2 of the **Peer to Peer Communication Configuration** window.

Section characteristics	Description	
Name	In the local Control project	<i>Pack_<Remote project identifier>_P2P_X</i> Where <i>_X</i> is integer that is incremented if more than one section is required.
	In the remote Control project	<i>Pack_<Local project identifier>_P2P_X</i> Where <i>_X</i> is integer that is incremented if more than one section is required.
Count	At least one FBD section per Control project of which you have mapped variables. If one section cannot hold the required number of function blocks, additional sections are created.	
Content	For variables that are read	<p><i>PG_X</i> function blocks of the following types are created:</p> <ul style="list-style-type: none"> In the remote Control project: <i>BIT_TO_WORD</i> to pack variables In the local Control project: <i>WORD_TO_BIT</i> to unpack variables The necessary variables to store the packed data. <p>As many function blocks as needed to support the number of packed variables are created and identified by using the <i>_X</i> integer increment.</p> <p>NOTE: Sections and variables are locked, page 452.</p>
	For variables that are written	<p><i>PG_X</i> function blocks of the following types are created:</p> <ul style="list-style-type: none"> In the local Control project: <i>BIT_TO_WORD</i> to pack variables In the remote Control project: <i>WORD_TO_BIT</i> to unpack variables The necessary variables to store the packed data. <p>As many function blocks as needed to support the number of packed variables are created and identified by using the <i>_X</i> integer increment.</p> <p>NOTE: Sections and variables are locked, page 452.</p>
Variable properties	Variables assigned to the output of the function block packing the data	<p>Name: <i>Pack_<First letter of counterpart project identifier>_PG_P2P_X</i> where <i>_X</i> is an integer incremented for name uniqueness.</p> <p>For additional properties, refine the Control project and open the Data Editor.</p>
	Variables assigned to the input of the function block unpacking the data	<p>Name: <i>Pack_<First letter of counterpart project identifier>_UPG_P2P_X</i> where <i>_X</i> is an integer incremented for name uniqueness.</p> <p>Data type: <i>WORD</i></p> <p>For additional properties, refine the Control project and open the Data Editor.</p>

Sections for Mapped Variables of *REF_TO* Data Type

The following logic is generated when at least one variable of *REF_TO* data type is mapped in step 2 of the **Peer to Peer Communication Configuration** window.

Section characteristics	Description	
Name	In the local Control project	<i>Ref_<Remote project identifier>_P2P_X</i> Where <i>_X</i> is an integer that is incremented if more than one section is required.
	In the remote Control project	<i>Ref_<Local project identifier>_P2P_X</i> Where <i>_X</i> is an integer that is incremented if more than one section is required.
Count	At least one FBD section per communication direction and per Control project of which you have mapped a <i>REF_TO</i> variable. That is, if you have mapped one <i>REF_TO</i> variable to be read and one to be written, two sections are created in the local project and two in the remote project. If one section cannot hold the required number of function blocks, additional sections are created.	
Content	For variables that are read	<p>The following function block instances of type <i>MOVE</i> are created:</p> <ul style="list-style-type: none"> In the remote Control project: Named <i>Ref_<First letters of local project identifier>_P2P_X_Reference_Y</i> to change the variable data type to non <i>REF_TO</i>. In the local Control project: Named <i>Ref_<First letters of remote project identifier>_P2P_X_Reference_Y</i> to revert the variable to <i>REF_TO</i> data type. The necessary variables to store the packed data. <p>As many function blocks as needed to support the number of mapped variables are created and identified by using the <i>_Y</i> integer increment.</p> <p>The <i>_X</i> integer increment corresponds to the section number.</p> <p>NOTE: Sections and variables are locked, page 452.</p>
	For variables that are written	<p>The following function block instances of type <i>MOVE</i> are created:</p> <ul style="list-style-type: none"> In the local Control project: Named <i>Ref_<First letters of remote project identifier>_P2P_X_Reference_Y</i> to change the variable data type to non <i>REF_TO</i>. In the remote Control project: Named <i>Ref_<First letters of local project identifier>_P2P_X_Reference_Y</i> to revert the variable to <i>REF_TO</i> data type. The necessary variables to store the packed data. <p>As many function blocks as needed to support the number of mapped variables are created and identified by using the <i>_Y</i> integer increment.</p> <p>The <i>_X</i> integer increment corresponds to the section number.</p> <p>NOTE: Sections and variables are locked, page 452.</p>
Variable properties	<p>Variables of non <i>REF_TO</i> data type assigned to the input or output of the <i>REF_TO</i> function.</p> <p>Data type: Equivalent to the <i>REF_TO</i> data type (for example, <i>INT</i> for a <i>REF_TO_INT</i> variable).</p> <p>Name: <i>Ref_<First letters of counterpart project identifier>_Ref_P2P_X</i> where <i>_X</i> is an integer incremented for name uniqueness.</p> <p>For additional properties, refine the Control project and open the Data Editor.</p>	

Creating Peer to Peer Communication Mapping

This topic describes the steps to create peer to peer communication mappings and generate the corresponding logic in the respective Control projects.

Prerequisites

Verify that the following prerequisites to generate peer to peer communication mappings are fulfilled:

- Controllers are of the following platform and/or firmware version to support peer to peer communication:
 - M580
Version 3.20 or higher required for the local Control project.
 - M340
Version 3.30 or higher required for the local Control project.

NOTE: BMXPRA0100 modules cannot be used for peer to peer communication because they do not support the data dictionary.

- Quantum (supported only in the remote Control project).

Using the [Controller simulator](#), page 597 as topological engine is supported.

- The data dictionary is enabled for all controllers.
- The variables exist in the local and remote Control Participant projects so that they appear in the peer to peer communication configuration window. That is, the facets creating them are generated and/or variables have been created during refinement.
- The service mapping is done for the executable of the remote Control project so that the **Manage Peer to Peer** command is available.
- The communication modules to be used are configured with an IP address that is unique.

NOTE: For more information on using the *READ_REMOTE* and *WRITE_REMOTE* functions, refer to the help of the Communication Library of the Control Participant.

Creating and Generating the Peer to Peer Communication Mapping

Step	Action
1	<p>In the Project Browser of the Project Explorer, right-click the Control project that is the scanner (local) and select Manage Peer to Peer.</p> <p>Result: The first Peer to Peer Communication Configuration window opens (step 1).</p>
2	<p>Select the Control project (remote) that contains the variables that you want to read from and/or write to, an executable, and a communication module.</p> <p>NOTE: If you have already generated peer to peer mappings and reopen the Peer to Peer Communication Configuration window, a different Control project and executable may be selected by default.</p>
3	<p>Click Next.</p> <p>Result: Step 2 of the Peer to Peer Communication Configuration window opens and after a moment, the local and remote variable panes are populated.</p>
4	<p>In the remote variables pane, select one or more variables and drag them onto a row of the local variables pane.</p> <p>Result: A dialog box opens prompting you to enter an identifier for the sections, page 481 for peer to peer communication that will be created in the local Control project.</p>
5	<p>Enter a meaningful name and click OK.</p> <p>Result: The remote variables are mapped, in the order of selection, to contiguous local variables starting from the row onto which you have dragged them. State of the local variables changes to <i>Created</i>. For boolean variables, Pack, page 483 is selected by default.</p> <p>NOTE:</p> <p>To undo a mapping, page 487, right-click the corresponding row in the local variables pane and select Unmap</p>

Step	Action
	To modify a mapping, page 487, drag a different remote variable onto the mapped local variable.
6	<p>To change the read/write direction, right-click one or more rows and select a new direction.</p> <p>NOTE: Once the mapping for a variable has been generated (State is <i>Mapped</i>), you cannot change the direction, page 488.</p>
7	<p>Once you are done with the mapping configuration, click OK.</p> <p>Result:</p> <ul style="list-style-type: none"> • The configuration window closes. • The necessary sections and logic, page 481 are created in the local Control project. • The build state of both executables is set to Out Of Date (if it was Built). For the executable of the remote Control project because the <i>HMI</i> attribute of mapped variables is set to true. • The outcome of the mapping operation is indicated in the notification panel. <p>NOTE: To map variables of another remote Control project, click Back instead and change your selection in the first window. Any changes and mappings (State other than <i>Mapped</i>) are discarded. To retain your changes and mappings, generate them first, reopen the Peer to Peer Communication Configuration window, and select the other remote Control project.</p>
8	Open the executable of the local Control project in the Service Mapping Editor , page 723 by selecting Manage from its context menu and select the communication module that you want to use for peer to peer communication in the Data Server for Peer to Peer Communication column.
9	<p>Build the executables of the local and the remote Control projects.</p> <p>Result: The address of the selected communication module of the remote controller is set as value in the address variable of the <i>ADDMX/ADDM</i> function blocks of the local Control project.</p>
10	Refine the local Control project and validate the sections that have been created for peer to peer communication.

Modifying and Deleting Peer to Peer Communication Mapping

Overview

This topic describes various scenarios of peer to peer communication mapping modification and their impact on the information that is shown in the **Peer to Peer Communication Configuration** window.

It also describes a limitation, page 491 related to Control project export/import.

Modifying an Existing Peer to Peer Communication Configuration

To modify an existing peer to peer communication configuration, proceed as follows.

Step	Action
1	Right-click the Control project that is the scanner (local) and select Manage Peer to Peer . Result: The Peer to Peer Communication Configuration window opens (step 1).
2	Select the Control project (remote) whose peer to peer mapping you want to modify, the executable, and the communication module that you had initially selected.
3	Click Next . Result: Step 2 of the Peer to Peer Communication Configuration window opens and after a moment, the local and remote variable panes are populated with the existing mappings.
4	Proceed with your changes and click OK . Result: <ul style="list-style-type: none">• The configuration window closes.• The logic, page 481 is created/updated in the local Control project.• The build state of both executables is set to OutOfDate (if it was Built).• The outcome of the mapping operation is indicated in the notification panel.

Deleting or Changing a Variable Mapping

Depending on the state of the mapping, the table describes how to perform various actions in the **Peer to Peer Communication Configuration** window.

Action	Steps	Additional steps when the mapping has already been generated
Deleting the mapping	Right-click the mapping and select Unmap .	State of the unmapped local variable changes to <i>Deleted</i> .
Moving a mapping to another local variable	<ol style="list-style-type: none"> Right-click the mapping and select Unmap. The mapping is undone. Drag the remote variable onto another local variable. 	<p>To complete the process and/or map the variable again, generate the change.</p> <p>Open the window again to perform a new mapping.</p>
Mapping another remote variable to an already mapped local variable	<p>Drag the other remote variable onto the mapped local variable.</p> <p>The existing mapping is undone.</p>	<p>State of the local variable changes to <i>OutOfDate</i>.</p> <p>The existing mapping is updated after you generate the change.</p> <p>NOTE: The new mapping cannot be changed anymore until you generate it.</p>

The following table describes a specific scenario where the local variable is not of *REF_TO* data type and one of the following conditions is satisfied:

- The remote variable that is already mapped to the local variable is of *REF_TO* data type and the remote variable that you are dragging to the left pane is not of *REF_TO* data type.
- The remote variable that is already mapped to the local variable is not of *REF_TO* data type and the remote variable that you are dragging to the left pane is of *REF_TO* data type.

Action	Steps	Additional steps when the mapping has already been generated
Mapping another remote variable to an already mapped local variable.	<ol style="list-style-type: none"> Right-click the mapping and select Unmap. The mapping is undone. Drag the remote variable onto the local variable. 	<p>State of the unmapped local variable changes to <i>Deleted</i>.</p> <p>To complete the unmapping process, generate the change.</p> <p>Open the window again to perform the new mapping.</p>

Changing the Communication Direction of Mapped Variables

Once the mapping of a variable has been generated (**State** is *Mapped* or *OutOfDate*), you cannot change the direction anymore by using the context menu in the **Peer to Peer Communication Configuration** window.

You need to unmap, generate, and redo the mapping.

Deleting the Remote Control Project

When you delete a Control project that is used as *remote* project for peer to peer communication mapping, the corresponding peer to peer communication logic, page 481 is not deleted from the *local* Control project.

By refining the local Control project, you can delete the peer to peer communication function blocks and variables but not the sections.

To delete the corresponding peer to peer communication logic and sections from the local Control project after deleting a remote Control project, proceed as follows.

NOTE: Steps 1 and 2 of this procedure are only required to enable the **Manage Peer to Peer** command from the local Control project.

Step	Action
1	Create a new Control project and an executable. NOTE: Alternatively, you can use any existing Control project other than the local one. If you deleted only the executable of the remote Control project, recreate one.
2	Proceed with the service mapping of the executable to a controller or workstation of the topology.
3	Right-click the local Control project and select Manage Peer to Peer . Result: The first Peer to Peer Communication Configuration window opens.
4	Click Next . Result: The second Peer to Peer Communication Configuration window opens and the mappings that exist with variables of the deleted remote Control project are shown with State Deleted . The identifier of the deleted remote Control project is indicated in the Remote Project column.
5	Click OK . Result: The Peer to Peer Communication Configuration window closes and peer to peer communication variables that correspond to mappings with State Deleted are removed from the local Control project. NOTE: Peer to peer management variables (for example, <i>Write_ENABLE_1</i>) are not deleted.
6	Double-click the Containers node of the local Control project in the Control Project Browser .
7	In the Containers pane of the Assignment Editor , right-click the sections that are related to peer to peer communication mapping with the deleted remote Control project and select Delete .
8	If you had created a Control project and/or executable in step 1, delete it.

Deleting the Executable of the Remote Control Project

When you delete the executable of a Control project that is used as *remote* project for peer to peer communication mapping and no other executable exists, you cannot open the **Peer to Peer Communication Configuration** window anymore.

In order to do so, you need to create an executable and perform the service mapping.

Thereafter, when you reopen the **Peer to Peer Communication Configuration** window and select the same remote Control project, existing mappings have the same **State** but the variables assigned to the ADDMX/ADDMM function blocks, page 478 have the **State OutOfDate**.

Click **OK** to generate the changes.

Deleting the Local Control Project

When you delete a Control project that is used as *local* project for peer to peer communication mapping, the corresponding peer to peer communication logic, page 481 is not deleted from the *remote* Control project. Logic is present if you have packed variables or mapped variables of *REF_TO* data type.

By refining the remote Control project, you can delete the peer to peer communication function blocks and variables but not the sections.

To delete the corresponding peer to peer communication logic and sections from the remote Control project after deleting the local Control project, proceed as follows.

Step	Action
1	Double-click the Containers node of the remote Control project in the Control Project Browser .
2	In the Containers pane of the Assignment Editor , right-click the sections that are related to peer to peer communication mapping with the deleted local Control project and select Delete .

Refining the Local and Remote Control Projects

The following describes refinement scenarios and their impact when you reopen the **Peer to Peer Communication Configuration** window from the same local Control project and select the same remote Control project.

NOTE: Changes that you make when refining sections, page 481 that are created for peer to peer communication are not detected when you do a consistency check, page 460. Therefore, it is a good practice to create, modify, and delete peer to peer communication mappings only by using the **Peer to Peer Communication Configuration** window.

Control project	Action	Impact
Local and/or remote	Deleting an intermediate variable or unlinking a variable from a function block created for peer to peer communication.	State of the corresponding mapping is set to <i>Deleted</i> . Generating changes sets it to <i>Free</i> .
Local	Creating a mapping with a variable from the remote Control project in the section containing the <i>READ_REMOTE</i> or <i>WRITE_REMOTE</i> function blocks.	Variables that you create are shown as local variables in the window. However, the mapping itself is not reflected. If the mapping is configured properly, it is functional. Refinements that you make in sections created for peer to peer communication are not overwritten if you create or modify mappings in the Peer to Peer Communication Configuration window.

Limitation Related to Project Export/Import

When peer to peer communication mappings have already been generated between the local and the remote Control project, exporting either one of these projects or both, reopening the **Peer to Peer Communication Configuration** window for the same projects and creating or modifying mappings and then, importing the project(s) that you have exported may create an inconsistency between the mapping information that is shown in the **Peer to Peer Communication Configuration** window and the mappings that are created and visible when you refine the local project.

To help avoid inconsistencies, proceed as follows when exporting/importing Control projects for which peer to peer communication mappings exist:

- Export both the local and the remote Control projects.
- Import both together.
- Before importing the Control projects, delete or rename the corresponding local and remote Control projects in the **Control Project Browser**.

Managing Peer to Peer Communication Example

Overview

By using a simple example, this topic illustrates which sections and logic are generated in Control projects whose variables are mapped for peer to peer communication by using the **Peer to Peer Communication Configuration** window.

Various scenarios are described by using one local and two remote Control projects.

- Mapping of variables of remote Control project 1:
 - Variables of non-REF_TO data type, not packed, page 493
 - Packed variables of non-REF_TO data type, page 496
 - REF_TO variables, not packed, page 498
 - Packed REF_TO variables, page 501
- Mapping of variables of remote Control project 2, packed, non-REF_TO data type, page 504
- Building the local project, page 507

As a starting point, instances are created in the **Application Browser**, their facets are assigned to the Control projects and generated.

An executable is created for each remote Control project. Each one is mapped to a separate controller.

Remote Control Project Selection

The **Peer to Peer Communication Configuration** window is opened from *Local_Project* and *Remote1_Project* is selected as remote project.

Local_Project: Peer to Peer Communication Configuration

Select the remote control project containing the variables to read from or write to

Control Project: Remote1_Project

Executable: Executable1_Rem1

Data Server Module: M580_Rem1 0:PriLocal 0:D 0:R 0:BMEP586040

Peer to Peer Communication Mapping Window

Step 2 of the window shows the variables of *Remote1_Project* (right) that can be mapped to variables of *Local_Project* (left).

Local_Project: Peer to Peer Communication Configuration Step 2/2

Select remote variables and drag them onto local variables

Identifier	Type	State	Mapped Variable Name	Mapped Variable Type	Pack	Read/Write	Remote Proj
PES_CONST_TRUE	BOOL	Free			<input type="checkbox"/>	Read	
Init1_Local	INT	Free			<input type="checkbox"/>	Read	
Bool1_Local	BOOL	Free			<input type="checkbox"/>	Read	
REF_TO_REAL1_Local	REF_TO REAL	Free			<input type="checkbox"/>	Read	
Bool2_Local	BOOL	Free			<input type="checkbox"/>	Read	

Identifier	Type
PES_CONST_TRUE	BOOL
Init1_Rem1	INT
Bool1_Rem1	BOOL
REF_TO_REAL1_Rem1	REF_TO REAL
Bool2_Rem1	BOOL

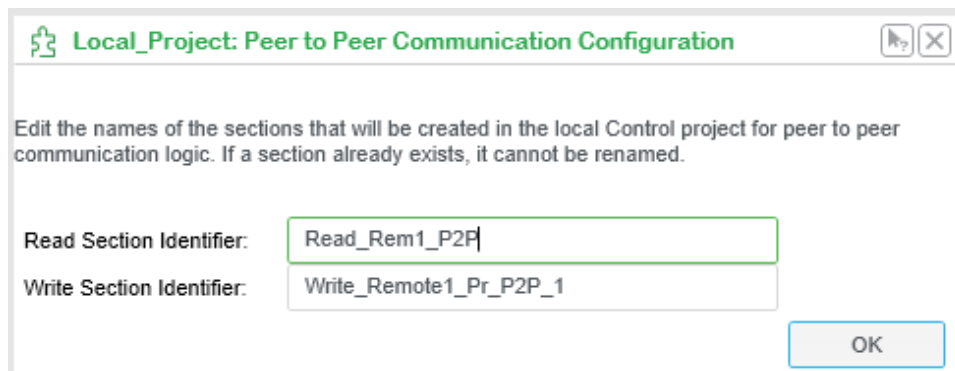
Click OK to generate the peer to peer communication mapping. Clicking Back will not discard your mapping.

Back OK Cancel

Peer to Peer Communication Mapping Creation

In this example, only the name of the section for the *READ_REMOTE* function block was edited when the first mapping was created by dragging variables of *Remote1_Project* to the left pane,.

NOTE: You can edit sections names further by refining the Control projects.



Local_Project: Peer to Peer Communication Configuration

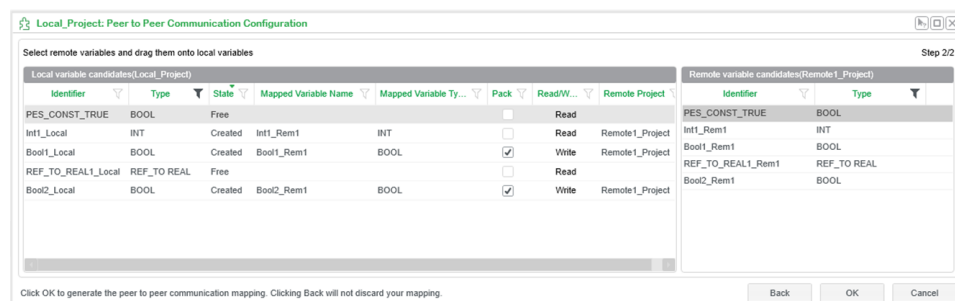
Edit the names of the sections that will be created in the local Control project for peer to peer communication logic. If a section already exists, it cannot be renamed.

Read Section Identifier:

Write Section Identifier:

OK

The following figure shows the mappings that are created.



Local_Project: Peer to Peer Communication Configuration Step 2/2

Select remote variables and drag them onto local variables

Local variable candidates(Local_Project)							Remote variable candidates(Remote1_Project)		
Identifier	Type	State	Mapped Variable Name	Mapped Variable Ty...	Pack	ReadW...	Remote Project	Identifier	Type
PES_CONST_TRUE	BOOL	Free			<input type="checkbox"/>	Read		PES_CONST_TRUE	BOOL
Int1_Local	INT	Created	Int1_Rem1	INT	<input type="checkbox"/>	Read	Remote1_Project	Int1_Rem1	INT
Bool1_Local	BOOL	Created	Bool1_Rem1	BOOL	<input checked="" type="checkbox"/>	Write	Remote1_Project	Bool1_Rem1	BOOL
REF_TO_REAL1_Local	REF_TO_REAL	Free			<input type="checkbox"/>	Read		REF_TO_REAL1_Rem1	REF_TO_REAL
Bool2_Local	BOOL	Created	Bool2_Rem1	BOOL	<input checked="" type="checkbox"/>	Write	Remote1_Project	Bool2_Rem1	BOOL

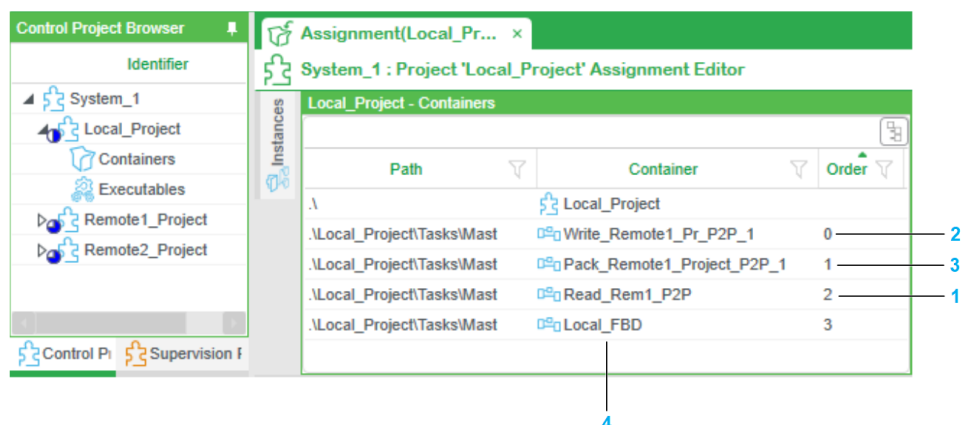
Click OK to generate the peer to peer communication mapping. Clicking Back will not discard your mapping.

Back OK Cancel

Local variable	Remote variable	Data type	Direction	Packed
<i>Int1_Local</i>	<i>Int1_Rem1</i>	<i>INT</i>	Read	N/A
<i>Bool1_Local</i>	<i>Bool1_Rem1</i>	<i>BOOL</i>	Write	Yes
<i>Bool2_Local</i>	<i>Bool2_Rem1</i>	<i>BOOL</i>	Write	Yes

Peer to Peer Communication Mapping Sections in the Local Project

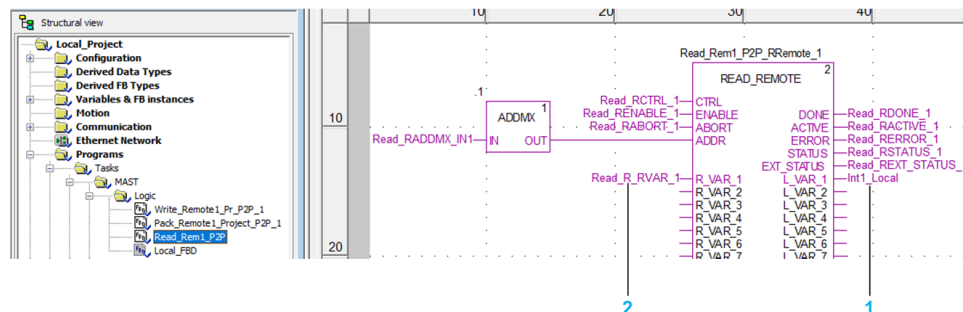
After mappings are generated by clicking **OK** in the communication mapping window, the following FBD sections are shown in *Local_Project*.



Item	Description
1	Peer to peer section containing the <i>READ_REMOTE</i> function block and the necessary variables for the mapping to <i>Int1_Local</i> .
2	Peer to peer section containing the <i>WRITE_REMOTE</i> function block and the necessary variables for the mapping to <i>Bool1_Local</i> and <i>Bool2_Local</i> (packed).
3	Peer to peer section containing the <i>BIT_TO_WORD</i> function block, which packs the values of <i>Bool1_Local</i> and <i>Bool2_Local</i> into one variable (packing).
4	Existing section to which facets of the instances have been assigned.

Peer to Peer Communication Mapping Logic in Local Project – Non-Packed Variables

The following refinement window of *Local_Project* shows the logic that is created for the mapping to *Int1_Local*.



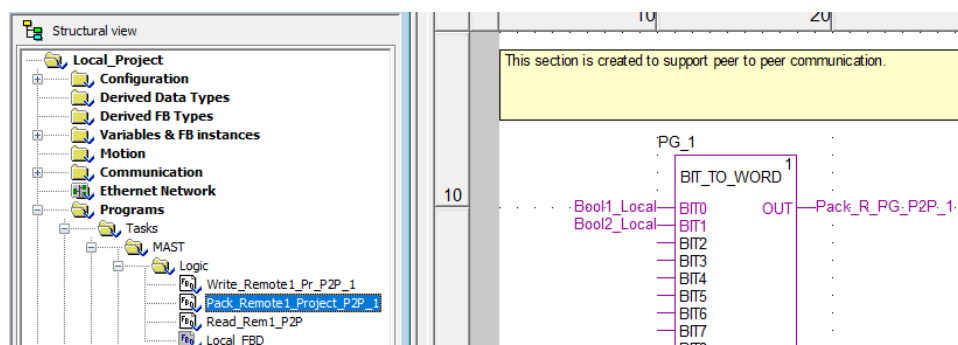
Item	Description
1	Mapped local variable <i>Int1_Local</i> , which receives the value from the intermediate variable.
2	Intermediate variable, which holds the value of the mapped remote variable <i>Int1_Rem1</i> whose value is read.

The following figure shows the values held by the intermediate variable *Read_R_RVAR_1* that is used for peer to peer communication.

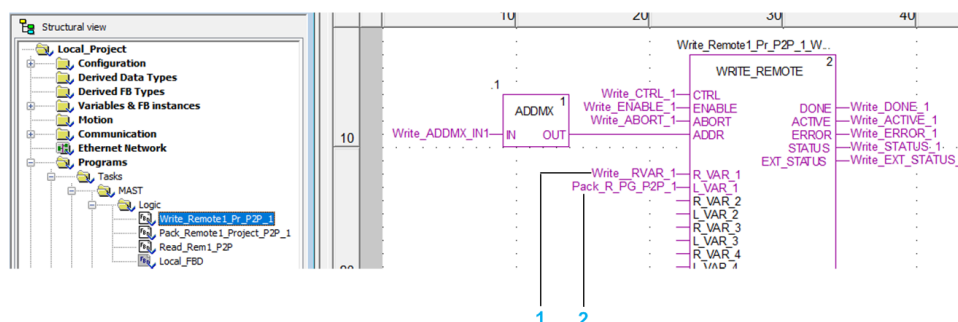
Name	Type	Value	Comment
Read_R_RVAR_1	string[128]	Int1_Rem1	PES generated variable...

Peer to Peer Communication Mapping Logic in Local Project – Packed Variables

The following refinement window of *Local_Project* shows the logic that is created in a dedicated section to transfer the values of *Bool1_Local* and *Bool2_Local* (**Write** direction) into one variable because their **Pack** check box is selected in the mapping window (packing).



The following refinement window of *Local_Project* shows the logic that is created to transfer the values of packed variables *Bool1_Local* and *Bool2_Local* to one intermediate variable.



Item	Description
1	Intermediate variable.
2	Variable that holds the values of <i>Bool1_Local</i> and <i>Bool2_Local</i> to be written to the remote variables.

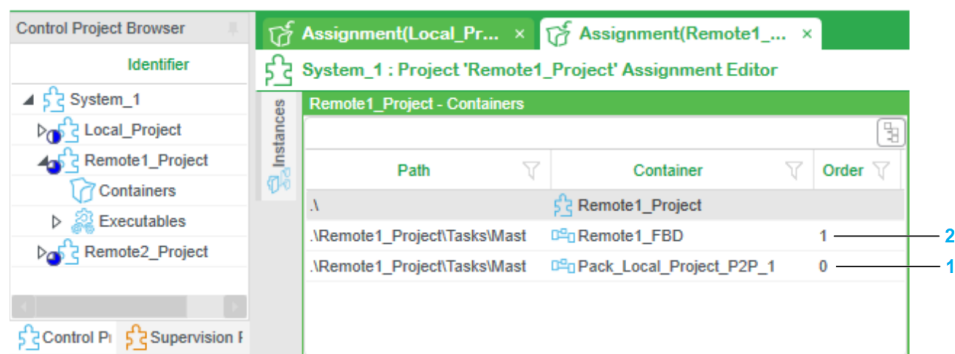
The following figure shows the value held by the intermediate variable *Write_RVAR_1* in *Local_Project*, which is used for peer to peer communication.

The 'Data Editor' window shows the 'Variables' tab. The table below lists the variables and their values.

Name	Type	Value	Comment
Write_RVAR_1	string[128]	Pack_L_UPG_P2P_1	PES generated variable...
Read_R_RVAR_1	string[128]	Int1_Rem1	PES generated variable...
Write_ADDMX_IN1	string[32]		PES generated variable...
Read_RADDMX_IN1	string[32]		PES generated variable...

Peer to Peer Communication Mapping Sections in the Remote Project

After generating the peer to peer communication mapping, the following FBD sections are shown in *Remote1_Project*.



Item	Description
1	Peer to peer section containing the <i>WORD_TO_BIT</i> function block, which transfers the values held by one variable to the two remote variables <i>Bool1_Rem1</i> and <i>Bool2_Rem1</i> (unpacking).
2	Existing section to which facets of the instances have been assigned.

The following refinement window of *Remote1_Project* shows the logic that is created in a dedicated section to transfer the value from one variable to the mapped remote variables.



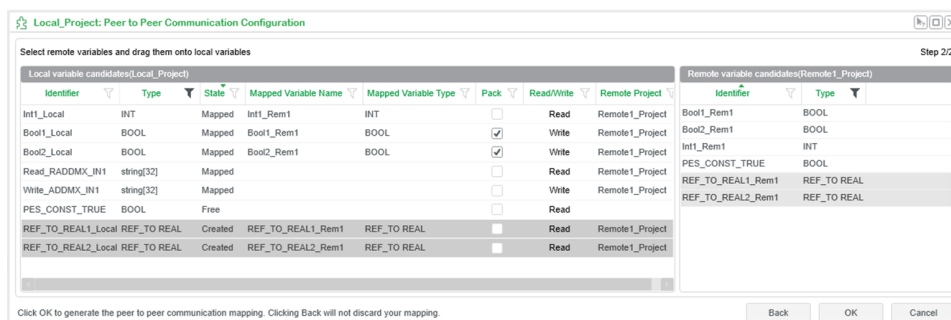
Item	Description
1	Variable holding the values of <i>Bool1_Local</i> and <i>Bool2_Local</i> to be written to the remote variables.
2	Remote variables <i>Bool1_Rem1</i> and <i>Bool2_Rem1</i> .

Peer to Peer Communication Mapping Sections and Logic Local Project – *REF_TO* Variables

Two additional variables, *REF_TO_REAL2_Local* and *REF_TO_REAL2_Rem1*, have been created in the application and their facets assigned and generated in *Local_Project* and *Remote2_Project* respectively.

Then, the **Peer to Peer Communication Configuration** window is reopened from *Local_Project*.

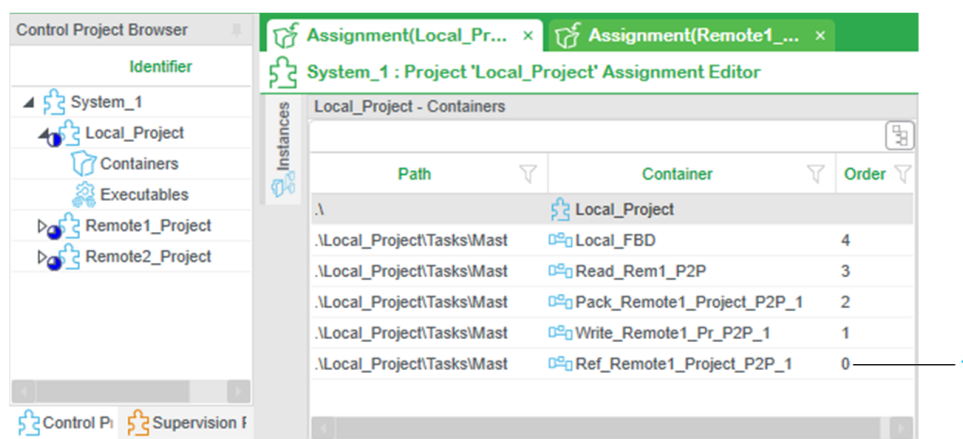
The following figure shows the mappings that have been created.



Local variable	Remote variable	Data type	Direction	Packed
<i>REF_TO_REAL1_Local</i>	<i>REF_TO_REAL1_Rem1</i>	<i>REF_TO_REAL</i>	Read	N/A
<i>REF_TO_REAL2_Local</i>	<i>REF_TO_REAL2_Rem1</i>	<i>REF_TO_REAL</i>	Read	N/A

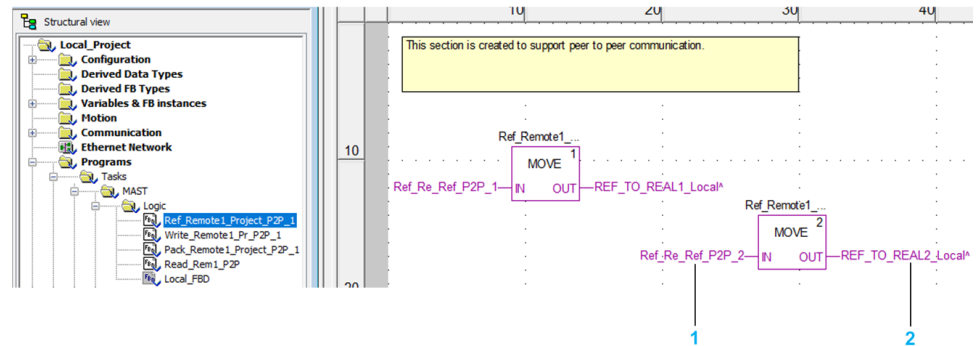
NOTE: In the configuration window, you can see the mappings that have been generated previously and the two variables that hold the topological and IP address of the communication module.

After mappings are generated by clicking **OK** in the communication mapping window, the following FBD sections are shown in *Local_Project*.



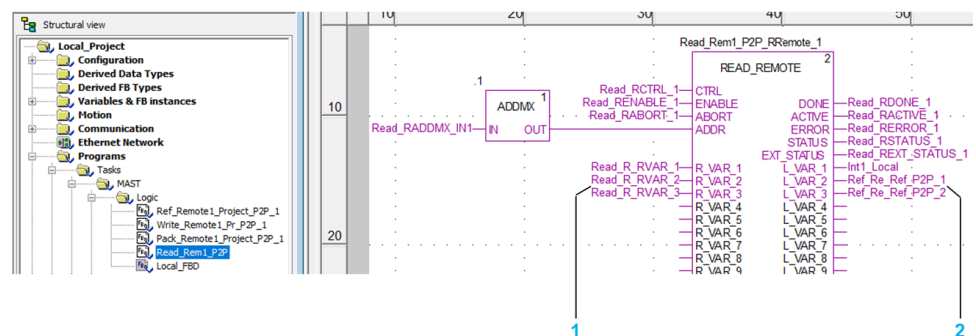
Item	Description
1	New peer to peer section that is created specifically when creating a mapping with local variables of <i>REF_TO</i> data type. NOTE: The section is not created if you map a remote variable of <i>REF_TO</i> data type (for example, <i>REF_TO_REAL</i>) to a local variable of non- <i>REF_TO</i> data type (for example, <i>REAL</i>).

The following refinement window of *Local_Project* shows the logic that is created in a dedicated section to transfer the value from non-REF_TO variables to REF_TO variables.



Item	Description
1	Variable that holds the value of the mapped remote variable and which is of compatible data type (REAL).
2	Mapped local variable of REF_TO_REAL data type, which receives the value that is read from the remote variable.

The following refinement window of *Local_Project* shows the logic that is added to the READ_REMOTE function block to transfer the values held by the intermediate variables to the non-REF_TO variables.



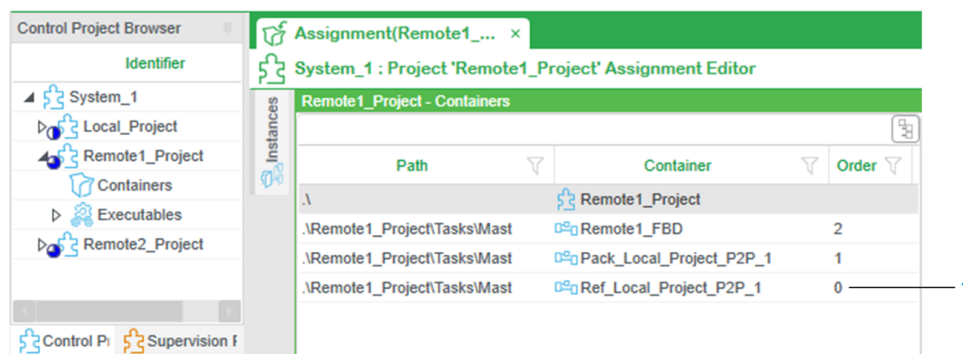
Item	Description
1	Intermediate variables.
2	Variables of non-REF_TO data type holding the value of the remote mapped variables REF_TO_REAL1_Rem1 and REF_TO_REAL2_Rem1.

The following figure shows the value held by the intermediate variables *Read_R_RVAR_2* and *Read_R_RVAR_3* in *Local_Project*, which are used for peer to peer communication.

Name	Type	Value	Comment
Read_R_RVAR_2	string[128]	Ref_Lo_Ref_P2P_1	PES generated variable...
Read_R_RVAR_3	string[128]	Ref_Lo_Ref_P2P_2	PES generated variable...

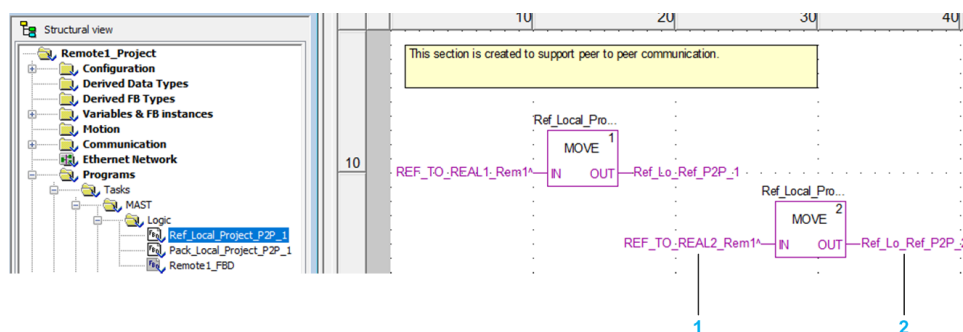
Peer to Peer Communication Mapping Sections and Logic Remote Project – REF_TO Variables

After generating the peer to peer communication mapping, the following FBD sections are shown in *Remote1_Project*.



Item	Description
1	<p>New peer to peer section that is created specifically when creating a mapping with remote variables of <i>REF_TO</i> data type.</p> <p>NOTE: The section is not created if you map a remote variable of non-<i>REF_TO</i> data type (for example, <i>REAL</i>) to a local variable of <i>REF_TO</i> data type (for example, <i>REF_TO_REAL</i>).</p>

The following refinement window of *Remote1_Project* shows the logic that is created in a dedicated section to transfer the value from *REF_TO* variables to non-*REF_TO* variables.



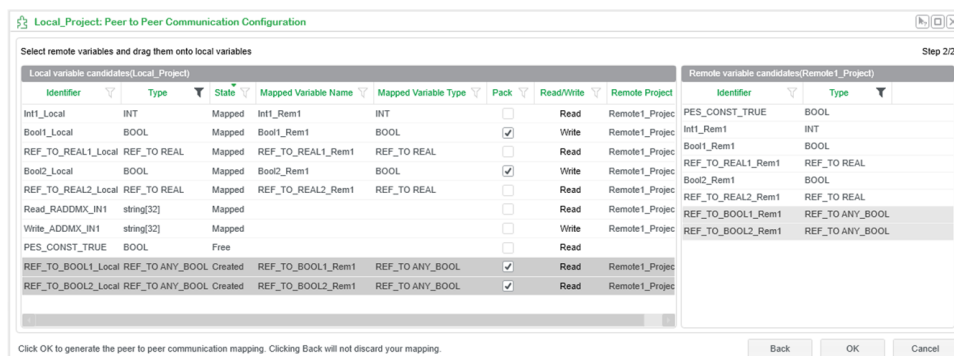
Item	Description
1	Mapped remote variable of <i>REF_TO_REAL</i> data type whose value is read.
2	Variable of compatible non- <i>REF_TO</i> data type (<i>REAL</i>).

Peer to Peer Communication Mapping Sections and Logic – Packed *REF_TO* Variables

New variables of data type *REF_TO_ANY_BOOL* have been created in the application and their facets assigned and generated in *Local_Project* and *Remote1_Project* respectively.

Then, the **Peer to Peer Communication Configuration** window is reopened from *Local_Project*.

The following figure shows the mappings that are created.

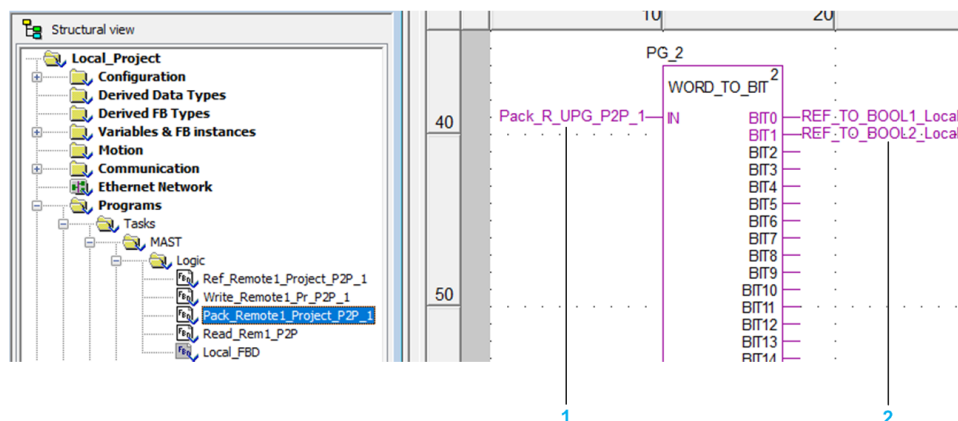


Local variable	Remote variable	Data type	Direction	Packed
<i>REF_TO_BOOL1_Local</i>	<i>REF_TO_BOOL1_Rem1</i>	<i>REF_TO_ANY_BOOL</i>	Read	Yes
<i>REF_TO_BOOL2_Local</i>	<i>REF_TO_BOOL2_Rem1</i>	<i>REF_TO_ANY_BOOL</i>	Read	Yes

After communication mappings are generated by clicking **OK** in the mapping window, no section is added to *Local_Project* nor *Remote1_Project*.

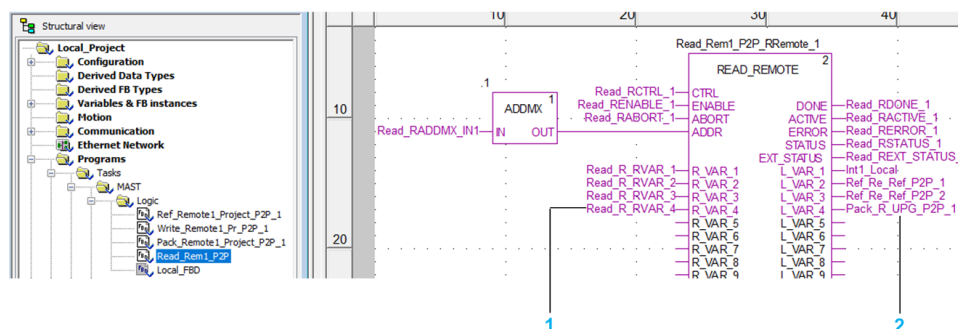
The following refinement window of *Local_Project* shows the logic that is added to the existing *Pack_Remote1_Project_P2P_1* section.

Because the mapped variables are packed, a *WORD_TO_BIT* function block is added to transfer the value from a non-*REF_TO* variable to *REF_TO* variables.



Item	Description
1	Variable of non- <i>REF_TO</i> data type holding the values of the remote mapped variables <i>REF_TO_BOOL1_Rem1</i> and <i>REF_TO_BOOL2_Rem1</i> .
2	Mapped local variables which receive the values that are read from the remote variables.

The following refinement window of *Local_Project* shows the logic that is added to the *READ_REMOTE* function block to transfer the values held by the intermediate variable to one non-*REF_TO* variable.

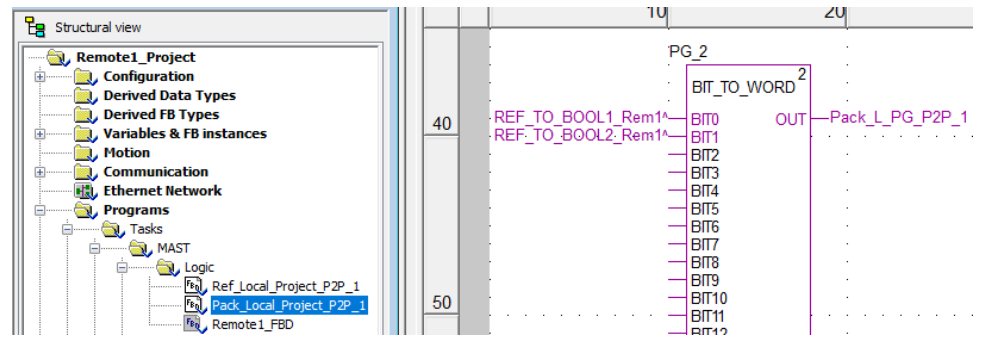


Item	Description
1	Intermediate variable.
	Variable of non- <i>REF_TO</i> data type holding the values of the remote mapped variables <i>REF_TO_BOOL1_Rem1</i> and <i>REF_TO_BOOL2_Rem1</i> .

The following figure shows the value held by the intermediate variable *Read_R_RVAR_4* in *Local_Project*, which is used for peer to peer communication.

Name	Type	Value	Comment
Read_R_RVAR_4	string[128]	Pack_L_PG_P2P_1	PES generated variable...

The following refinement window of *Remote1_Project* shows the logic that is added to existing section *Pack_Local_Project_P2P_1* to transfer the value from the mapped remote *REF_TO* variables to the non-*REF_TO* variable.



Peer to Peer Communication Mapping With Second Remote Control Project

The **Peer to Peer Communication Configuration** window is reopened from *Local_Project* and *Remote2_Project* is selected.

Local_Project: Peer to Peer Communication Configuration

Select the remote control project containing the variables to read from or write to

Control Project: Remote2_Project

Executable: Executable1_Rem2

Data Server Module: M580_Rem2 0:PriLocal 0:D 0:R 0:BMEP586040

The following figure shows the mappings that are created.

Local_Project: Peer to Peer Communication Configuration Step 2/2

Select remote variables and drag them onto local variables

Local variable candidates(Local_Project)							Remote variable candidates(Remote2_Project)	
Identifier	Type	State	Mapped Variable Name	Mapped Variable Type	Pack	Read/Write	Remote Project	
PES_CONST_TRUE	BOOL	Free				Read		
Bool4_Local	BOOL	Created	Bool4_Rem2	BOOL	<input checked="" type="checkbox"/>	Read	Remote2_Project	
Bool3_Local	BOOL	Created	Bool3_Rem2	BOOL	<input checked="" type="checkbox"/>	Read	Remote2_Project	

Click OK to generate the peer to peer communication mapping. Clicking Back will not discard your mapping.

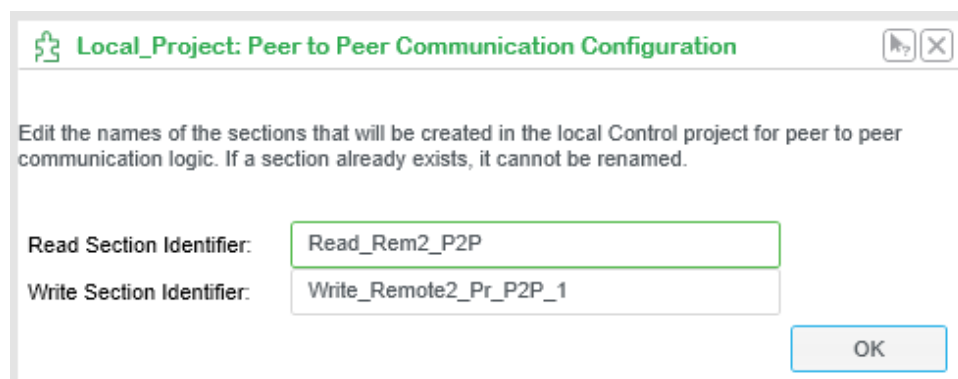
Back OK Cancel

NOTE: In the left pane, you only see the local variables of *Local_Project* that have not been mapped to any remote project yet.

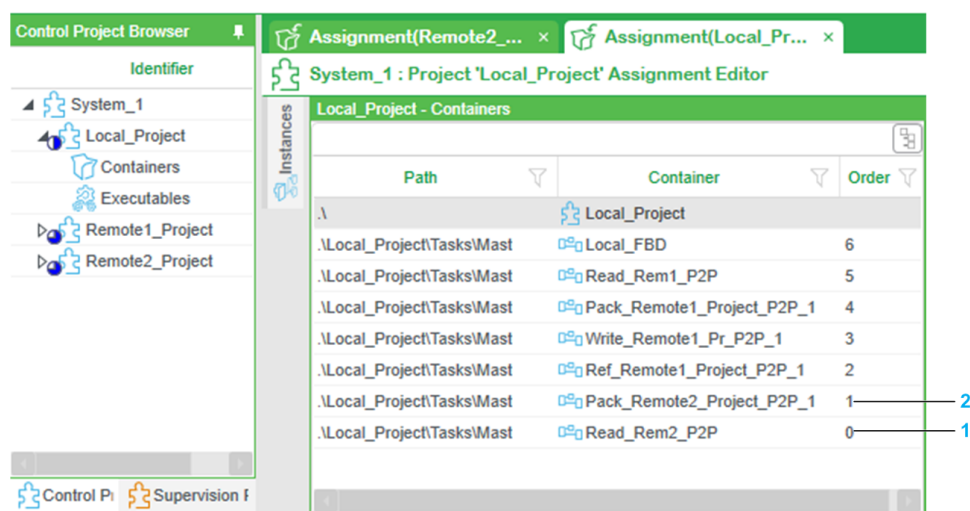
Local variable	Remote variable	Data type	Direction	Packed
<i>BOOL3_Local</i>	<i>BOOL3_Rem2</i>	<i>BOOL</i>	Read	Yes
<i>BOOL4_Local</i>	<i>BOOL4_Rem2</i>	<i>BOOL</i>	Read	Yes

In this example, only the name of the section for the *READ_REMOTE* function block was edited when the first mapping with *Remote2_Project* was created by dragging variables to the left pane. The section for the *WRITE_REMOTE* function block is not created even though its name is shown.

NOTE: You can edit sections names further by refining the Control projects.

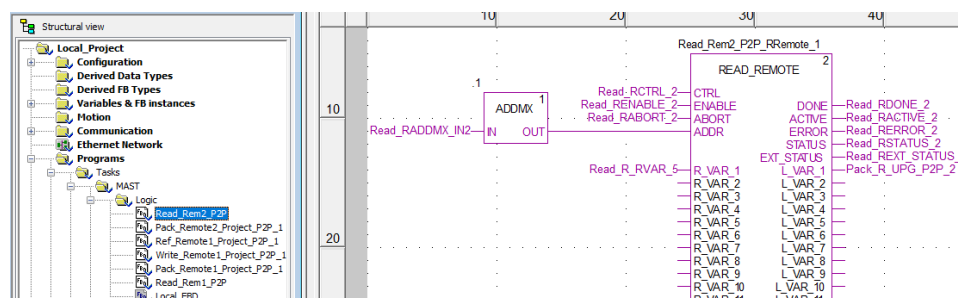


After mappings are generated by clicking **OK** in the communication mapping window, the following FBD sections are shown in *Local_Project*.

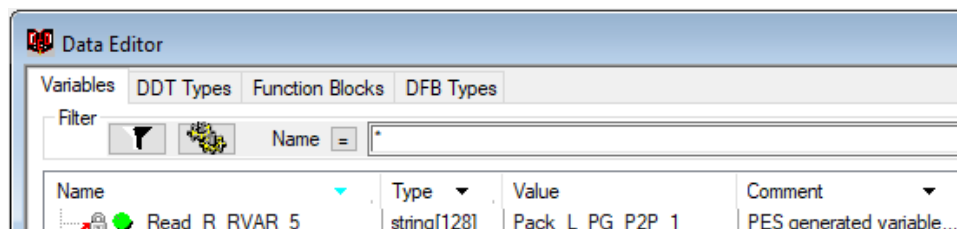


Item	Description
1	New FBD section dedicated to peer to peer communication with variables of <i>Remote2_Project</i> . It contains a <i>READ_REMOTE</i> function block and the necessary variables.
2	New FBD peer to peer section for <i>Remote2_Project</i> containing a <i>WORD_TO_BIT</i> function block, which transfers the values that are held by variable <i>Pack_R_UPG_P2P_2</i> to the two local variables <i>Bool3_Local</i> and <i>Bool4_Local</i> .

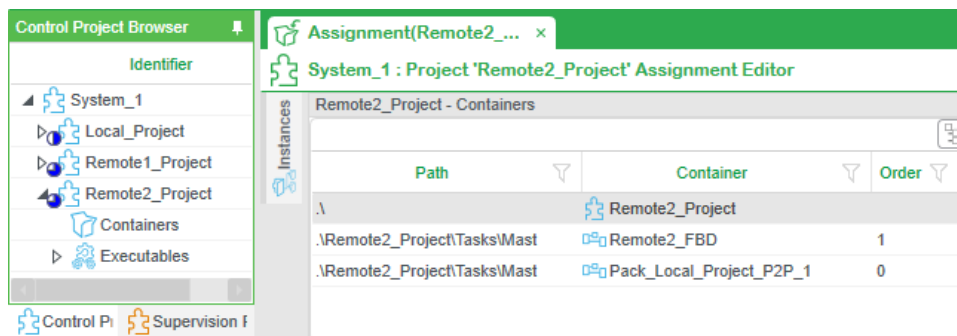
The following refinement window of *Local_Project* shows the logic that is created in the new section to transfer the values held by the intermediate variable to one variable.



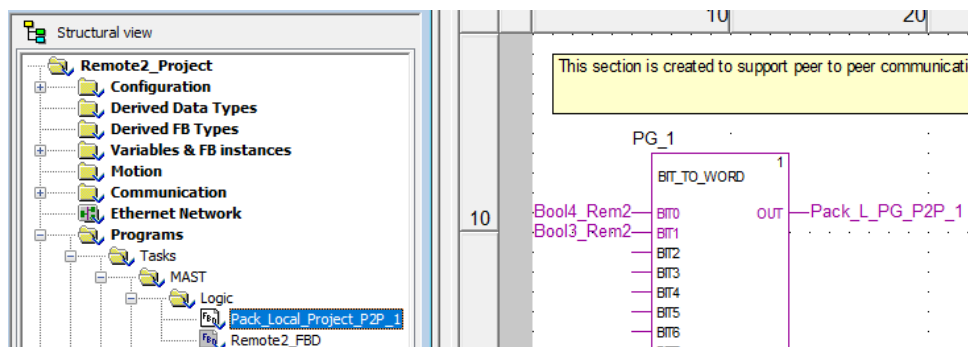
The following figure shows the value held by the intermediate variable *Read_R_RVAR_5* in *Local_Project*, which is used for peer to peer communication.



In *Remote2_Project*, section *Pack_Local_Project_P2P_1* is also added to transfer the values held by the mapped remote variables to one variable.

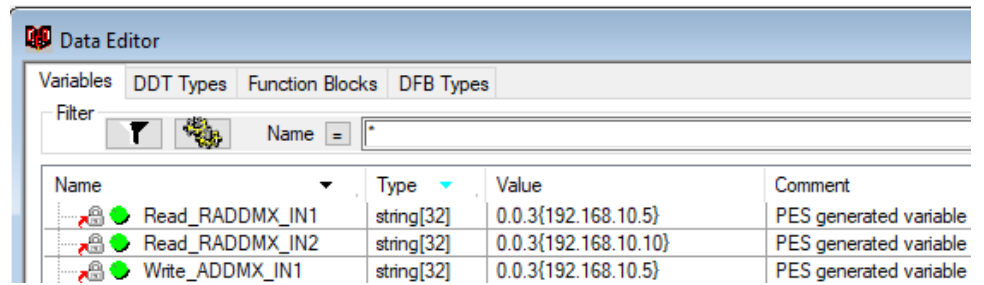





The following refinement window of *Remote2_Project* shows the logic that is added in section *Pack_Local_Project_P2P_1* to transfer the value from the two mapped remote variables *Bool3_Rem2* and *Bool4_Rem2* to one variable.



Building the Executable of *Local_Project*

After building the executable of *Local_Project*, the topological and IP addresses of the communication modules of executables of *Remote1_Project* and *Remote2_Project* appear as value for the corresponding variables.



Data Editor				
Variables				
Filter				
Name = *				
Name		Type	Value	Comment
Read_RADDMX_IN1		string[32]	0.0.3{192.168.10.5}	PES generated variable
Read_RADDMX_IN2		string[32]	0.0.3{192.168.10.10}	PES generated variable
Write_ADDMX_IN1		string[32]	0.0.3{192.168.10.5}	PES generated variable

Supervision Project Refinement Stage

Refining the Supervision Project

Overview

The refinement process of Supervision Participant projects allows you to:

- View and modify Participant project data by using the **Refine** command. Limitations apply to the modification of project data that is managed by EcoStruxure Process Expert.
- Add animated graphics to pages, page 358 of the project, using the **Edit** command.
- Edit Participant project settings by using the **Advanced Settings** command.
- Add included projects and attach user files.

For each Supervision Participant project, you can use only one of the aforementioned commands at a time. However, you can edit several Supervision pages of a project at the same time.

NOTE: While you refine the Supervision Participant project or edit advanced settings, commands such as **Generate** or **Delete** are not available for pages or containers.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Supervision Projects Created in Plant SCADA Standalone

If you have created a Supervision project by using Plant SCADA opened in standalone mode (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*), when you refine any Supervision project on the same computer by using the engineering client, you can see the Supervision project that you have created outside the EcoStruxure Process Expert environment.

The same applies when you edit a Supervision page.

NOTE: Changes to externally-created Supervision projects by using the engineering client are retained and have no impact on projects managed by EcoStruxure Process Expert. Do not modify an externally-created Supervision project by using the engineering client. Work only on the Supervision project managed by EcoStruxure Process Expert, which is the active project.

Refining the Supervision Project

To refine the Supervision project, proceed as follows.



Step	Action
1	Verify that AVEVA Plant SCADA is not open outside EcoStruxure Process Expert.
2	In the Project Explorer , right-click the Supervision project you want to refine and select Refine . Result: The Supervision Participant opens extracted, page 118 and maximized.
3	Proceed with the refinement. For more information on the refinement process and restrictions that apply, refer to the topic describing the refinement of the Supervision project (see <i>EcoStruxure Process Expert, Supervision Participant Services, User Guide</i>).

Step	Action
4	Close the Participant window. Result: The Save Refinement dialog box opens.
5	Click the appropriate button in the dialog box. Result: <ul style="list-style-type: none"> • Yes: Closes the Supervision Participant window and keeps only changes that you have saved in the Supervision Participant. • No: Closes the Supervision Participant window and discards all changes, including those that you have saved in the Supervision Participant. • Cancel: Closes the Save Refinement dialog box and keeps the Supervision Participant window open.

Editing Supervision Project Pages


When you add an animated graphic to a page, EcoStruxure Process Expert adds the required included project to the Supervision Participant project if it is not yet present. You can [download included projects, page 791](#) as soon as they are added.

To edit a Supervision page, proceed as follows.

Step	Action
1	Verify that AVEVA Plant SCADA is not open outside EcoStruxure Process Expert.
2	In the Containers pane of the Assignment Editor , right-click the page, page 358 that you want to edit and select Edit . Result: The Supervision opens extracted, page 118 and maximized with the Instances browser pane next to it.
3	Edit the page as required. For more information on adding animated graphics, refer to Editing Supervision Pages (see EcoStruxure Process Expert, Supervision Participant Services, User Guide) .
4	Click either save button in the toolbar of the Participant window:  to save changes only in the active window.  to save changes in all open windows. NOTE: If you have made changes in several pages, clicking the Save All button completes the save operation faster. Result: The facets that encapsulate the animated graphics that you have added to a page appear in the Assignments pane when you select the page. The Assignment and Generation statuses of these facets are Assigned and Generated respectively. If you have deleted an animated graphic that had already been added to a page, the corresponding facet is removed from the Assignments pane.
5	Close the Supervision Participant window. NOTE: If you have unsaved changes and click the close button on the outer frame of the Participant window, the Save Refinement dialog box opens where you have the following choices: <ul style="list-style-type: none"> • Yes: Saves changes you made to the page and closes the Participant window. • No: Discards changes and closes the Participant window. • Cancel: Does not save changes and leaves the Participant window open.

Editing Advanced Supervision Project Settings

To edit advanced project settings, proceed as follows.

Step	Action
1	Verify that AVEVA Plant SCADA is not open outside EcoStruxure Process Expert.
2	In the Project Explorer , right-click the Supervision project and select Advanced Settings . Result: The Supervision Participant opens extracted, page 118 and maximized.
3	Proceed with changes to the advanced project settings. For more information, refer to Editing Advanced Project Settings (see <i>EcoStruxure Process Expert, Supervision Participant Services, User Guide</i>).
4	Click the save button in the toolbar of the Participant window to save changes. 
5	Close the Supervision Participant window. NOTE: If you have unsaved changes and you click the close button in the Participant window, the Save Advanced Settings dialog box opens where you have the following choices: <ul style="list-style-type: none"> • Yes: Saves changes you made to the advanced settings and closes the Participant window. • No: Discards changes and closes the Participant window. • Cancel: Does not save changes and leaves the Participant window open.

NOTE: You need to deploy, page 843 the Supervision project to apply changes to advanced project settings.

Adding Included Projects

When you add an animated graphic, page 509 to a Supervision page, EcoStruxure Process Expert automatically adds the required included project to the Supervision Participant project.

You can add additional included projects. However, you cannot add an included project if an included project with the same name was already added to the Participant project.

To add additional included projects to a Supervision Participant project, proceed as follows.

Step	Action
1	In the Project Explorer , right-click the Supervision project to which you want to add an included project and select Include Projects . Result: The Include Projects window opens.
2	Click Add . Result: The Open dialog box opens.

Step	Action
3	<p>Browse to the location where the included project file (.ctz) is stored, select it, and click Open.</p> <p>Result:</p> <ul style="list-style-type: none"> If an included project with the same name already exists in the Participant project, a notification is displayed: <ul style="list-style-type: none"> Click OK to acknowledge. Close the Include Projects window. If an included project with the same name already exists in the Global Constituents folder of the contents repository, a notification is displayed: <ul style="list-style-type: none"> Click OK to add the included project that exists in the contents repository. Click Cancel to cancel the process without adding the included project. If no included project with the same name already exists: <ul style="list-style-type: none"> The selected file is stored in the contents repository. The selected file appears in the window. <p>NOTE: You can deploy, page 841 included project that you add.</p>
4	Close the Include Projects window.

NOTE: Adding an included project that contains a configuration file (*citect.ini*) of an earlier version of the Supervision Participant may require that you upgrade the configuration file (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).

NOTE: To remove an included project from the Supervision project, select the file in the **Include Projects** window and click **Remove**. After you acknowledge the command, the file is removed and deleted from the contents repository.

Attaching External Files

To add external files to a Supervision project, proceed as follows.

Step	Action
1	<p>In the Project Explorer, right-click the Supervision project to which you want to add files and select Attach User Files.</p> <p>Result: The Attach User Files window opens.</p>
2	<p>Click Add.</p> <p>Result: The Open dialog box opens.</p>
3	<p>Browse to the location where the file is stored, select it, and click Open.</p> <p>Result:</p> <ul style="list-style-type: none"> The selected file is stored in the contents repository. The selected file appears in the window. <p>NOTE: Files that you attach to the Supervision project are deployed.</p>
4	Close the Attach User Files window.

NOTE: To remove a file from the Supervision project, select the file and click **Remove**. The file is removed and deleted from the **contents repository**. If user files had already been deployed, they are not removed from the location they were deployed to when you re-deploy the Supervision project, page 844.

Configuration Stage

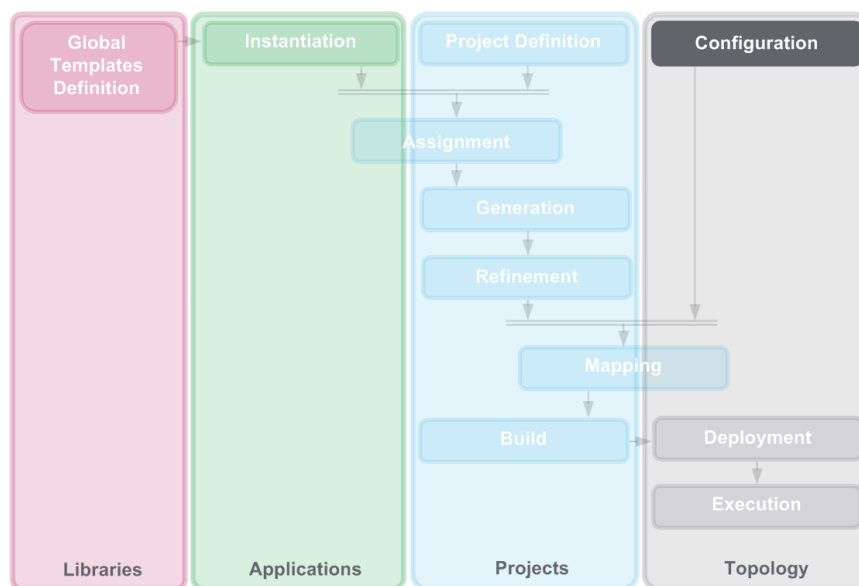
What's in This Chapter

Topology Explorer Description	513
SYSTEM PROJECT Pane	525
Topology Explorer Views	536
Creating the Topology of the System	562
Representation of Entities and Ports	578
Configuring Topological Entities	584
Configuring Fieldbuses	620
Managing the Topology	622
Managing Logical Networks and IP Addresses	625
Using Device Type Managers	629
Implementing Communication With Non-EPE Managed Devices and Controllers	638
Exporting and Importing Topological Entities (SBK)	647
Exporting and Importing I/O Devices to/from CSV Format	660

Overview

This chapter describes how to model the entire topology of the system, which consists of the hardware and software infrastructure.

The following figure shows the position of the **Configuration** stage within the system engineering life cycle.



Refer to the Configuration stage, page 61 for a description of the purpose of this stage.

Topology Explorer Description

Topology Explorer

Overview

The topology consists of topological entities, [page 43](#), which use topological templates and model the hardware of the system. You can organize these entities by using a folder structure that you can create in the **SYSTEM PROJECT** pane.

In addition, you can create a graphical representation of these entities and their network connections by using [physical views, page 536](#).

Opening the Topology Explorer

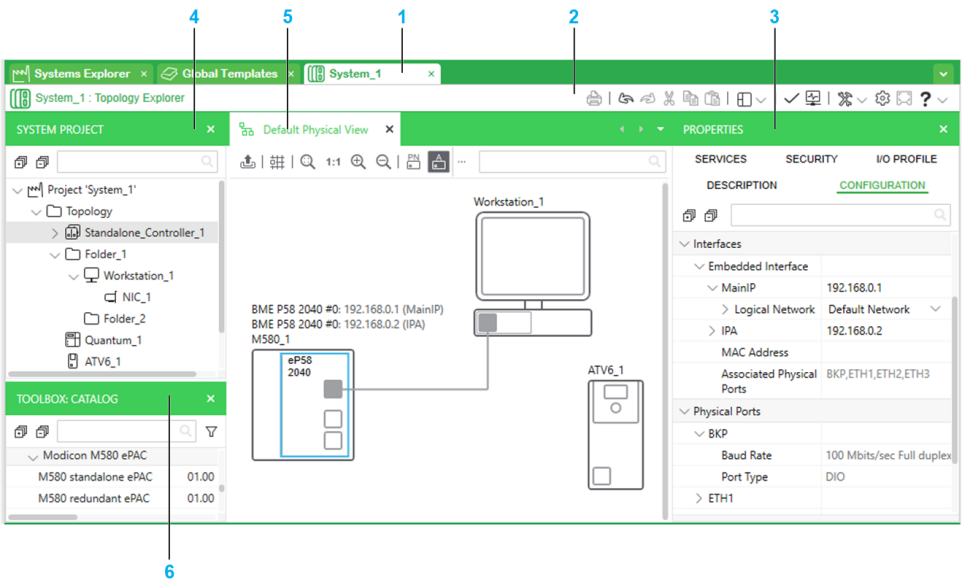
To open the **Topology Explorer**, right-click the system root folder in the **Systems Explorer** or one of the domain explorers (for example, the **Application Explorer**) and select **Open Topology**.

NOTE: You can open a maximum of four **Topology Explorer** instances simultaneously per engineering client.

NOTE: If the **Topology Explorer** shows no content, ensure that on the system sever computer, the *SE System Manager ControlExpert.Topology 15.3* service that is required by the **Topology Explorer** is running; otherwise, start it. Then, log out of the engineering client and log in again, [page 72](#).

Graphical User Interface

The following figure shows an example of the **Topology Explorer** window.



Item	Description
1	Tab of the Topology Explorer window.
2	Toolbar of the Topology Explorer .
3	PROPERTIES pane, page 673 that lets you view and edit the properties of the entity, page 565 that is selected.
4	SYSTEM PROJECT pane, page 525 showing entities (tree view), physical views, and logical networks, page 576 of the system topology, page 562.
5	Physical view, page 536 showing a graphical representation, page 578 of entities that were added to it.
6	TOOLBOX pane, page 516 showing content related to the view, page 536 or pane that is active.

Topology Explorer Toolbar Description



Item	Description	
1	Lets you print the contents of the active window to file (for example, PDF). The button is enabled only for certain views, such as physical views or rack views.	
2	Opens the Views menu, which lets you open various views, page 536 and panes of the Topology Explorer . If a view or pane is already open, it becomes the active one.	
3	Analyzes the consistency of the topology, page 1090.	
4	Activates/deactivates system monitoring, page 1107. The button is highlighted when system monitoring is active.	
5	Opens the Tools menu, which contains the following commands and submenus:	
	Update DTM Catalog	Lets you update the local DTM Hardware Catalog with the DTMs, page 629 that you installed by using an executable or the EDS/GSD installation tool. It also updates the DTM Hardware Catalog of the Control Participant. A restart of the system server is required to complete the update of the Hardware Catalog .
	Manage DTMs	Opens the MANAGE DTMs window, page 633, which contains the following tabs: <ul style="list-style-type: none"> IN USE: Displays information from the DTM Audit Tool, which compares the DTMs that are used by devices of the topology with those available in the local Hardware Catalog. UPDATES: Displays a list of DTMs that are used by devices of the topology and for which a later version of the DTM is available in the local Hardware Catalog. Otherwise, the tab is hidden. You can update, page 636 the DTM of one or several devices with this later version.
	Manage EDS Library	Contains commands that let you manage the EDS files, page 629 that are used to create DTMs for EtherNet/IP devices. A restart of the system server is required to complete the update of the Hardware Catalog .
	Manage GSD Library	Contains commands that let you add and remove GSD files, page 629 that are used to create DTMs for PROFIBUS devices. Additional commands let you export the GSD files that are installed on the local computer to a Library file and import such a file. A restart of the system server is required to complete the update of the Hardware Catalog .
6	Opens the User Preferences window, page 524.	
7	Switches the active view to full screen mode. The functionality is disabled if the active view is a floating, independent window.	

Catalog of the Topology

Overview

The entities that you can add to the topology by using the **Topology Explorer** appear in the **TOOLBOX** pane, which displays the catalog of equipment. They are grouped in several product types.

Each product type contains one or more product ranges.

NOTE: Other equipment that is supported by EcoStruxure Process Expert but does not appear in the **TOOLBOX** (for example, modules of an M580 controller) can be added to the topology by configuring entities with the Control Participant.

Catalog Content

Product type	Product range	Description
Modicon PAC Controllers	Modicon M580 ePAC	<p>Models a configurable standalone and redundant local main rack configuration. Each one consists of:</p> <ul style="list-style-type: none"> An Ethernet backplane X80. A Modicon M580 controller module. <p>NOTE: Safety controllers also contain a coprocessor module.</p> <ul style="list-style-type: none"> A power supply module. <p>The M580 redundant controller is composed of a pair of local racks with roles A and B.</p>
	Modicon M340 PAC	<p>Models a configurable standalone local main rack configuration consisting of:</p> <ul style="list-style-type: none"> An X80 Ethernet backplane. A Modicon M340 controller module. A power supply module.
	Modicon Legacy Controller	<p>Models a configurable standalone local main rack configuration consisting of:</p> <ul style="list-style-type: none"> An XBP backplane. A Modicon Quantum controller module.
Automation Devices	Altivar Variable Speed Drives	<p>Models families of configurable Altivar Process ATV600 and ATV900 drives.</p> <p>The corresponding device type manager (DTM), page 629 must be installed on the local computer to configure the device.</p> <p>NOTE: To configure devices of this product range, use the Communication Mapping View, page 544 and the I/O Mapping Editor, page 547 of the Topology Explorer .</p>
	EPE Managed Devices	<p>Models the following entities, which you can configure by following the EcoStruxure Process Expert system engineering lifecycle:</p> <ul style="list-style-type: none"> I/O devices, page 603: The topological entities let you manage communication, page 935 between the device and the controller by using Modbus TCP (I/O scanning and explicit) or EtherNet/IP (explicit) messaging depending on the template that you select. Advantys islands: Lets you configure, page 599 in the Control Participant the segments, communicator, modules, and devices connected through the CANopen bus of the island. PROFIBUS remote masters (PRMs): Lets you configure, page 601 in the Control Participant the decentralized devices (DPs) connected to the PRM by using device type managers (DTMs) and generic station description (GSD) files.
	Generic Devices	<p>Models configurable devices that communicate by Modbus TCP or EtherNet/IP protocol.</p> <p>For both device types, you can select a DTM, page 629 to configure the device.</p> <p>NOTE: To configure devices of this product range, use the Communication Mapping View, page 544 and the I/O Mapping Editor, page 547 of the Topology Explorer .</p>
	TeSys Motor Controllers	<p>Models several references of configurable TeSys T motor controllers that communicate by using Ethernet-based protocols. You can select a reference, page 565 based on the DTM, page 629 that is installed on the local computer.</p> <p>It also contains a configurable TeSys island.</p> <p>NOTE: To configure devices of this product range, use the Communication Mapping View, page 544 and the I/O Mapping Editor, page 547 of the Topology Explorer .</p>

Product type	Product range	Description
Network Infrastructure Devices	ConneXium Switches and Hubs	Models various references of ConneXium devices that communicate by using Ethernet-based protocols.
	ConneXium Firewalls	
	Generic Switches	Models various references of generic, managed, and unmanaged switches with or without routing capability. You can configure and use them to model your hardware.
	Hirschmann Switches and Hubs	Models various references of RED25 customizable entry-level redundancy switches for M580 applications.
	Modicon Switches	Models various references of Modicon managed, unmanaged, and extended switches.
Network Fieldbus Devices	Gateway	Models various references of communication gateways (for example, a PROFIBUS remote master (PRM)) enabling you to connect Modbus TCP and other fieldbus networks. NOTE: To configure devices of this product range, use the Fieldbus Editor , page 553 of the Topology Explorer .
Workstations and Servers	Generic	Models computers of the control room on which SCADA systems or the controller simulator are running. They are connected to the control and/or operation network and belong to level 2 of process control systems as defined by the ISA-95.00.01 standard.

Scope of the Topology Explorer

Overview

When you configure of a controller, certain configuration parameters, page 522 and commands are unavailable in the Control Participant. You must configure the corresponding attributes and use the corresponding commands in the **Topology Explorer** instead.

General Actions

The following table indicates which tool must be used to perform a given action.

Action	Topology Explorer	Control Participant
Configuration - Offline		
Creating controllers - all platforms	Yes	No
Configuring M580 controller modules (replace, configuration other than communication, such as I/O objects).	No	Yes
Actions on modules (add, move, replace, delete, and configuration other than communication).	No	Yes
Actions on a rack (add, move, replace, delete)	No	Yes
Configuration - Online		
All actions	No	Yes
Data		
Derived data types	No	Yes
Ethernet Networks (M580 platform)		
Creating and managing networks, assigning IP addresses	Yes	No
Communication Networks (platforms other than M580)		
Creation, assigning IP addresses	No	Yes

I/O Devices and Master Modules

The following table indicates which tool must be used to create and configure I/O devices and configure master modules of Modicon M580 controllers.

Action	Topology Explorer	Control Participant
EtherNet/IP/Modbus TCP I/O Devices		
Creating devices	Yes	No
Network configuration of devices	Yes	No
Configuring parameters of DTMs	Yes (inside the DTM, page 629)	No
Creating DTMs from EDS and GSD files	Yes	No
Updating the local DTM catalog	Yes	No
Master Module Network Configuration		
M580 controller modules	Yes	No
Communication modules (Other than those listed below)	Yes	No
BMECXM0100	Yes (IP and DHCP configuration)	Yes (security, services, Ethernet I/O, CANopen configuration)
BMENOP0300	Yes	Yes (by using the IEC 61850 configuration Tool)
BMENOR0200	No	Yes (I/O objects)
BMENOR2200	Yes	Yes (communication channels and services)
BMENOS0300	Yes	No
BMENUA0100	Yes	Yes (time stamping for version .2)
BMXNGD0100 BMXNOM0200 BMXEIA0100	No	Yes
HART analog input/output modules	Yes (network and fieldbus)	Yes (analog module)
PROFIBUS remote master (PRM)	Yes	Yes (for EPE Managed Devices)
BMXPRA0100 (Additional actions to be performed in EcoStruxure Control Expert Classic)	Yes	Yes

Action	Topology Explorer	Control Participant
PMEPXM0100	Yes (IP configuration, services, I/O profile, and DTM, page 629)	Yes (by using the ProSoft Configurator for Modicon Tool)
Implicit Messaging (I/O scanning)		
Communication and I/O mapping	Yes	No
Local slaves	Yes	No

The following table indicates which tool must be used to configure communication parameters of Modicon M340 and Quantum controllers.

Action	Topology Explorer	Control Participant
Ethernet Devices Network Configuration		
Communication module network configuration	No	Yes
I/O scanning	Yes	Yes

Downloads to Controllers or Devices and Uploads From Controllers

The following table indicates which tool must be used to perform a given action.

Action	Topology Explorer	Control Participant
Download to Controllers or Devices		
Transferring the entire project to controllers	Yes	No
Transferring project changes to controllers	Yes	No
Downloading the configuration to one or more I/O devices	Yes	No
Upload From Controllers		
Uploading the program from M580 controllers.	No (only supported in EcoStruxure Control Expert Classic)	
Uploading the program from controllers other than M580 platform	No	No

Control Participant Parameter Correspondence

Overview

When you configure an M580 controller by using the Control Participant, certain configuration parameters are unavailable. You must configure the corresponding attribute, page 673 in the **Topology Explorer** instead.

This topic lists the parameters of the Control Participant that are disabled and that have a different name in the **Topology Explorer**.

List of Parameters

Module/Service	Parameter name and/or location in the Control Participant	Attribute name and/or location in the PROPERTIES of the Topology Explorer
Controller module	ServicePort\Service Port and Service Port Mode	Configuration\Physical Ports\ETH1 = Enabled and Service Port Mode
	BMECRA**** IP address configuration	Configuration\Interfaces\Embedded Interface when you select the BMECRA**** adapter module
BMENOC03•1	DTM\Channel Properties\Switch	Configuration\Physical Ports
	DTM\Services\Service Port	Configuration\Physical Ports\ETH1
	DTM\Services\Network Time Service	Services\NTP State
BMENOC0321	DTM\Services\IP Forwarding	Configuration\Physical Ports\BKP= Enabled and Interface = Enabled
	DTM\Services\IP Forwarding\Control Network	Configuration\Interfaces\Embedded Interface\MainIP
	DTM\Services\IP Forwarding\Fieldbus Network	Configuration\Interfaces\BKP_Interface\IP Address
	DTM\Services\IP Forwarding\Extended Network	Configuration\Physical Ports\BKP= Enabled and \ETH1\Interface= Enabled and Interfaces\ETH1_Interface\IP Address
BMENUA0100	IPConfig\Control Port\IPv4	Configuration\Interfaces\Embedded Interface\IPv4 A
	IPConfig\Backplane Port\IPv4	Configuration\Interfaces\BKP_Interface\IPA
BMENUA0100	IPConfig\Fast Sampling Rate	Services\OPC UA\Sampling Rate
	IPConfig\OPCUA TCP Listening Port	Services\OPC UA\OPCUA TCP Listening Port
Module or DTM \Security	Enforce Security	Security\Security Level = Enforced
	Unlock Security	Security\Security Level = No
	Access Control = Enabled	Security\Services\Access Control = Enabled and click Edit
	Pre-Shared Key	Security\Secure Protocol = IPsec
DHCP	Services\Address Server	Services\DHCP Client\DHCP Server Name and DHCP VIEW
	Device Name (including Tens and Ones parameters)	Services\DHCP Client\Identifier
	Project Settings\General\Configuration \Device name prefix for DHCP Address Server	Services\DHCP Server\DHCP Prefix
DTMs	Advanced Configuration (to open the DTM. May vary by device.)	Device context menu and select Open . For devices managing fieldbuses, edit the fieldbus first.
	DTM context menu\Device menu\Additional functions	Device context menu, select Open , and Additional Functions button. For devices managing fieldbuses, edit the fieldbus first.

Module/Service	Parameter name and/or location in the Control Participant	Attribute name and/or location in the PROPERTIES of the Topology Explorer
I/O scanning	DTM\Device List\Device X\Properties\Number	Add device I/O profile to scanner in Communication Mapping view and select active I/O profile and Device Number
	DTM\Device List\Device X\Properties\Active Configuration	Add device I/O profile to scanner in Communication Mapping view
	DTM\Device List\Device X\Properties\I/O Structure Name\Structure Name	I/O Profile\Profile Name X\Type Name
	DTM\Device List\Device X\Properties\I/O Structure Name\Variable Name	I/O Profile\Profile Name X\Instance Name
Item mapping	DTM\Device List\Device X\Properties\Items Management\Import Mode = Automatic/Manual	I/O Profile\Profile Name X\Mapping = Automatic/Manual
	DTM\Device List\Device X\Properties\Items Management\Reimport Items	I/O Profile\Profile Name X and change Mapping from Manual to Automatic
Scanner\local slaves	DTM\EtherNet/IP Local Slaves\Local Slave X \Active Configuration (adapter project)	I/O Profile\Select DTM Reference and add local slave. When the local slave is deleted, Active Configuration is set to <i>FALSE</i> in the controller configuration project.
	DTM\EtherNet/IP Local Slaves\Local Slave X \Items (scanner and adapter projects)	I/O Profile\Select DTM Reference and add local slave and Local Slave X\Mapping = Manual and click Edit (for adapter only, page 645)
Syslog server	Project Settings\General\PLC diagnostics \PLC Event Logging	Services\Syslog Server\Enabled

User Preferences

Overview

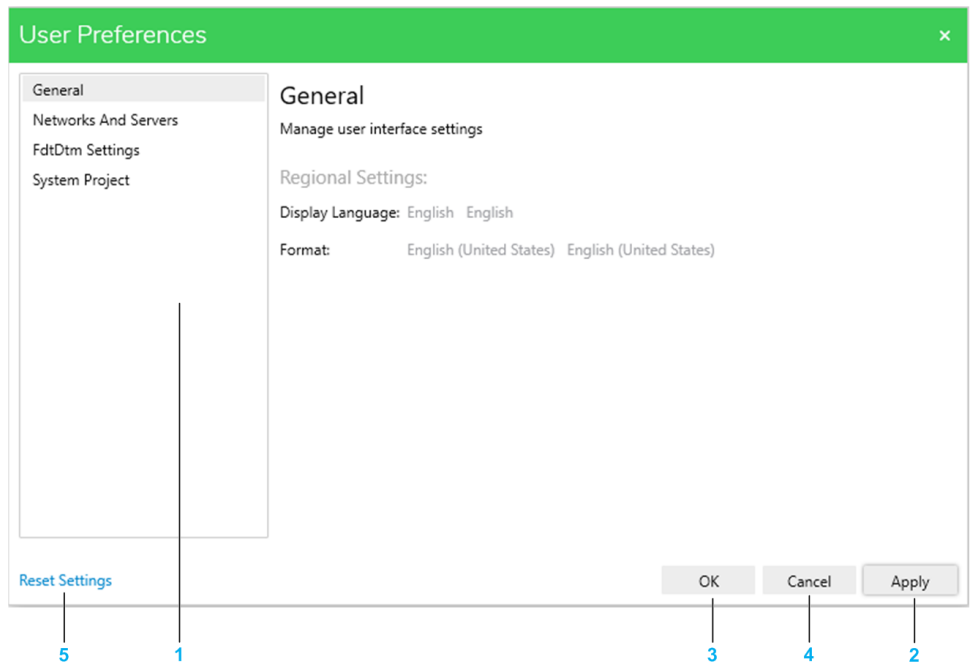
The **User Preferences** window lets you view and configure various settings of the topology.

You can open the **User Preferences** window by clicking the gear icon button on the **Topology Explorer** toolbar.



Description

The following figure shows the **User Preferences** window.



Item	Description	
1	General section	Lets you view the following settings: <ul style="list-style-type: none">• Display Language: Language used for the user interface.• Format: Name of the language culture. The setting defines the formatting of dates, time, currency, and numbers. Both settings are followed by the name in English.
	Networks And Servers section	The section is not used in this version of EcoStruxure Process Expert.
	FdtDtm Settings section	Lets you configure FdtDTM settings, page 629.
	System Project section	Lets you manage the Web Diagnostic/Data Storage password, page 588 of M580 controllers of the topology.
2	Applies your changes and keeps User Preferences open.	
3	Applies your changes and closes User Preferences .	
4	Discards your changes and closes User Preferences .	
5	Reverts the settings of all the sections, except System Project , to their default values after you confirm the command.	

SYSTEM PROJECT Pane

SYSTEM PROJECT Pane Description

Overview

The **SYSTEM PROJECT** pane lets you view, create, organize, and/or interact with the following objects of the topology:

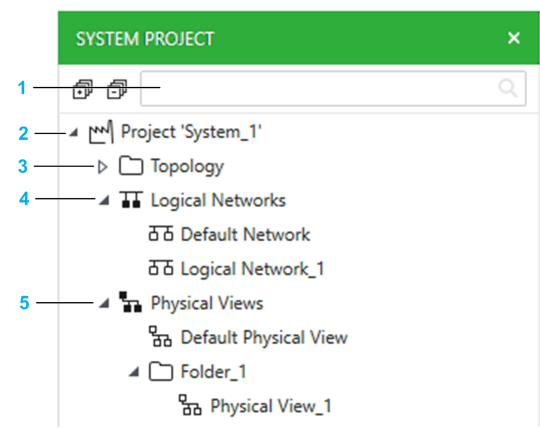
- The structure of the topology and the entities that it contains.
- Logical networks.
- Physical views.

You can view and configure properties, page 673 of each object by selecting the object and perform other actions by opening its context menu.

You can reopen the pane by using the **Views** menu in the toolbar, page 515.

SYSTEM PROJECT Pane General Description

The following figure shows an example of the **SYSTEM PROJECT** pane and the nodes that it contains.



Item	Description
1	Search field. Enter one or more terms separated by a space and the pane displays objects with names that contain any of these terms. Objects are displayed with their parent structure.
2	System root node, page 135.
3	Topology root node, page 527.
4	Logical Networks root node, page 532.
5	Physical Views root node, page 534.

System Root Node Actions

The following table describes the commands of the context menu of the system root node.

Command	Description
Open Logical Network View	Opens the LOGICAL NETWORK VIEW , which lets you view and edit communication attributes of topological entities.
Open DHCP View	Opens the DHCP VIEW , which lets you view and modify the DHCP client/server relation of topological entities that feature the DHCP service.
Open Access Control View	Opens the ACCESS CONTROL VIEW , which lets you view and edit which topological entities are authorized to communicate with other entities that manage security attributes. It also lets you manage their communication protocols.
Open With	Opens a submenu containing a list of views that you can open for the topology.
Lock	Locks the topology so that other users cannot modify it. Changes that you make are visible to others. Other users working on the same topology are informed about the locked/unlocked status in the notification panel.
Unlock	Unlocks the topology so that other users can modify it. Only the user who locked the topology can unlock it.

SYSTEM PROJECT Pane - Topology Node

Overview

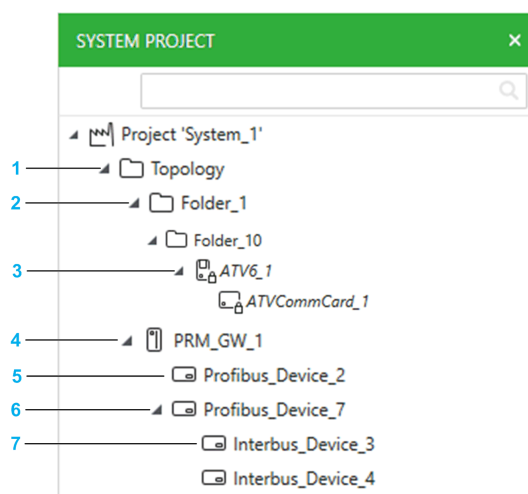
The **Topology** node shows the entities of the topology of the system. Their location depends on how you create the entities, page 565.

The pane lets you do the following:

- View and configure **properties**, page 673 of each entity by opening its context menu and/or using the **PROPERTIES** pane.
- Organize the topology by creating a folder structure. You can change the order in which folders and entities are displayed by dragging an object to its new position within the node, the same folder, or to a different folder.
- Export and import entities, page 647.

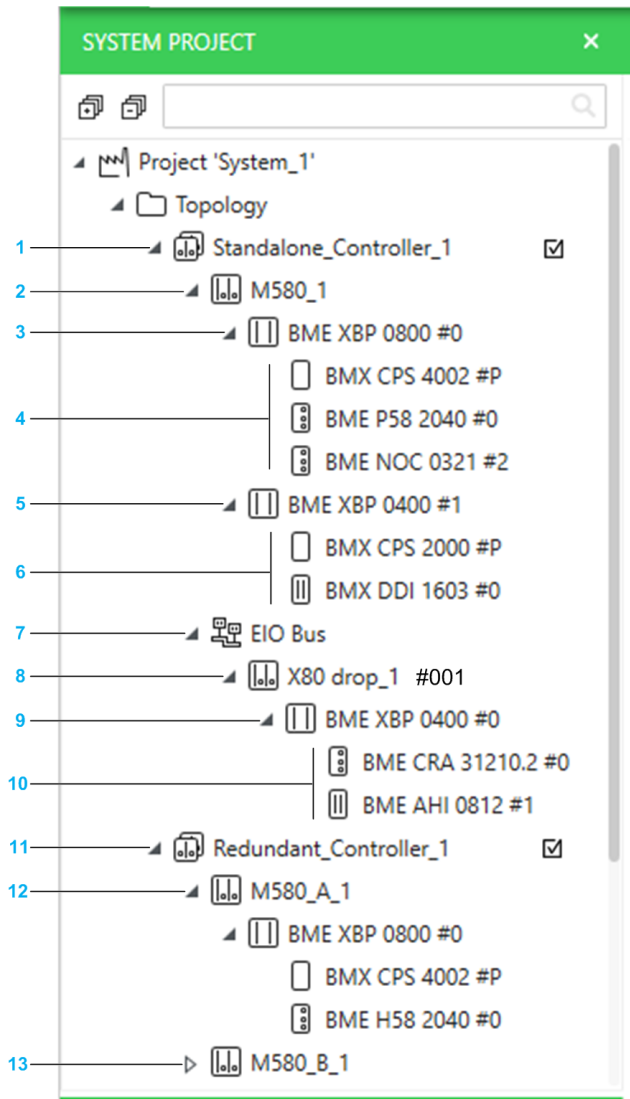
Topology Node Description

The following figure shows a partial view of the **SYSTEM PROJECT** pane with examples of various objects that the **Topology** node can contains.



Item	Description
1	Topology node. By default, each entity that you add from the catalog to a physical view is assigned to it. Right-click it to open a context menu, page 531.
2	User-generated folder.
3	Entity with a child element (In this example, a device of the Altivar Variable Speed Drives product range with a communication card) assigned to a user-generated folder. The entity and child elements are shown with a lock and their name appears in italic when the entity is locked, page 530.
4	PROFIBUS remote master (PRM) of the Gateway product range.
5	PROFIBUS secondary devices.
6	PROFIBUS-to-INTERBUS gateway.
7	Secondary devices of the PROFIBUS-to-INTERBUS gateway.

The following figure shows a partial view of the **SYSTEM PROJECT** pane with examples of M580 controllers.

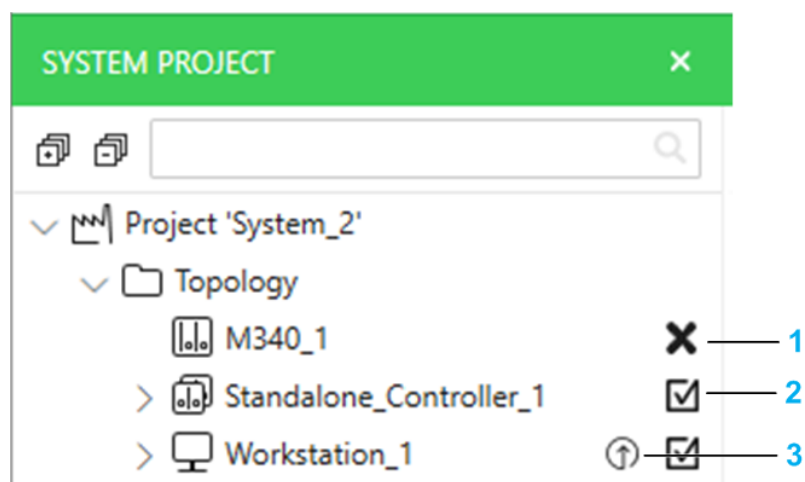


Item	Description
1	<p>Standalone controller node.</p> <p>The configuration changes you make to the hardware of the controller are reflected in this pane.</p> <p>Various icons indicate the mapping and online refinement state, page 529 of the entity.</p> <p>NOTE: For non-M580 controllers, only the node is shown.</p>
2	<p>Local rack.</p> <p>Racks and modules are shown with a lock and their name appears in italic when the controller is locked, page 530.</p>
3	<p>Backplane of the main local rack.</p> <p>The name of a backplane is followed by its address in the rack.</p>
4	<p>Modules of the main local rack.</p> <p>The name of a module is followed by its position (address) on the backplane.</p>
5	Backplane of the extended local rack.
6	Modules of the extended local rack.
7	EIO bus that is shown only if you add a remote rack.
8	<p>Remote rack (drop).</p> <p>The name of the drop is followed by the values of the rotary switches of its BMECRA.... adapter module.</p> <p>NOTE: The address of the drop on the bus is indicated by the Device Number attribute, page 710 of its BMECRA.... adapter module, which is shown when you select the adapter in the Communication Mapping View.</p>
9	Backplane of the main remote rack.
10	Modules of the main remote rack.
11	Redundant controller node.
12	<p>Local rack with role A.</p> <p>You can change the hardware configuration by configuring either local rack. Changes are reflected in both racks.</p>
13	Local rack with role B.

Topological Entity Description

Service mapping and online refinement indicators appear on controllers and workstations on which a Control Expert service is created.

The following figure shows a partial view of the **SYSTEM PROJECT** pane with several topological entities featuring mapping and online refinement indicators.



Item	Description
1	<p>Service mapping, page 723 indicator.</p> <p>The entity is not mapped when a cross is displayed.</p> <p>A tooltip indicates the mapping status.</p>
2	<p>The entity is mapped when a selected check box is displayed.</p>
3	<p>Online refinement indicator.</p> <p>The icon indicates that you made changes online to the Control Participant project that is deployed to the controller or controller simulator (workstation). These changes must be applied to the associated logical Control Participant project and controller configuration to make them consistent with the project that is deployed.</p> <p>The icon is removed once you successfully update the associated logical Control project, page 897 and the configuration with the changes that EcoStruxure Process Expert detected in the following sections of the Update Project dialog box:</p> <ul style="list-style-type: none"> • Derived Data & FB Types • Variables & FB Instances • Sections • Hardware <p>If you update the project only partially with changes of these categories, the icon remains visible.</p> <p>A tooltip indicates the status.</p> <p>NOTE: If you select the changes of the aforementioned categories, whether you select or not changes pertaining to the Animation Tables and/or Operator Screens categories, the refinement indicator is still removed. If you only made changes pertaining to the Animation Tables and/or Operator Screens categories, you must select them to remove the indicator.</p> <p>NOTE: When a workstation has several Control Expert services, the icon is displayed as soon as online changes are made for either one. The tooltip indicates the identifier of the Control projects that the icon applies to.</p>

Entities Locked for Editing

An entity and its child elements are shown with a lock and their name appears in italic when they are locked because you opened an associated object in an editor or Participant. For example, you are configuring a controller, editing the fieldbus of a gateway device, or have opened a DTM.

Once the editor/Participant is closed, the node is unlocked and updated (if applicable).

Selecting Multiple Folders and/or Entities

You can select a combination of folders and entities, including controllers that appear in the **Topology** folder structure. However, nodes need to be collapsed.

Moving Entities to Another Topological Folder

You can change the location of entities inside the **Topology** folder structure by selecting one or more entities and dragging them onto another existing topological folder.

For controllers and other entities with child elements (such as gateways), you cannot move child elements individually (for example, the communication module of a rack or a PROFIBUS secondary device).

Topology Node Actions

The following table describes the commands of the context menu of the **Topology** node.

Command	Description	
New Folder	Creates a new folder, page 563 with properties set to default values.	
Export	Opens a submenu with the following commands:	
	Topology , page 647	Opens the Export window, which allows you to select the topological entities that you want to export as a topology export file (.sbk). You can select entities of the folder and its subfolders as well as the folder hierarchy.
	Devices , page 660	Opens the Export window, which allows you to export the devices of the topology and their network connections as a topology export file (.csv). The command is available only from the Topology root folder.
Import	Opens a submenu with the following commands:	
	Topology , page 651	Opens the Import window to select an export file (.sbk). In a second step, you can select which entities to import in the topology.
	Devices , page 667	Opens the Import Devices window to select a device export file (.csv). In a second step, you can select which devices to import in the topology. The command is available only from the Topology root folder.

Topology Folder Actions

Refer to the description of commands in the topic describing how to organize the topology, page 563 in the **SYSTEM PROJECT** pane.

SYSTEM PROJECT Pane - Logical Networks Node

Overview

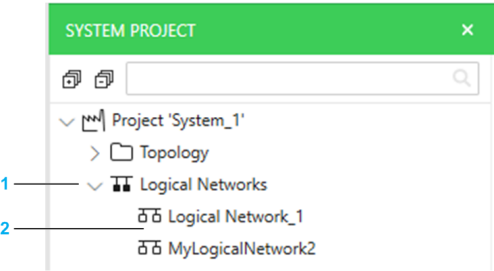
The **Logical Networks** node shows the logical networks that exist in the topology. It lets you create, [page 576](#) and manage them.

You can view and configure properties of each logical network by opening its context menu and/or using the **PROPERTIES** pane.

NOTE: You can highlight logical networks in physical views, [page 625](#).

Logical Networks Node Description

The following figure shows an example of the **Logical Networks** node and its content.



Item	Description
1	Logical Networks root node. NOTE: Entities assigned to a logical network are not shown in this view. To see to which logical network an entity is assigned to, select the entity and expand the IP address node in the PROPERTIES pane (CONFIGURATION tab) or open the LOGICAL NETWORK VIEW , page 541 .
2	User-created logical networks. You can change the order in which logical networks are displayed by dragging a logical network to its new position.

Logical Networks Node Actions

The following table describes the commands of the context menu of the **Logical Networks** node.

Command	Description
New Logical Network	Creates a new logical network, page 576 with properties set to default values.
Open Logical Network View	Opens the LOGICAL NETWORK VIEW , which lets you view and edit communication attributes of entities and reassign them to other existing logical networks.

Logical Network Actions

The following table describes the commands of the logical network context menu.

Command	Description
Open With	Opens a submenu that contains a list of views that are suited to view the logical network.
Show in Active Views	Selects and shows the logical network in the other windows and panes that are open and where the logical network appears (for example, in the LOGICAL NETWORK VIEW).
Delete	Deletes the logical network after confirming the command. You cannot delete a logical network if entities are assigned to it.

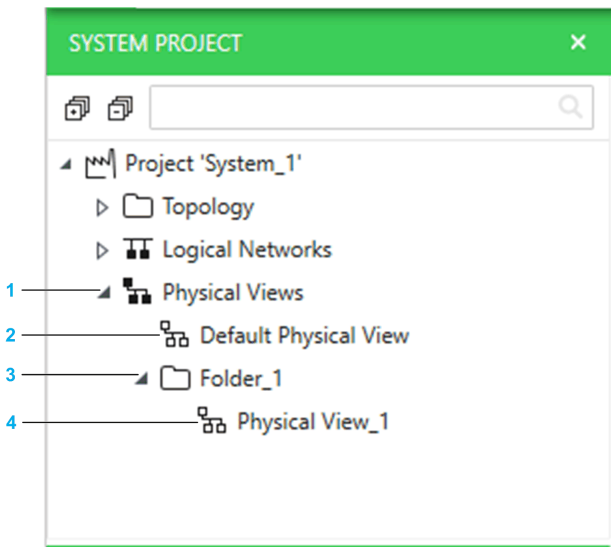
SYSTEM PROJECT Pane - Physical Views Node

Overview

The **Physical Views** node shows the physical views, page 536 that exist in the topology. It lets you create and manage them.

Physical Views Node Description

The following figure shows an example of the **Physical Views** node and its content.



Item	Description
1	Physical Views root node. NOTE: Entities assigned to a physical view are not shown in this view. To view them, open the physical view.
2	Default physical view.
3	User-generated physical view folder. NOTE: You can change the order in which folders are displayed by dragging a folder to its new position.
4	User-generated physical view. You can change the order in which physical views are displayed by dragging a view to its new position.

Physical Views Node and Folder Actions

The following table describes the commands of the context menu of the **Physical Views** node and folders therein.

Command	Description
New Folder	Creates a folder.
New Physical View	Creates a new physical view and opens it.

Physical View Actions

The following table describes the commands of the physical view context menu.

Command	Description
Open	Opens the physical view. You can select several physical views and open them at once.
Delete	Deletes the physical view after you confirm the command. NOTE: Deleting a physical view does not delete the entities that appear in it from the topology. However, physical links are deleted.

Topology Explorer Views

Physical Views

Overview

You can open a physical view either:

- By double-clicking it in the **Physical Views** node of the **SYSTEM PROJECT** pane.
- By clicking **Physical View Editor** in the **Open With** submenu of an entity.

The physical view lets you design in a graphical way the hardware architecture (for example, entities and racks of M580 controllers), connect, page 608 the ports of entities, and view network-related information of entities.

You can add entities, page 565 to one physical view either from the **TOOLBOX** or from the **SYSTEM PROJECT** pane.

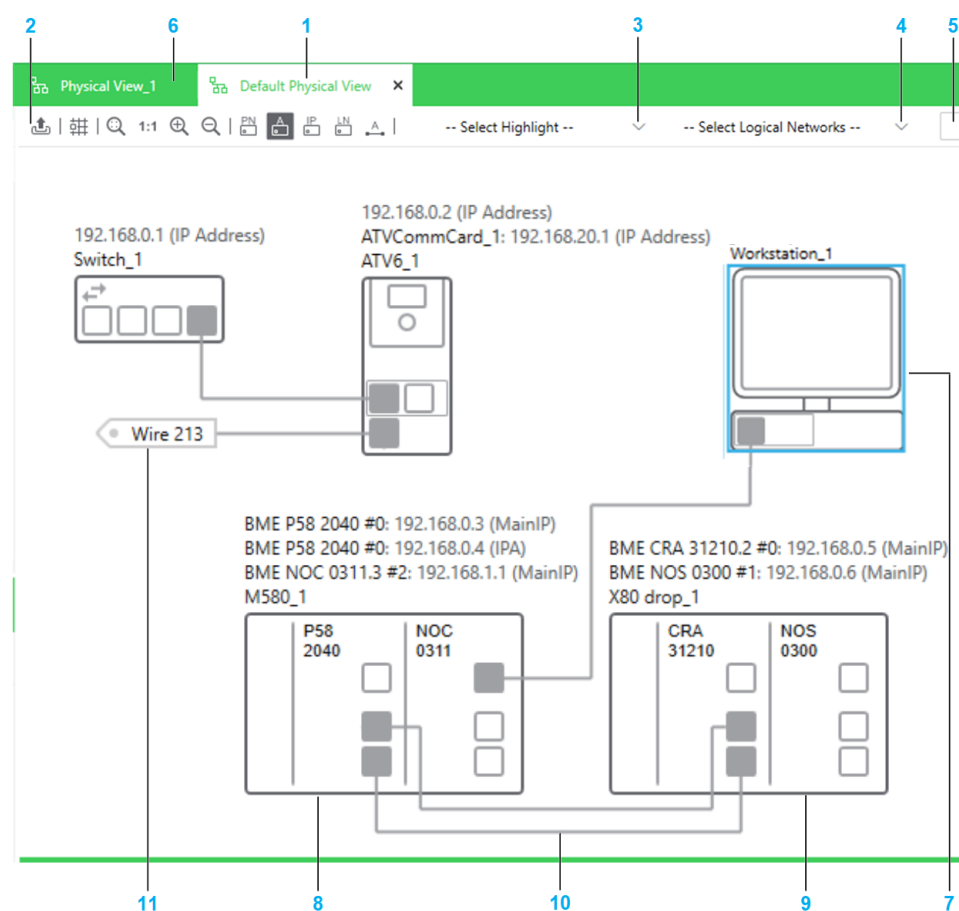
You can create several physical views in the **SYSTEM PROJECT** pane and move entities from one physical view to another.

Creating Physical Views

Step	Action
1	Right-click the Physical Views node in the SYSTEM PROJECT pane. Result: A context menu opens.
2	Click New Physical View . Result: A new physical view is created and opens.

Description

The following figure shows an example of a physical view.



Item	Description
1	Tab of the physical view that is active.
2	Toolbar of the physical view. Buttons that have a toggle function are highlighted when the command is enabled. For details, refer to the description of the context menu commands of the physical view workspace, page 538.
3	List that lets you identify graphically one of the following in the physical view: <ul style="list-style-type: none"> Port types, page 539 Redundant networks, page 539 Select None or close the physical view to remove highlights.
4	List that lets you select one or more existing logical networks and highlight the objects, page 625 that belong to them.
5	Search field that lets you locate objects, such as entities, modules, or ports in the physical view that is active.
6	Tabs of other physical views that are open.
7	Representation of the entity, page 578. The highlighted outline indicates that the entity is selected.
8	Representation of the local rack, page 527 of an M580 controller whose configuration was modified by configuring it.
9	Representation of the drop of the controller. You can reposition it, remove it from the view, or add it to another physical view.
10	Physical link, page 608 connecting the ports of two controllers appearing in the same physical view.
11	Connector of a physical link connecting the ports of two devices appearing in different physical views.

Workspace Actions

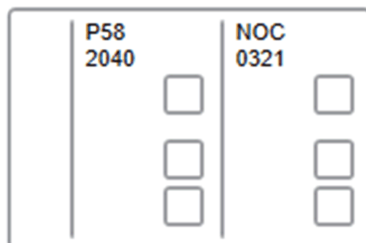
The actions that you can perform by using the commands described in the following table apply to individual physical views. Modifications to the appearance of each physical view are persisted when you close the engineering client and reopen the system on the same computer.

The following table describes the context menu commands of the physical view workspace.

Command		Description
Snap Device to Grid		Enables a functionality that lets you position entities by aligning them with an invisible grid.
View		Opens a submenu that lets you modify the zoom level of the view.
Display	Display Part Number Labels	When selected, shows the Part Number attribute of entities. For devices for which you have selected a DTM at the time of creation, the DTM Device Name attribute is displayed. The name is displayed above each entity, left-aligned.
	Display Device Labels	When selected, shows the Name attribute, page 676 of entities. The name is displayed above each entity, left-aligned.
	Display Link Labels	When selected, shows the Label attribute of physical links. The label is displayed towards the middle of the link and parallel to it.
	Display IP Address-es	When selected, shows IP addresses and logical networks of entities and modules. The information is shown on top of the label of the entity (if shown).
	Display Logical Networks	When both are selected, the format is <i>Name: logical network/CIDR - IP address (IP address name)</i> , where: <ul style="list-style-type: none"> <i>Name</i> appears only for modules and for communication cards. It corresponds to the Name attribute. <i>Logical network</i> corresponds to the Logical Network attribute. If the entity belongs to more than one logical network, one line for each network is displayed. <i>CIDR</i> corresponds to the CIDR notation suffix. <i>IP address</i>: If the entity has more than one IP address, one line for each IP address is displayed. For example, for an M580 controller, <i>BMEP582040 #0: MyNetwork/16 - 192.168.0.2 (MainIP)</i> NOTE: The following applies to the display of network data: <ul style="list-style-type: none"> IP address and logical network are not displayed if set to NOCONF. The data described in this table may not appear if the magnification is too low. You can view the <i>Logical network/CIDR - IP address</i> information by hovering with the pointer over an entity, module, or port, including the NOCONF configuration.
Export		Lets you save to a file the graphical layout of the entire physical view independently of the zoom level. The data that is displayed in the workspace (for example, entities, links, names of entities, labels, highlights) is captured. It enables you to open the graphical layout by using a CAD software. You can create files of the following types: <ul style="list-style-type: none"> Microsoft XPS GraphML

The following figure shows an example of the information that is displayed for a local rack when the display of names, IP addresses, and logical networks is enabled.

M580CPU_1: Default Network /24 - 192.168.0.2 (Main)
 M580CPU_1: Default Network /24 - 192.168.0.3 (A)
 NOCCTRL_1: Logical Network_1 /24 - 192.168.10.1 (Main)
 NOCCTRL_1: Logical Network_1 /24 - 0.0.0.0 (A)
 M580_1



Identification of Port Types

When you select **Port Types** from the **Select Highlights** list, ports are shown in a specific color depending on the type of network that they support.

Network type	Color
DIO	Orange
RIO	Green

NOTE: Disabled ports are shown in a light shade of gray.

Identification of Redundant Networks

When you select **Redundant Networks** from the **Select Highlights** list, the redundant ports of the following devices are shown in a specific color.

Network type	Color	Module/entity	Port
Primary network	Red	Hirschmann switches	Port1
Redundant network	Blue	Hirschmann switches	Port2

NOTE: Disabled ports are shown in a light shade of gray.

Physical links that connect ports of the same color are shown in that color. The link must be direct or connect ports of one or more ConneXium switches configured as **RIO** type and located between the Hirschmann switches.

Physical links connecting ports that belong to different redundant networks are shown in black. Analyze, page 1090 the topology to obtain details.

Entity Actions

The following table describes the commands of the context menu of an entity that are related to physical views. For a description of the other commands, refer to the topic describing the entity context menu, page 570.

Command	Description
Cut From View	Removes the representation of the entity (or the selection of entities) from the physical view after you paste it in another physical view. If the other connected entity is in a different physical view, physical links are also cut and pasted as <i>remote links</i> , page 608.
Remove From View	Removes the representation of the entity (or the selection of entities) from the physical view. Ports that the entity is connected to remain connected and are specifically identified, page 608. The corresponding physical links are hidden but appear again when the entity is added to a physical view again. NOTE: To keep the ports to which the entity is connected available for connection, remove the links before removing the entity from the view.

Locating Objects by Using the Search Field

The search field lets you locate the following objects by inputting all or part of their name:

- Workstations
- Controllers
- Devices
- Modules
- Network interface cards
- Ports
- Physical links

The search field is not case-sensitive.

The full name of objects that contain the string are displayed in a list as you type. Results are listed in ascending alphanumeric order.

To locate an object from the list, double-click it. The object is shown centered in the physical view. To locate several objects, select each one in the list then, double-click one of them. Objects that are selected in the list remain selected until you clear them by clicking them again.

The list of results remains available until you clear the search field. To show the list again, click inside the search field (if it has lost the focus) and press the **Down Arrow** key.

You can also use the following keys with the search field.

Key	Action
The Down Arrow key	Shows and/or enters the list of results. Once in the list, lets you move down the list. The object that has the focus is highlighted.
The Up Arrow key	Lets you move up in the list of results.
The Spacebar key	Selects a highlighted object. To clear a selected object, put the focus on the object again and press the key.
Enter	Locates the selected objects in the physical view.

Multiselection of Entities

When you select multiple objects in a physical view, the **PROPERTIES** pane shows the tabs and attributes that the objects have in common.

LOGICAL NETWORK VIEW

Overview

You can open the **LOGICAL NETWORK VIEW** either:

- By clicking **Open Logical Network View** in the context menu of the system root folder, the **Logical Networks** node, or a logical network in the **SYSTEM PROJECT** pane.
- By clicking **Logical Network View** in the **Open With** submenu of an entity.
- By using the **Views** menu in the toolbar, [page 515](#).

It gives you an overview of the network configuration of entities that have at least one IPv4 address that can be configured in the **PROPERTIES** pane. It facilitates the management of many entities, [page 627](#) at the system level.

The view is refreshed when you or other users modify the configuration of entities that are displayed.

NOTE: IPv6 addresses are not shown in this view.

Description

The following figure shows an example of the **LOGICAL NETWORK VIEW**.

Logical Network / Device	Interface	IP Address	Status
NOCONF			
M580_1\BME XBP 0800 #0\BME P58 2040 #0	Embedded Interface\MainIP	0.0.0.0	No Configuration
M580_1\BME XBP 0800 #0\BME P58 2040 #0	Embedded Interface\IPA	0.0.0.0	No Configuration
ATV6_1	Embedded Interface\IP Address	0.0.0.0	No Configuration
Logical Network_2 /24 - 192.168.20.0			
Logical Network_1 /24 - 192.168.10.0			
ModbusTCP_1	Embedded Interface\IP Address	192.168.100.5	Invalid IP Address
M580_1\BME XBP 0800 #0\BME NOC 0321 #2	Embedded Interface\MainIP	192.168.10.2	Valid IP Address
ATV6_1\ATVCommCard_1	Ethernet Interface\IP Address	192.168.10.4	Valid IP Address
Default Network /24 - 192.168.0.0			
M580_1\BME XBP 0800 #0\BME NOC 0321 #2	BKP_Interface\IP Address	192.168.0.254	Valid IP Address

Item	Description
1	The column shows the name of the following objects that exist in the topology: <ul style="list-style-type: none"> Logical networks. Entities and modules for which at least one IPv4 address can be configured in the PROPERTIES pane of the Topology Explorer.
2	Name of the interface, page 679 of the entity to which an IP address is assigned. If the interface manages several IP addresses, their name is indicated. <p>The following syntax is used in this column to identify interfaces and their IP addresses:</p> <ul style="list-style-type: none"> <i>Embedded Interface</i>: Identifies the internal interface of the entity (host address). <i><Port name>_Interface</i>: Indicates that the interface is not the embedded one (for example, the interface of the external ports of a routing switch, <i>Port1_Interface</i>).
3	IP address that is assigned to the interface. <p>You can change the address, page 627.</p>
4	Indicates the validity of the IP address in relation to the logical network that the interface belongs to. <p>Possible values:</p> <ul style="list-style-type: none"> No Configuration: The interface does not belong to a logical network. Valid IP Address: The IP address is valid on the logical network the interface belongs to. Invalid IP Address: The IP address is not valid on the logical network the interface belongs to. You can reassign it.
5	Lets you create a logical network, page 576.
6	Search field. Enter one or more terms separated by a space and the window displays objects that contain any of these terms. Objects are displayed with the logical network they belong to.
7	Entries are grouped by logical network. The logical network group indicates the name of the logical network that the interface belongs to. In addition, the CIDR notation suffix and address of the network are indicated. <p>You can change the logical network assignment, page 627 but you cannot create logical networks in this view.</p>
8	For entities that manage several IP addresses, an entry is displayed for each one of them. The entry appears under the respective logical network group.
9	Interfaces pertaining to a same entity appear under different logical network groups if the assignment of their respective interfaces is different.

Item	Description
10	Existing logical networks to which no interface is assigned are also shown.
11	Interfaces for which no logical network is defined are shown in the NOCONF group.

Context Menu Commands

The following table describes the commands of the entity context menu that are related to the **LOGICAL NETWORK VIEW**. For a description of the other commands, refer to the topics describing the context menu commands for logical networks and entities, page 570.

Command	Description
Reassign IP Address	<p>Lets you replace the invalid IP address of the selected interfaces with the next available IP address on the logical network.</p> <p>If the status of an address is Valid IP Address, it remains unchanged except if it is not unique, in which case the next available IP address is assigned to the interface.</p> <p>If you select several interfaces in the view, IP addresses are reassigned in the order that you selected the interfaces.</p> <p>Within a logical network, if you select the interfaces that have a same IP address, the IP address of the interface that you selected first is unchanged (given it is valid).</p> <p>NOTE: If you select several interfaces to reassign their IP addresses and not enough IP addresses are available, the reassignment of IP addresses is done only partially and a notification is displayed.</p>

Multiselection of Entries

When you select multiple entries in the **LOGICAL NETWORK VIEW**, the **PROPERTIES** pane shows the tabs and attributes that the entities have in common.

Communication Mapping Views

Overview

The Communication Mapping Views let you establish communication, page 641 between devices (except devices of the **EPE Managed Devices** product range, page 516) and scanner services of a given M580 controller and between M580 controllers.

Two distinct views exist to manage communication for the following protocols:

- Modbus TCP
- EtherNet/IP

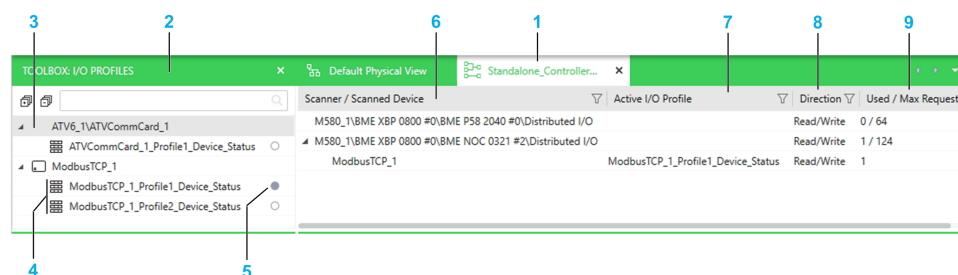
Use either method to open the Communication Mapping View from the context menu of the device or module acting as the scanner:

- By clicking a protocol in the **PAC > Communication Mapping** submenu.
- By clicking **Open With > Communication Mapping <Protocol> View**.

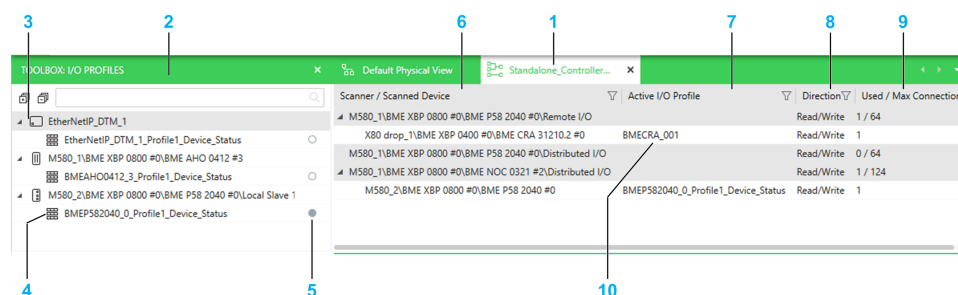
NOTE: You can also open the Communication Mapping View from the context menu of a device given it was added to a scanner. If the device is scanned by several services that belong to different controllers, it opens a Communication Mapping View for each one.

Communication Mapping Views

The following figure shows an example of the Communication Mapping View for devices communicating by using the Modbus TCP protocol.



Communication Mapping View for devices communicating by using the EtherNet/IP protocol.



Item	Description
1	Name of the M580 controller whose scanner services appear in this view. The communication protocol is indicated in a tooltip.
2	<p>TOOLBOX pane that lists the devices and their modules that exist in the topology and which can be scanned. Devices appear in the pane only once their I/O profile data is available, page 709.</p> <p>The content of the pane is contextual and related to the Communication Mapping View that is active.</p> <p>You can filter the list by using the search field.</p> <p>NOTE: Devices that appear under this node may already be scanned by using the scanner services of another M580 controllers.</p>

Item	Description
3	Node representing a device that can be scanned.
4	<p>I/O profiles, page 709 that exist for each device including local slaves, page 710.</p> <p>NOTE: The values that you see in the PROPERTIES pane when you select an I/O profile in this pane are those that are configured for the device.</p>
5	<p>If an instance of the I/O profile was added to a scanner, the button is shown selected.</p> <p>The information is related to the Communication Mapping View that is active.</p>
6	<p>Column showing the scanner services of the controller and the devices that they are scanning including local slaves.</p> <p>If an M580 controller is equipped with a remote I/O (RIO) scanner, it appears in the Communication Mapping View for EtherNet/IP.</p>
7	Column showing the I/O profile of the device that is being used for scanning and that is the active profile. It is an instance of the I/O profile of the device. You can edit certain attributes, page 638 of the active profile and its requests/connections without impacting the values of the device I/O profile.
8	For each scanner and I/O profile, indicates the direction of the communication.
9	<ul style="list-style-type: none"> • Used: <ul style="list-style-type: none"> ◦ For each active I/O profile, indicates the number of requests or connections, page 709 that are defined. ◦ For each scanner, indicates the total number of requests or connections that are defined in the active I/O profiles. <p>The values are updated when you add/remove requests or connections in the I/O profile of the device.</p> • Max Requests/Max Connections: The maximum number of requests or connections that the scanner supports.
10	<p>If you add a drop with a BMECRA**** adapter module to an M580 controller and apply changes, the adapter module and its I/O profile appear automatically under the RIO scanner of the controller module and cannot be moved.</p> <p>In general, modules of an M580 controller that appear automatically under a scanner in the Communication Mapping View cannot be moved and do not appear in the TOOLBOX pane.</p>

NOTE: You can shift columns and sort and filter entries.

Context Menu Commands

The following table describes the commands of the active I/O profile context menu.

Command	Description
Reset To Default	<p>Replaces the values of the active I/O profile with the values from the device, page 709 that appear in the I/O PROFILE tab of the PROPERTIES pane when you select the device or its I/O profile in the TOOLBOX pane.</p> <p>The command applies to the following attributes:</p> <ul style="list-style-type: none"> For Modbus TCP: <ul style="list-style-type: none"> UnitID Health Timeout (ms) Repetitive Rate (ms) Last Value Gateway/Bridge Device For EtherNet/IP: <ul style="list-style-type: none"> RPI T->O Timeout Multiplier Input Fallback Mode RPI O->T
Cut	Copies the active I/O profile and removes it from the scanner.
Copy	Copies the active I/O profile.
Remove From Scanner	<p>Removes the active I/O profile from the scanner.</p> <p>Values that you changed, page 638 in the active I/O profile are discarded.</p>
Device	Opens a submenu containing commands, page 570 that pertain to the scanned device.

The following table describes the commands of the context menu of the scanner. For a description of the other commands, refer to the topic describing *entity context menu commands*, page 570.

Command	Description
Paste	Adds the communication profile that you cut or copied to a scanner in the same or another compatible Communication Mapping View.

Multiselection of Entries

When you select multiple entries in the Communication Mapping View or **TOOLBOX** pane, the **PROPERTIES** pane shows the tabs and attributes that the objects have in common.

I/O Mapping Editor

Overview

The **I/O Mapping Editor** lets you view, create, and modify I/O items, page 642 of communication profiles and read/write requests/input/output connections that are configured for devices.

Modifications that you make to the read/write length and input/output size of a request or connection are automatically reflected in the editor.

You cannot use the **I/O Mapping Editor** to configure devices of the **EPE Managed Devices** product range, page 516. you must use the communication mapping, page 725 from the **Projects Explorer** (mapping stage).

You can open the editor either:

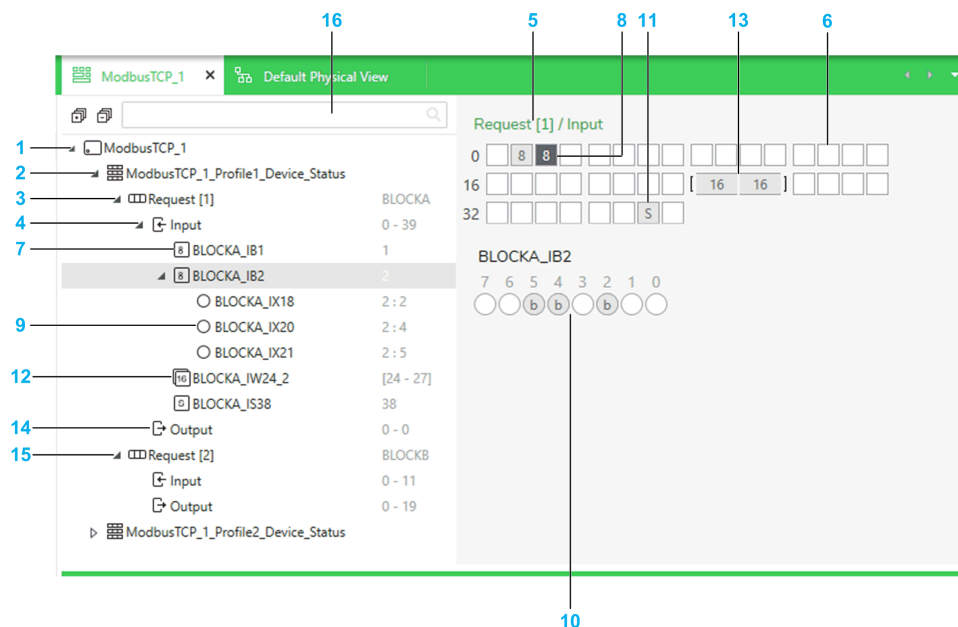
- By clicking **I/O Mapping Editor** in the **Open With** submenu of a device that communicates by using the Modbus TCP or EtherNet/IP protocol.
- By clicking **Edit** in the properties of a communication profile that is shown in the **I/O PROFILE** tab of the **PROPERTIES** pane.

NOTE: Depending on the protocol, the following terminology is used :

- *Read/write requests* for the Modbus TCP protocol.
- *Input/output connections* for the EtherNet/IP protocol.

Description

The following figure shows an example of the **I/O Mapping Editor** showing the items that exist for the first read request (input) of *Profile1* of a sample device communicating by using Modbus TCP. Manual mapping mode is enabled for this communication profile. The view is similar for EtherNet/IP devices.



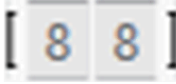




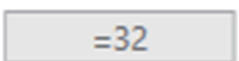

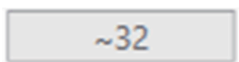



Item	Description
1	Name of the device whose I/O mapping is being edited.
2	Node representing the first communication profile, page 709 that exists for the device. The other profiles that exist appear below.
3	Node representing the first request or connection of the communication profile. By default, registers that are addressed by a request/connection are assigned the name <i>BLOCKx</i> where <i>x</i> is an alphabetical index from A to BL that is incremented for each request/connection of a profile.

Item	Description
4	<p>Node representing the input register that is addressed by the request/connection.</p> <p>The register size, page 709 that you configured appears next to it (format, 0 - <length (bytes) -1>).</p> <p>Select the node to view a graphical representation of the items that are defined.</p>
5	Graphical representation of the register size that is read or written for each request/connection.
6	Representation of bytes for which no mapping has been performed.
7	<p>Items that were created appear under the Input and Output nodes with the name, page 642 that you configured. An icon indicates their data type.</p> <p>Their offset appears next to them.</p> <p>Select an item to show it in the graphical representation of the register.</p>
8	<p>Example of representation of an item of type Byte that is selected. The number indicates the size in bits.</p> <p>(This data type is shown for illustration only and cannot be created with the Modbus TCP protocol.)</p>
9	<p>If bit mappings are defined for an item, each one is shown as a sub-item.</p> <p>The offset of the item and the position of the bit appear next to them (format, <offset> : <position>).</p>
10	<p>Representation of the bit field of the selected item (if applicable). The name, page 642 of the item appears on top.</p> <p>In the example, bits were mapped at positions 2, 4, and 5 of item <i>BLOCKA_IB2</i> of type Byte at offset 2.</p>
11	<p>Example of representation of an item of type String.</p> <p>The number indicating the size is replaced by S.</p> <p>(This data type is shown for illustration only and cannot be created with the Modbus TCP protocol.)</p>
12	<p>Item consisting of an array of words.</p> <p>The offset of the low byte of the first word and of the high byte of the last word are indicated next to it in brackets (format, [<offset low byte> - <offset high byte>]).</p>
13	Example of representation of an item of type Array of Word with two words.
14	<p>Node representing the output register that is addressed by the request/connection.</p> <p>For requests, the register write length, page 709 that you configured appears next to it (format, 0 - <length (bytes) -1>).</p> <p>Select the node to view a graphical representation of the items that are defined.</p>
15	Second request/connection of the first communication profile for which no mapping has been performed.
16	Search field. Enter one or more terms separated by a space and the pane displays objects that contain any of these terms. Objects are displayed with their parent structure.

Item Representation

The following table shows how items of the various types appear in the graphical representation of the register.

Item	Representation	Representation of arrays
Bit item		n/a
Byte item		
String item		
Word, Int, and UInt items		
DWord, DInt, and DUInt items		
Real and Time items		

Context Menu Commands

The following table describes the context menu commands that are available for the various elements shown in the **I/O Mapping Editor**. For a description of the other commands, refer to the topic describing entity context menu commands, page 570.

Command	Description
Define Items	Opens a dialog box that lets you create one or more items, page 642 by using compatible data types.
Delete	Deletes the selected items including bit mappings if applicable.

NOTE: The commands are not available when the mapping mode, page 709 for the communication profile is set to **Automatic**.

DHCP VIEW

Overview

You can open the **DHCP VIEW** either:

- By clicking **Open DHCP View** in the context menu of the system root folder in the **SYSTEM PROJECT** pane.
- By clicking **DHCP View** in the **Open With** submenu of a device or module that features the DHCP service.
- By using the **Views** menu in the toolbar, page 515.

It lets you view and modify, page 624 the DHCP client/server relationship of entities that feature the service.

Devices and modules can appear in light gray or italic depending on the status of their DHCP service.

The **DHCP VIEW** is refreshed when you or other users modify the topology and/or change the DHCP attribute values of a device/module in the **PROPERTIES** pane.

Description

The following figure is an example of the **DHCP VIEW** that shows the DHCP-capable devices and their relationship.

The screenshot shows the DHCP VIEW window with the following data:

1	2	3	4	5	6
DHCP Server / DHCP Client	Interface	Protocol	IP Address	Identified by	Identifier
Not Assigned					
ATV6_1	Embedded Interface	DHCP	192.168.0.2	DeviceName	ATV6_1
M580_1\BME XBP 0800 #0\BME P58 2040 #0	Embedded Interface	BOOTP, DHCP	192.168.0.4		Standalone_Controller_1\M580_1\BME XBP 0800 #0\BME P58 2040 #0\Embedded Interface
M580_1\BME XBP 0800 #0\BME AHO 0412 #3	Embedded Interface	DHCP	192.168.0.6	DeviceName	Mx80_03_AHO0412
X80 drop_1\BME XBP 0400 #0\BME CRA 31210.2 #0	Embedded Interface	DHCP	192.168.0.7	DeviceName	BMECRA_001
ModbusTCP_1	Embedded Interface	DHCP	192.168.20.1	DeviceName	ModbusTCP_1
M580_1\BME XBP 0800 #0\BME NOC 0321 #2	Embedded Interface	BOOTP, DHCP	192.168.20.254		Standalone_Controller_1\M580_1\BME XBP 0800 #0\BME NOC 0321 #2\Embedded Interface
EtherNetIP_DTM_1	Embedded Interface	DHCP	192.168.0.2	MacAddress	
M580_2\BME XBP 0800 #0\BME P58 2040 #0	Embedded Interface	BOOTP, DHCP	192.168.0.8		Standalone_Controller_2\M580_2\BME XBP 0800 #0\BME P58 2040 #0\Embedded Interface
TeSysT_1	Embedded Interface	DHCP	0.0.0.0	DeviceName	TeSysT_1

Item	Description
1	Lists the devices and modules of the topology that feature a DHCP server or client service.
2	For a DHCP server, indicates the name of the interface, page 679 that provides the DHCP services. For a DHCP client, indicates the name of the interface to which the DHCP server provides the IP address settings.
3	Protocol, page 694 that is configured for the device/module.
4	IP address of the interface that is indicated in the Interface column.
5	Value of the Identified by attribute, page 694 that is configured for the client device/module.
6	For a DHCP client device/module, the value of its Identifier attribute, page 694. For a DHCP server, the value of its DHCP Name attribute.
7	Node that lists the devices/modules that do not subscribe to a DHCP service. Devices/modules whose DHCP client service is disabled, page 694 appear in light gray.
8	Device/module that features a DHCP server that is enabled. For redundant controllers, only DHCP servers of local rack A appear.
9	Devices/modules that are automatically added to the list of clients of the DHCP server of the parent device. Their DHCP client service is enabled and the DHCP Server Name attribute read-only.
10	Device/module that you moved to add it to the list of clients of the DHCP server of the parent device. Its DHCP client service is enabled.
11	Device/module whose DHCP client service is disabled, page 694 (appears in light gray).
12	Device/module that features a DHCP server, which is disabled, page 694 (appears in light gray).
13	Device/module that belongs to the list of clients of a device whose DHCP server is disabled (appears in italic) and does not receive IP addresses and network parameters from the server.

NOTE: You can shift columns and sort and filter entries.

Context Menu Commands

You can open a context menu for devices and modules, page 570.

Multiselection of Entries

When you select multiple entries in the **DHCP VIEW**, the **PROPERTIES** pane shows the tabs and attributes that the selected entries have in common.

Fieldbus Editor

Overview

You can use the **Fieldbus Editor** with PROFIBUS Remote Master (PRM) gateways of the **Network Fieldbus Devices** product type, **Gateway** product range.

It shows the field devices that are connected to the gateway from which the editor was opened. It lets you view and [edit the fieldbus](#), [page 620](#) by adding field devices and network coupling devices from the **TOOLBOX** and configure them.

You can edit several fieldbuses at the same time by opening a **Fieldbus Editor** for the gateway of each fieldbus.

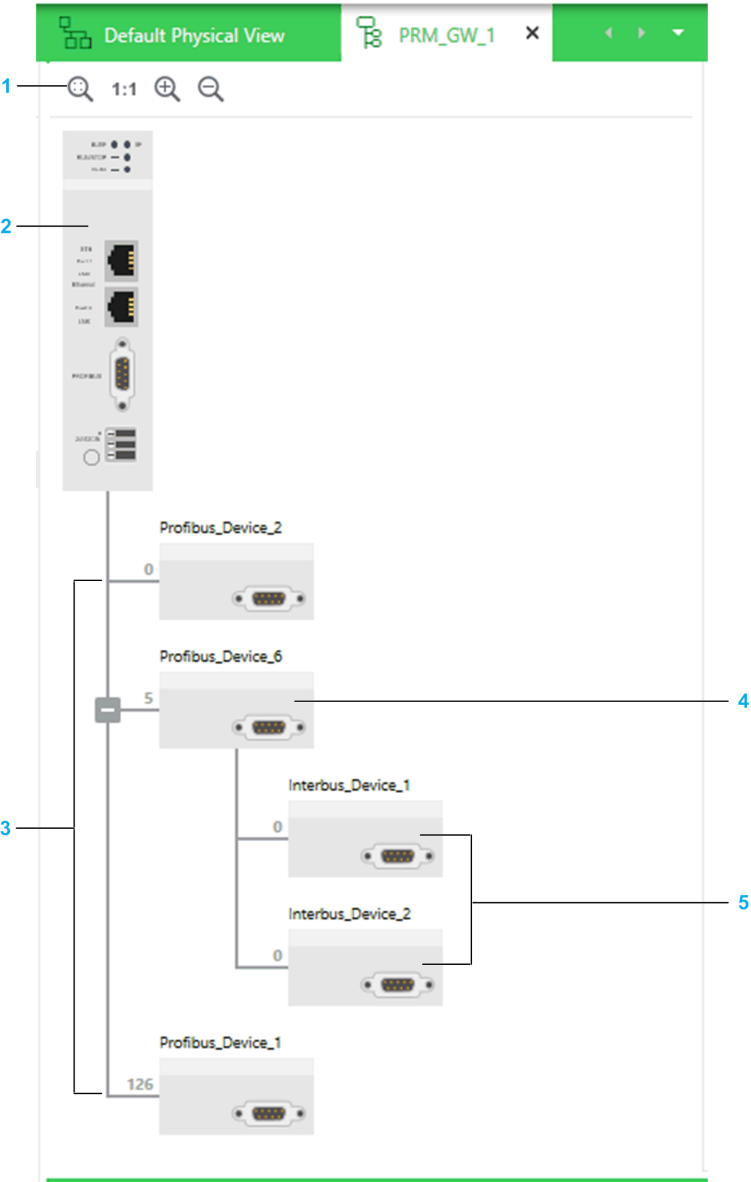
You can open the **Fieldbus Editor** either:

- By clicking **Edit Fieldbus** in the context menu of a PRM device that manages a fieldbus (for example, a gateway or a HART module).
- By clicking **Fieldbus Editor** in the **Open With** submenu of a field device.

NOTE: To configure a PRM of the **EPE Managed Devices** product range, refer to the topic describing how to [configure PRMs](#), [page 601](#).

Description

The following figure shows an example of the **Fieldbus Editor** showing a PROFIBUS gateway and three secondary devices.



Item	Description
1	Editor toolbar.
2	PRM gateway.
3	PROFIBUS secondary devices. They are automatically positioned on the fieldbus as you create them.
4	PROFIBUS-to-INTERBUS gateway (coupling module) device.
5	INTERBUS secondary devices.

Context Menu Commands

The following table describes the context menu commands that are related to the **Fieldbus Editor**. For a description of the other commands, refer to the topic describing entity context menu commands, page 570.

Command	Description
Cut/Copy	Lets you copy or cut a field device and its DTM. NOTE: When you copy a PROFIBUS-to-INTERBUS gateway, its secondary devices are also copied.
Paste	Lets paste copied or cut field devices within the same fieldbus or another one of the same type that is open in another Fieldbus Editor .

Multiselection of Devices

When you select multiple devices in the **Fieldbus Editor**, the **PROPERTIES** pane shows the tabs and attributes that the devices have in common.

ACCESS CONTROL VIEW

Overview

You can open the **ACCESS CONTROL VIEW** either:

- By clicking **Open Access Control View** in the context menu of the system name in the **SYSTEM PROJECT** pane.
- By clicking **Access Control View** in the **Open With** submenu of a module.
- By clicking **Edit** in the **Access Control** attribute of the **SECURITY** tab (**PROPERTIES** pane) of modules managing security attributes.
- By using the **Views** menu in the toolbar, page 515.

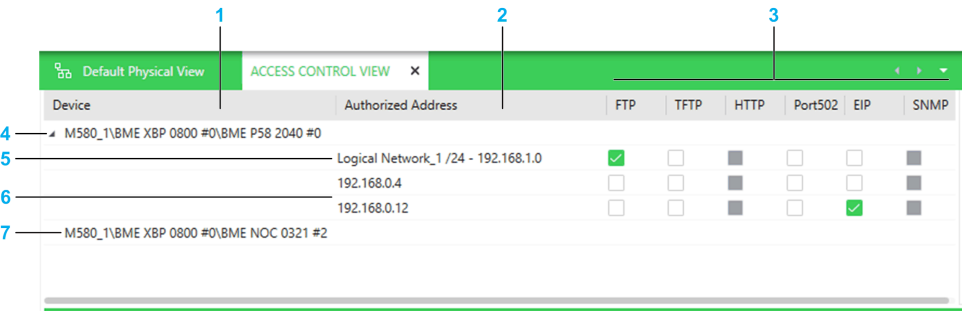
It gives you an overview of the Ethernet communication permissions that are configured for entities that exist in the topology and that manage security attributes, page 690.

The view is refreshed when you or other users modify the configuration of objects that are displayed.

NOTE: You must enable Ethernet services and access control at the module level by using the **SECURITY** tab before you can use this view.

Description

The following figure shows an example of the **ACCESS CONTROL VIEW**.



Item	Description
1	Column showing modules of entities that exist in the topology and that manage security attributes related to Ethernet communication.
2	For each module that appears in the Device column, shows the IP addresses of entities that are authorized to communicate with the entity by sending requests to it. Instead of or in addition to individual IP address, existing logical networks can appear. This allows entities that belong to the logical network to communicate with the module. The field is empty by default.
3	Shows the Ethernet services that are featured by the modules and which ones are used when access control, page 691 is enabled at the module level. You can enable (check box selected) or disable services individually for each authorized address. The corresponding service, page 690 must be enabled at the module level first (if applicable). Disabling a service at the module level makes the service unavailable in this view and resets its value for the authorized entries of the module.
4	Entity for which the IP address of an authorized entity and a logical network were added.
5	Authorized logical network. The CIDR notation suffix and subnet address of the network are indicated.
6	IP address of an entity authorized to communicate with the module.
7	Module for which no authorized IP address nor logical network has been added.

NOTE: You can shift columns and sort and filter entries.

Context Menu Commands

The following table describes the context menu commands of objects that are related to the **ACCESS CONTROL VIEW**. For a description of the other commands, refer to the topic describing [entity context menu commands](#), page 570.

Object	Command	Description
Entity	Add Authorized Address	Opens a dialog box that lets you add the following: <ul style="list-style-type: none">• An existing logical network.• An IP address
Authorized address	Delete	Removes the address or the selection of addresses from the view and revokes the permission to communicate with the entity.

Multiselection of Entries

When you select multiple entities in the **ACCESS CONTROL VIEW**, the **PROPERTIES** pane shows the tabs and attributes that they have in common.

Basic Diagnostic

Overview

You can open the basic diagnostic window for certain devices by right-clicking a device and selecting **Open Basic Diagnostic**.

It displays various diagnostic codes, page 1118 for the device.

You can open diagnostic windows for several devices at the same time.

The diagnostic window is refreshed when you or other users modify the configuration of the device.

Devices Supporting Basic Diagnostics

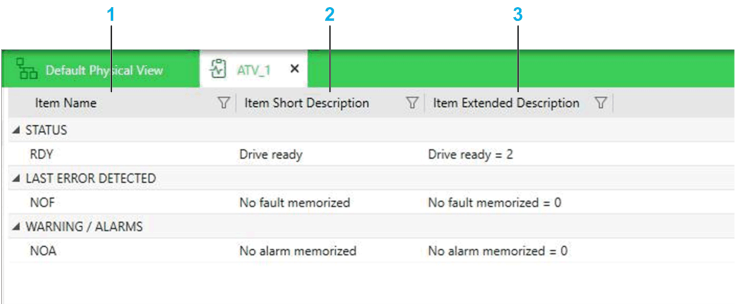
Basic diagnostics are available for devices of the **Altivar Variable Speed Drives** product range, page 516.

Prerequisites

Diagnostic data can be displayed for a device only if the engineering client can communicate with the device. If communication cannot be established or is interrupted, a message is shown.

Description

The figure shows an example of the basic diagnostic window.



Item	Description
1	Diagnostic codes are grouped in three categories. If no diagnostic data is received for a given category, the corresponding column is empty.
2	Short description of the condition that is detected.
3	The extended description provides additional data about the condition.

Troubleshooting

If you cannot obtain diagnostic information, ensure that the SE SODB ControlExpert Host X service (where X represents the version) of the Service Oriented Device Bus (SODB) diagnostic engine is running on the Control Expert server computer (**Services** window of the Windows operating system). Otherwise, start it.

Rack Viewer

Overview

You can open the **Rack Viewer** inside the **Topology Explorer** either:

- By clicking **Open Rack Viewer** in the context menu of a rack of an M580 controller.
- By double-clicking the handle of a local rack or drop in a physical view.
- By clicking **Rack Viewer** in the **Open With** submenu of a backplane or module of a controller.

You can view the configuration and/or diagnostic information of several racks and controllers at the same time by opening a **Rack Viewer** for each one.

NOTE: When the **Rack Viewer** is the active view, the **TOOLBOX** is empty.

View Modes

The following view modes are available depending on whether system monitoring, page 515 is active or not.

System monitoring	View mode (tab name)	Description
Inactive	None	Shows the physical configuration of the local rack or a drop of a controller in a graphical way.
Active	OFFLINE	<p>Local racks of standby controllers are shown in a separate view.</p> <p>The view is static and read-only.</p> <p>You can select a backplane or a module to view and/or edit its properties in the PROPERTIES pane.</p> <p>NOTE: The Rack Viewer (offline) is refreshed when you or other users save changes that were made to the rack by configuring it.</p>
	ONLINE	<p>Shows the embedded Web pages of the controller or a communication module of the local rack if the prerequisites are fulfilled.</p> <p>The part number of the module that you are connected to is indicated in parenthesis next to the name of the tab.</p>

Prerequisites

To view the embedded Web pages of a module, the following prerequisites must be fulfilled:

- System monitoring is active, page 1107.
- The physical module is reachable at the IP address that is configured in the **PROPERTIES** pane.
- The identity of the physical module matches, page 1111 that of the module in the **Topology Explorer**.
- To view the **Rack Viewer** Web page (**ONLINE** tab), the M580 controller supports the functionality (see *Rack Viewer* in Modicon M580, Hardware, Reference Manual).

Working Principle

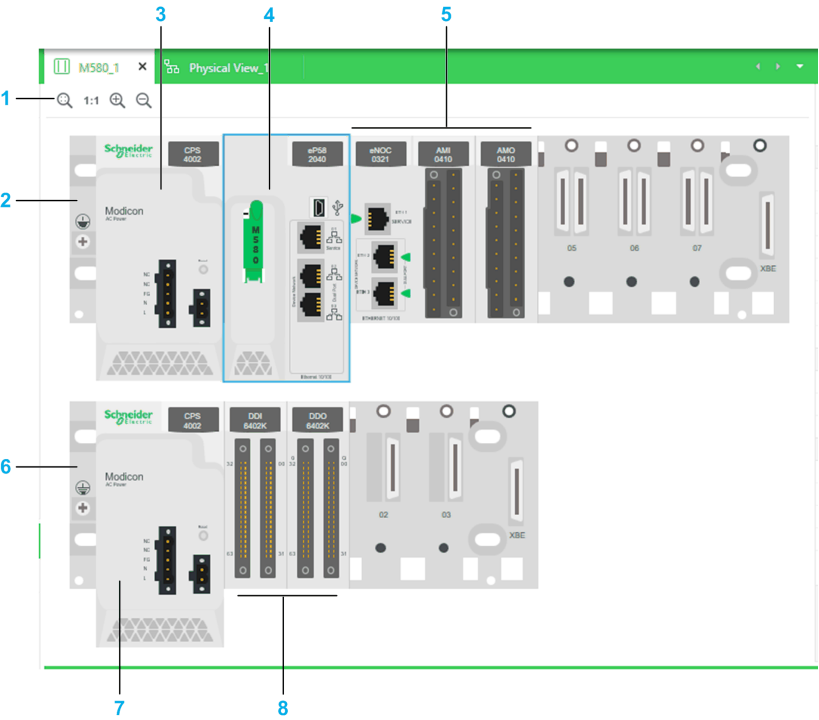
When system monitoring is active, if you try to open the **Rack Viewer** from the M580 local rack, the controller, or a BMENOC03•1 communication module, EcoStruxure Process Expert first verifies the identity of the physical M580 controller module.

- If the identity of the controller module matches and it supports the **Rack Viewer**, it is opened.
- If the controller module is not reachable, no match is detected, or if the controller module does not support the **Rack Viewer** (online), EcoStruxure Process Expert attempts to connect to each BMENOC03•1 module in the rack and opens the embedded Web pages of the first module that it can connect to.

You must enter the **Webuser Password**, page 588 when prompted.

Description

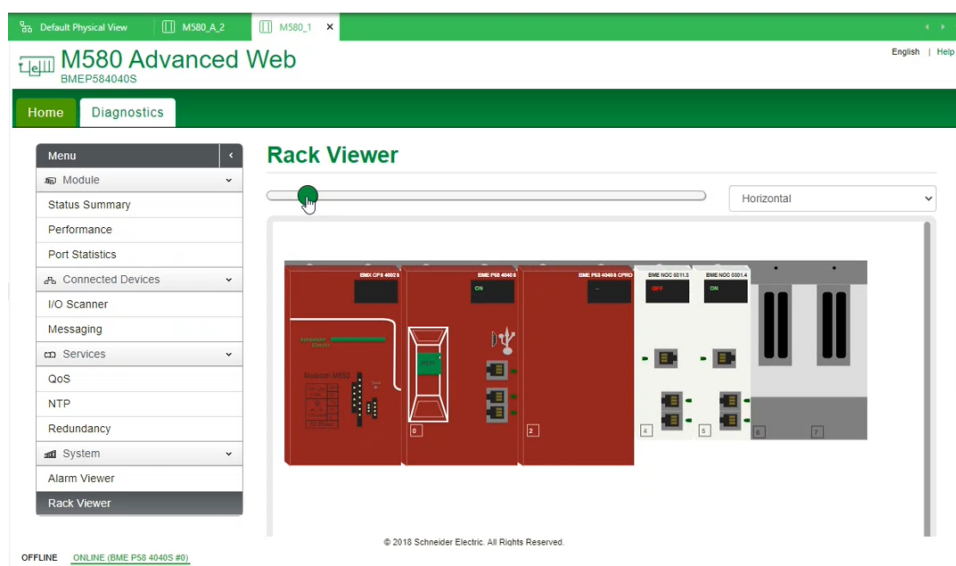
The following figure shows an example of the **Rack Viewer** when system monitoring is inactive. It shows a static view M580 local rack. An extended rack and various modules were added to the local rack by configuring the controller.



The content of the view is the same when system monitoring is active and the **OFFLINE** tab is selected.

Item	Description
1	Rack viewer toolbar.
2	X-bus/Ethernet backplane of the local main rack. Click either end to select it.
3	Power supply module.
4	Controller module (shown selected).
5	Various X80 communication and I/O modules that were added.
6	4-slot X-bus backplane of the local extended rack.
7	Power supply of the local extended rack.
8	Various X80 I/O modules.

The following figure shows an example of the **ONLINE** tab of the **Rack Viewer** for a BMEP584040S safety controller module.



For a description of the content, refer to the *Web pages* topic in the help of the module.

Workspace Actions

The following table describes the commands of the context menu of the static **Rack Viewer** workspace.

Command	Description
View	Opens a submenu that lets you modify the zoom level of the view.

Context Menu Commands

In the static **Rack Viewer**, you can open a context menu for the backplane and modules.

NOTE: For the backplane, right-click either end.

Multiselection of Objects

When you select multiple objects in the static **Rack Viewer**, the **PROPERTIES** pane shows the tabs and attributes that the objects have in common.

Creating the Topology of the System

Overview

This section describes how to use the **Topology Explorer** to create and configure topological entities modeling the topology of the system.

The Foundation Library contains templates that allow you to model the following infrastructure components:

- Logical networks (Ethernet)
- Workstations
- Controllers
- Ethernet STB islands and devices connected through CANopen extension
- Modbus TCP and EtherNet/IP I/O devices
- PROFIBUS remote masters (PRMs) and their decentralized peripherals (DPs)

Controllers and workstations function as engines where you can deploy Control or Supervision projects.

For more information on supported architectures, refer to the topic describing integrated architectures (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Organizing the Topology in the **SYSTEM PROJECT** Pane

Before Starting

Before creating the folder structure for the topology of the system in the **SYSTEM PROJECT** pane, page 527, define the appropriate segmentation and naming convention for your hardware infrastructure.

Creating Topology Folders

To create a topology folder, proceed as follows.

Step	Action
1	Right-click the Topology node in the SYSTEM PROJECT pane and select New Folder .
2	Edit the properties of the folder by using the PROPERTIES pane.

Topology Folder Actions

Right-click the **Topology** node or a topology folder to open a context menu with the following commands.

Command	Description	
New Folder	Creates a topology folder or subfolder.	
Open With ⁽¹⁾	Opens a submenu that contains a list of views and panes that are suited to view the folder.	
Export	Opens a submenu with the following commands:	
	Topology, page 647	Opens the Export window, which allows you to select the topological entities that you want to export in the form of a topology export file (.sbk). You can select entities of the folder and its subfolders as well as the folder hierarchy.
	Devices, page 660	Opens the Export window, which allows you to export the devices and logical networks of the topology in the form of a topology export file (.csv). The command is available only from the Topology root folder.
Import	Opens a submenu with the following commands:	
	Topology, page 651	Opens the Import window, which allows you to select an export file (.sbk). In a second step, you can select which compatible content of the export file you import in the topology.
	Devices, page 667	Opens the Import Devices window, which allows you to select a device export file (.csv). In a second step, you can select which devices of the export file you import in the topology. The command is available only from the Topology root folder.
Delete ⁽¹⁾	Deletes the folder, its contents, and associated configurations.	
(1) The command is not available from the Topology node.		

Moving Topological Folders

To move one or more folders in the **SYSTEM PROJECT** pane, drag your selection onto another folder. If a folder with the same name already exists, the contents of the two folders are merged after you confirm the operation.

Topological Folder Properties

The following table describes the attributes of topological folders that appear in the **PROPERTIES** pane.

Attributes	Default value	Description
Name	<i>Folder_n</i> where <i>n</i> is an incremental number starting at 1.	Editable name of the folder. You can use only names that are unique within the same folder level.
Comment	Blank.	Comment for the folder. You can edit the value by using free-form text.
Path	N/A	Path of the folder in the topology of the system. Read-only.

Adding Topological Entities to a System

Overview

To design the hardware architecture, add equipment from the [catalog, page 516](#) (**TOOLBOX** pane) to the topology of the system. Each instance of equipment that you add creates a topological entity.

Ways to Create Topological Entities

You can create topological entities by dragging equipment from the **TOOLBOX** to the following areas.

Target area	Result
The SYSTEM PROJECT pane.	You can select in which topological folder and in which position inside the folder you create the entity. NOTE: Consider creating the topological folder structure first. NOTE: You can add the entity to a physical view, page 568 later.
A physical view, page 536 .	<ul style="list-style-type: none"> You can position the entity in the workspace of the physical view. The entity is created in the Topology folder of the SYSTEM PROJECT pane. You can relocate it.

Selecting References

In general, the reference is predefined by the equipment that you select in the catalog.

For the following equipment, however, you can select a reference manually.

Object type	Reference selection
M580 controllers	In the Select CPU Reference dialog box after dragging the equipment to a physical view or the SYSTEM PROJECT pane. NOTE: To change the controller module reference once the controller is created, configure it.
Devices that use a device type manager (DTM), except devices of the EPE Managed Devices product range, page 516	You may need to select the reference that you want to use either: <ul style="list-style-type: none"> By selecting a DTM, page 629 from the local DTM catalog in the Select DTM Reference dialog box after dragging the equipment to a physical view or the SYSTEM PROJECT pane. Once the device is created, you cannot change the DTM reference anymore. By selecting a part number and version in the PROPERTIES pane after the device is created. By selecting a reference in the DTM window, page 629 after the device is created. Use the Open context menu command of the device.
TeSys T devices (TeSys Motor Controllers product range, page 516)	Refer to the topic describing how to add TeSys T devices, page 566 .
Modbus TCP and EtherNet/IP devices of the EPE Managed Devices product range, page 516	By selecting a template, page 603 in the Select Template dialog box after dragging the equipment to a physical view or the SYSTEM PROJECT pane.
PROFIBUS remote masters (EPE Managed Devices product range, page 516)	By selecting the platform of the controller platform, page 676 that communicates with the PRM in the PROPERTIES pane (Controller Family attribute) after the device is created.
Modicon managed and extended switches	By selecting a part number in the PROPERTIES pane after the entity is created.

Adding Entities to the Topology by Using the Physical View

Step	Action
1	Open the Topology Explorer .
2	Create and/or open the physical view.
3	In the TOOLBOX pane, select the equipment and drag it to the workspace of the physical view. Result: The entity is added to the system with properties set to default values and a graphical representation of the entity is displayed in the physical view. NOTE: The Select Template dialog box, page 584 may open.

Adding Entities to the Topology by Using the SYSTEM PROJECT Pane

Step	Action
1	Open the Topology Explorer .
2	If required, create the folder in which you want the entity to be located.
3	In the TOOLBOX pane, select the equipment and drag it to a folder or between existing entities in the SYSTEM PROJECT pane. Hover over the folder to expand its child structure. Result: The entity is added to the topology with properties set to default values. If you drag equipment to a folder that contains other entities, it is added in last position. NOTE: The Select Template dialog box, page 584 may open.

Adding TeSys T Devices to the Topology

To add a TeSys T equipment from the **Automation Devices** product type, **TeSys Motor Controllers** product range, page 516 to the topology, proceed as follows.

Step	Action
1	From the catalog, drag the TeSys T equipment to a physical view or the SYSTEM PROJECT pane. Result: The device is added to the topology with Part Number and Version attributes blank.
2	Right-click the device and select Open .
3	In the DTM window, select a commercial reference that supports Modbus TCP or EtherNet/IP communication and click OK .
4	To view the I/O profile data, page 709 of the device (PROPERTIES pane), in the Process Data section of the Parameter List tab, select a value other than None , and apply changes.
5	Close the DTM.
6	Result: The Part Number and Version attributes are populated with the information of the reference you selected.

NOTE: If you select a reference that communicates by using another protocol (for example, PROFIBUS), a notification opens when you close the DTM and the device is deleted from the topology when you acknowledge the notification.

This type of device can only be added after installing the required Library and it does not appear in the topology.

To add, for example, a Modbus serial TeSys T device, proceed as follows.

Step	Action
1	Install the Modbus Communication Library on the local computer.
2	In the Topology Explorer , configure the controller that is scanning the device.
3	In the DTM Browser , right-click the Host PC node and add the Modbus Serial Communication DTM from the Add window.
4	Add the TeSys T device to this DTM.
5	Open the TeSys T DTM and select a reference.

Adding Entities to a Physical View from the **SYSTEM PROJECT** Pane

Overview

Once you have added an entity to a system, page 565, you can add it to a physical view to design the hardware architecture in a graphical way, connect, page 608 the ports of entities, and view network-related information of entities.

You can show an entity only in one physical view at a time.

Adding Several Entities at Once

You can add to a physical view the entities contained in a folder by dragging the folder. The operation is not allowed if an entity already appears in another physical view.

Working With Controllers

If you add to a controller that already appears in a physical view a component that is normally shown separately in a physical view (for example, a drop), it is automatically added to the view.

If you replace a component of a controller that is normally shown separately in a physical view (for example, a drop), it is removed from the view.

You can add local racks and drops of a controller to different physical views by dragging them individually. To add the existing drops of a controller to a physical view, drag the **EIO Bus** from the **SYSTEM PROJECT** pane to the physical view.

For details of how controllers appear in physical views, refer to the topic describing the representation of entities, page 578.

Adding Entities to a Physical View from the **SYSTEM PROJECT** Pane

Step	Action
1	Create and/or open a physical view.
2	In the Topology section of the SYSTEM PROJECT pane, select one or more entities and drag them to the workspace of the physical view. Result: A graphical representation of the entity is displayed in the physical view.

Moving Entities Within the Physical View

You can drag one or more entities that you selected to another area of the same physical view if a cross with arrows appears next to the pointer. If no cross with arrows appears, click another area of the entity.

Moving Entities Between Physical Views

You can drag one or more entities that you selected to another physical view if a cross with arrows appears next to the pointer. If no cross with arrows appears, click another area of the entity. The physical views must be open side by side or overlapping in the same **Topology Explorer** window.

Existing physical links are redrawn according to the following rules:

- A physical link is moved only if both linked entities are moved. It is converted into a [remote link](#), [page 614](#) if only one of the linked entities is moved.
- A remote link is moved together with the entity it is connected to.
- A remote link is not converted into a solid link.
- The path of a link that was moved may be modified in the target physical view to help minimize collisions with other objects. This applies only to links for which automatic routing is enabled.

Topological Entity Context Menu Commands

Overview

This topic describes the commands of the context menu of topological entities and their modules in general. Depending on the entity and from where you opened the context menu, some commands may not appear .

The description of context menu commands of other objects or those that are related to a given view or pane can be found in the topic describing this view/pane.

NOTE: In physical views, commands that appear in the contextual menu of modules but that apply to the controller are grouped in the **PAC** submenu.

NOTE: For workstations, some commands are grouped in the **Control** and **Supervision** submenus. Commands in these submenus are available only if you created the corresponding services in the workstation, mapped an executable to the workstation, and built the executable. Additional conditions may apply; refer to the description of each command.

Description of Commands

The following table describes the commands of the context menu of topological entities and modules.

Command	Description	
Analyze	Performs an analysis, page 1090 of the configuration of the controller and its dependencies at the Topology Explorer level and displays the result in the notification pane.	
Back Up Data , page 880	Opens the Back Up Data dialog box, which lets you configure the data that you want to back up and create a controller data backup file. The command available only if you deployed a built Control Participant project to the controller.	
Clear Password	<p>Opens the Clear Password dialog box, which lets you remove the following passwords:</p> <ul style="list-style-type: none"> The Controller password, page 810 of the deployed Control Participant project and/or the Safety, page 925 password. The Simulator, page 810 password of the deployed Control Participant project. <p>If you created more than one Control Expert service, you must select one from a list.</p> <p>The command is available only if you set a password for at least one Control Expert service</p> <p>To use it for a given Control Expert service, the corresponding simulator instance must be running.</p>	
Communication Mapping	Opens a submenu containing the following commands:	
	EtherNet/IP	Each command opens the corresponding Communication Mapping View , page 544 to which the device has been added. It lets you configure communication between I/O scanner services and devices for each protocol.
	Modbus TCP	

Command	Description
Configuration File	The menu is available only if you stored a file in the database, page 676. Commands are disabled while the file is open in a user session.
	Attach... Lets you add a configuration file to the database or replace the file that is stored by a different one. Use this command to save changes that cannot be saved automatically after editing an attached file by using the Open command. The file extension must be allowed (see <i>EcoStruxure Process Expert, Configuration and Installation Guide</i>) to replace the file.
	Open Opens the file that is specified under Configuration File in the PROPERTIES pane, page 676 with the program that is associated to the file type on the local computer. The file extension must be allowed (see <i>EcoStruxure Process Expert, Configuration and Installation Guide</i>) to open the file. If you make changes to the file and save them in the program in which the file is open, the file stored in the database is replaced and a confirmation message is shown. You may need to close the program. The command does not appear if no program is associated to the file type. NOTE: Saving changes automatically to the database is not supported by all programs (for example, those that perform a save as) nor all file types (for example, full application exchange files (.zef) and archived application files (.sta) because, once opened, these files are saved in the .stu file format).
	Save As... Lets you save to disk a copy of the file that is stored in the database. Use this command to open a file that you cannot open by using the Open command. The file extension must be allowed (see <i>EcoStruxure Process Expert, Configuration and Installation Guide</i>) to save the file.
	Delete Deletes the file from the database after you confirm the command.
Configure, page 591	<p>Opens the Control Participant, which allows you to configure the hardware of the entity, such as modules and drops. For non-M580 controllers and STB islands, you can also configure communication services.</p> <p>For PROFIBUS remote masters (PRMs), it allows you to configure decentralized peripherals (DPs) by using DTMs and/or GSD files, page 629.</p> <p>For M580 controllers, You can use only names that are unique within the same folder level. first performs an analysis, page 1090 of the controller and its dependencies before opening the Control Participant if no incorrect configurations are detected.</p> <p>Opening the Control Participant applies the settings that you configured in the Topology Explorer (for example, for IP addresses, ports, and services).</p>
Connect	<p>Switches the device type manager (DTM) to online mode, connecting it to the physical device.</p> <p>Connecting a gateway or device DTM also connects its parent DTM.</p>
Delete	<p>Deletes the entity (or the selection of entities) from the system after you confirm the command.</p> <p>Deleting an entity also deletes the associated data, such as mappings and device DDT variables. For certain entities, the device DDT variable is deleted only after you perform a build.</p> <p>NOTE: If you use the command from the PAC submenu of a module in a physical view, the local racks and drops of the controller are deleted.</p> <p>NOTE: For M580 safety controllers, it also deletes the associated M580 Control project, page 912.</p>

Command	Description	
Deploy	Opens a submenu with the following commands:	
	<table> <tr> <td>Configuration</td><td> <p>For certain devices that support configuration by using a device type manager (DTM), lets you connect to the corresponding physical device and deploy its DTM configuration.</p> <p>The command is not available if the IP address of the device is not configured, if Under Construction is enabled, page 676.</p> </td></tr> </table>	Configuration
Configuration	<p>For certain devices that support configuration by using a device type manager (DTM), lets you connect to the corresponding physical device and deploy its DTM configuration.</p> <p>The command is not available if the IP address of the device is not configured, if Under Construction is enabled, page 676.</p>	
Deploy Built Project , page 794	<p>Opens the Deploy Built Project dialog box, which lets you select a built Control Participant project and deploy it to the controller or controller simulator.</p> <p>It can also be used to remove refinement changes made online by using the Refine Online command.</p>	
Deploy Built Project (Supervision)	Lets you select a built Supervision Participant project and deploy it to the workstation, page 846.	
Deploy Changes / Undo Online Changes , page 794	Deploys to the controller or controller simulator, the differences between the deployed Control Participant project and the corresponding built Control Participant project.	
Deploy Changes (Supervision)	<p>Lets you deploy a limited number of changes, page 851 to the workstation.</p> <p>The command is available only if you already deployed the built Supervision Participant project to the workstation and if the number and type of changes are supported.</p>	
Deploy Data , page 824	<p>Opens the Deploy Data dialog box in which you can select a controller data backup file that was created by using the Back Up Data command. It lets you deploy the data contained in the file to the controller or controller simulator.</p> <p>The command is available only if you already deployed a built Control Participant project to the entity.</p>	
Diagnosis	<p>Attempts to connect to the device and opens the diagnostic screen of the device type manager (DTM).</p> <p>The command is available only for devices whose DTM supports the diagnostic functionality and while the DTM is open. An IP address must be configured for the device.</p> <p>NOTE: For devices or modules that manage a fieldbus, the command is available in the Fieldbus Editor, page 553.</p>	
Disconnect	<p>Switches the device type manager (DTM) to offline mode, disconnecting it from the physical device.</p> <p>Disconnecting a DTM also disconnects its child DTMs.</p>	
Edit Fieldbus	<p>Opens the corresponding Fieldbus Editor, page 553, which lets you add secondary devices to a master gateway and configure them.</p> <p>The command is available only for devices or modules that manage a fieldbus except for devices of the EPE Managed Devices product range, page 516.</p>	
Export > Topology , page 647	Opens the Export window, which allows you to export the entity and its folder hierarchy in the form of a topology export file (.sbk).	
Forgot Password	Opens the Forgot Password dialog box, which contains an authentication code that is required to reset the forgotten Controller, page 810 or functional safety, page 925 password.	
Load Data From Device	<p>For certain devices that support configuration by using a device type manager (DTM), lets you connect to the corresponding physical device, upload from the device its DTM configuration, and update its I/O profile with this data.</p> <p>The command is not available if the IP address of the device is not configured or if its Under Construction attribute is enabled. It is not available for devices of the EPE Managed Devices product range, page 516.</p>	

Command	Description
Local Slaves	Opens a submenu with the following commands for each local slave, page 641 that exists in a controller. The command is available only on M580 controller and BMENOC03•1 modules.
	Open
	Connect
	Disconnect
Manage Password	<p>Opens the Manage Password dialog box, which lets you set and change the following passwords:</p> <ul style="list-style-type: none"> The Controller, page 808 and/or the functional safety, page 925 password. <p>Each command is available only if the corresponding property is enabled in the properties of the controller.</p> <ul style="list-style-type: none"> The Simulator, page 808 password for deployment and execution operations performed on a simulator instance. <p>If you created more than one Control Expert service, you must select one from a list.</p> <p>The command is available only if the Simulator password property of the workstation is set to Enabled.</p> <p>To use it for a given Control Expert service, the corresponding simulator instance must be running.</p>
Move Local Slave To	<p>The command appears only for EtherNet/IP devices that are created in the Local Slaves folder as a result of an import operation.</p> <p>Opens a menu that lists the controller modules and BMENOC03•1 modules of existing M580 controllers that have a part number that matches the device name of the EtherNet/IP device DTM (for example, BMEP582040).</p> <p>If a DTM reference, page 709 was selected for such a module, the module appears in the menu only if the device version also matches that of the EtherNet/IP device (for example, 3.10).</p> <p>The existence of a local slave, page 641 is irrelevant.</p> <p>If a module that appears in the menu has the same IP address as the EtherNet/IP device, it is shown separately in first position.</p> <p>Clicking a module in the list adds a local slave for the module, which has the same configuration as the EtherNet/IP device. Then, the device is automatically deleted.</p>
Open	<p>Opens the device type manager (DTM) for the selected device.</p> <p>The command is available only for devices that support configuration by using a DTM, page 629 (except for devices of the EPE Managed Devices product range, page 516) and requires that the DTM that supports the part number and version is available in the local DTM Hardware Catalog. If a later version of the DTM is available, opening the DTM updates it to this version.</p> <p>NOTE: For devices or modules that manage a fieldbus, the command is available in the Fieldbus Editor, page 553.</p>
Open Basic Diagnostic	<p>Opens the basic diagnostic window, page 558 for the selected device.</p> <p>The command is available only for devices of the Altivar Variable Speed Drives product range, page 516.</p>
Open Rack Viewer	<p>Opens the Rack Viewer, page 559, which lets you view the configuration of a local rack or drop. If system monitoring is active, it opens inside the Topology Explorer the embedded Web pages of modules that support this functionality.</p> <p>The command is available only for M580 modules and racks.</p>

Command	Description
Open Web Browser	<p>By default, lets you view in the default Web browser the embedded Web page of a module or automation device with the address <code>http://<IPv4></code>, where <code><IPv4></code> is the IP address of the module/device (for example, the embedded Web page for tuning or diagnostics).</p> <p>If you enter a URL in the Web Page Link attribute of the module or device, if available, it opens the external Web page instead. An Internet connection is required.</p> <p>For details on software requirements, the availability of embedded Web pages, and the services they provide, refer to the <i>Web pages</i> topic in the help of the corresponding module or device.</p> <p>The command is available only for modules and automation devices that support the functionality.</p>
Open With	<p>Opens a submenu containing a list of views that are available for the selected object (for example, a physical view or the Logical Network View). Selecting a view from the list shows or opens it and selects the corresponding object.</p> <p>The available views depend on the type of object.</p> <p>To open a physical view, the entity must appear in the view.</p>
Physical Connections , page 617	<p>Opens the Physical Connections dialog box, which allows you to connect the controller and communication modules of topological entities that support to logical networks, page 576 of the topology to model physical network connections.</p> <p>You can also use the command to view existing connections of modules and disconnect them from these networks.</p> <p>The command is available only for PROFIBUS Remote Masters (PRMs) and controllers of the M340 and Quantum platforms.</p>
Re-Deploy Last Project , page 794	<p>Lets you deploy once more the entire built Control Participant project that you last deployed to the controller or controller simulator.</p> <p>The command is available only if you previously used the Deploy Built Project command on the entity.</p>
Refine Online , page 855	<p>Lets you modify the deployed Control Participant project online.</p> <p>The command is available only for a controller or controller simulator (workstation).</p>
Replace Template , page 220	<p>Opens the Replace Device Template dialog box, which allows you to replaces the template that is used by the device with another template that meets the criteria, page 603 and that is available in the Global Templates Library on the system server.</p> <p>The command is available only for Modbus TCP and EtherNet/IP devices of the EPE Managed Devices product range, page 516.</p>
Run Supervision Client (Supervision)	<p>Launches the Plant SCADA runtime locally with the corresponding Supervision project after you confirm the command.</p> <p>The command is available only if the following conditions are fulfilled:</p> <ul style="list-style-type: none"> The Network Interface Card (NIC) of the workstation is configured with the IP address of the local computer. The build state of the Supervision executable that is mapped to the workstation is Built or Out Of Date. <p>For the Supervision project to launch in runtime (see <i>EcoStruxure Process Expert, Supervision Participant Services, User Guide</i>), some actions must be performed first, such as deploying the executable.</p> <p>If the runtime is already launched, you can restart it by using the command.</p> <p>The runtime is not launched if the Supervision Participant is being used by EcoStruxure Process Expert (for example, if a Supervision project is being refined or its executable built).</p>
Save Control Project As...	<p>Opens a Save Control Project As... dialog box, which lets you save to disk a copy of the controller configuration project (.stu).</p> <p>The command is not available for M580 controllers.</p> <p>NOTE: If an application password is set, page 692 for the Participant project, the project is saved with file encryption disabled.</p>
Show in Active Views	<p>Selects and shows the entity in the other windows and panes that are open (for example, in the topological folder structure of the SYSTEM PROJECT pane or in the LOGICAL NETWORK VIEW).</p>

Command	Description
Start , page 855	Starts the execution of the Participant project by the controller or controller simulator.
Stop , page 855	Stops the execution of the Participant project by the controller or controller simulator.
Sync (Primary→Standby) , page 866	Lets you synchronize the standby controller with the primary one. The command is available only for redundant controllers.
Synchronize Changes	Applies changes, page 590 you made in the properties of an M580 controller in the Topology Explorer to the controller configuration in the Control Participant, and vice versa.
Update Project , page 855	Updates the logical Control Participant project that is associated to the deployed Control Participant project with the changes that you made to the deployed project online by using the Refine Online command. For controllers, it also requires that you update the controller configuration with hardware changes that you made during online refinement.
Update Template , page 622	Opens the Update Device Template dialog box, which allows you to update the templates that are used by the entity with the highest version available in the Global Templates Library on the system server. If the entity is composed of subcomponents or contains services (such as workstations), a submenu may open allowing you to update only a specific template or the templates used by the entire entity.

Creating and Configuring Logical Networks

Overview

You can create logical networks either:

- From the **Logical Networks** node of the **SYSTEM PROJECT** pane, page 532. This creates a new logical network with default settings.
- From the interface of an entity, page 679 in the topology, which manages at least one IP address. This assigns the associated port of the entity to the newly created logical network, which is shown in the **SYSTEM PROJECT** pane.
- By using the **Create Logical Network** button in the toolbar of the **LOGICAL NETWORK VIEW**.

By using the second or third method, you can configure the logical network as you create it.

When you add, page 591 certain modules to a controller and apply changes, a new logical network may be created and the interface of one or more modules assigned to it.

NOTE: When you create a logical network and assign to it the interface of an entity, properties of the interface related to the logical network are automatically configured. For a description of the properties of entities, refer to the topic describing the **PROPERTIES** pane, page 673.

Creating Logical Networks from the SYSTEM PROJECT Pane

Step	Action
1	Right-click the Logical Networks node in the SYSTEM PROJECT pane.
2	Click New Logical Network . Result: A new logical network is created with properties set to default values, page 577.

Creating Logical Networks from the Interface of an Entity

Step	Action
1	Select the entity in one of the views (for example, the SYSTEM PROJECT pane or a physical view).
2	In the CONFIGURATION tab of the PROPERTIES pane, click the Logical Network list of under an IP Address . Result: The list opens showing existing logical networks.
3	Click New . Result: The Create Logical Network dialog box opens.
4	Edit the name if necessary and configure the properties, page 577 of the logical network.
5	Click OK . Result: <ul style="list-style-type: none"> • The logical network is created and appears in the SYSTEM PROJECT pane. • The next available IP address on the newly created logical network is assigned to the corresponding ports. NOTE: If not enough IP addresses are available on the logical network, the logical network is created but no IP address is assigned.

Logical Network Actions

For a description of the commands that are available for logical networks, refer to the topic describing their context menu, page 532 in the **SYSTEM PROJECT** pane.

Logical Network Properties

The following table describes the attributes that appear in the **PROPERTIES** pane when you select a logical network.

Attributes	Default value	Description
Name	–	Editable name of the logical network. You can only use names that are unique within the topology.
Subnet Address	192.168.0.0	IP address whose host ID is 0 (based on the subnet mask) to define the address of the subnet. NOTE: Modifying this attribute does not modify the IP addresses of entity interfaces that belong to this logical network. You must reassign IP addresses, page 541.
Subnet Mask	255.255.255.0 (class C)	Lets you define the portion of the IP address that represents the subnet address and, consequently, the range of host addresses on this subnet. NOTE: Modifying this attribute does not modify the IP addresses of entity interfaces that belong to this logical network if the resulting logical network does not have sufficient addresses. You must reassign IP addresses, page 541.
Gateway Address	192.168.0.254	IP address of the default gateway of the network.

Representation of Entities and Ports

Representation of Entities in Physical Views

Overview

Each entity that you add to a physical view, page 536 is represented as one or more graphical objects.

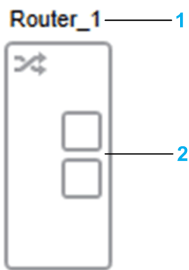
Controllers of platforms other than M580, STB islands, and PROFIBUS Remote Masters are represented by a single object, independent of their configuration.

This topic describes how various types of objects are represented in a physical view.

NOTE: For information on the diagnostic icons that appear on devices, refer to topic describing system monitoring, page 1107.

Devices

The following figure shows an example of a device that was created by using a managed switch of the **ConneXium Switches and Hubs** product range.



Item	Description
1	You can show/hide the following attributes of devices by using the Display menu, page 536: <ul style="list-style-type: none">• The name• IP addresses• Logical networks
2	Ports, page 582

Devices of the **EPE Managed Devices** product range are identified by a double outline as shown in the figure below.



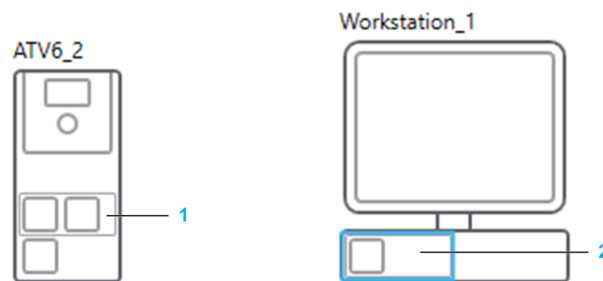
The following figure shows a device whose **Under Construction** attribute is set to *TRUE* (selected). The device is shown grayed out.



Entities Containing Modules

Entities that contain modules that have Ethernet connectivity feature a distinct graphical area, which represents the module and that you can select individually. It lets you view properties and open a context menu that are specific to the module.

The following figure shows an example of entities that contain a module.

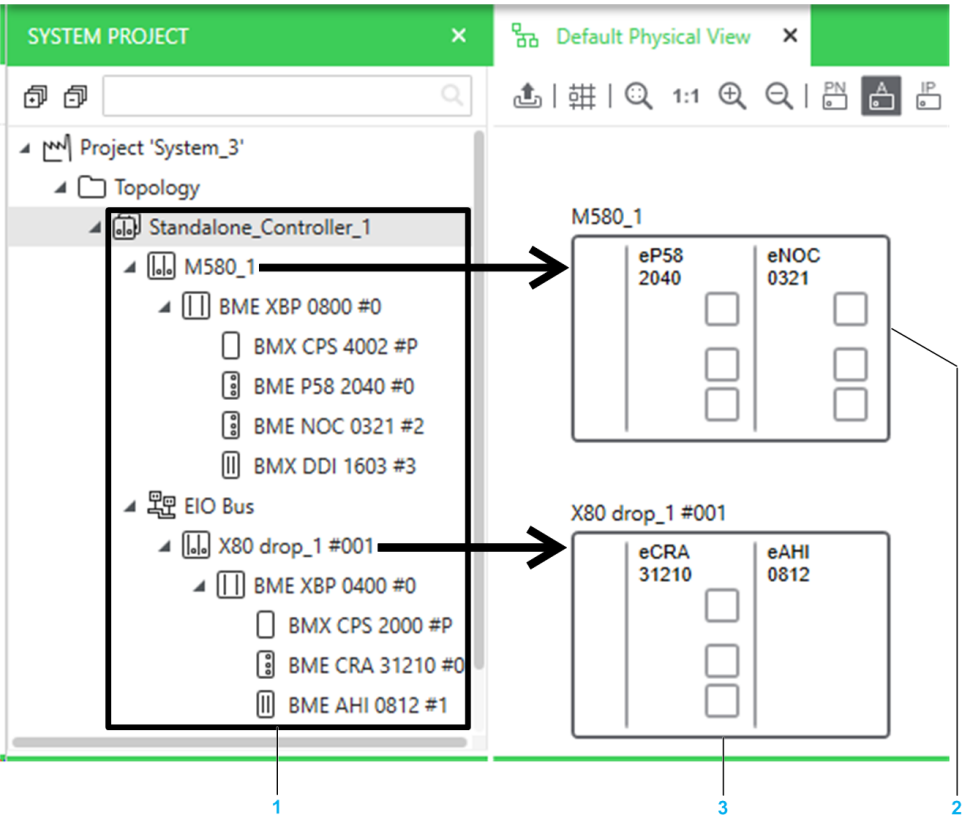


Item	Description
1	Graphical area that represents the communication card, page 686 of a drive. The outline of the area is highlighted when you hover over it with the pointer.
2	Graphical area that represents the Network Interface Card (NIC) of the workstation. Click the empty space inside the area to make it active and view the properties of the NIC. Right-click the area to open a context menu.

M580 Controllers

In a physical view, the local rack and drops of standalone and redundant M580 controllers are represented as distinct objects.

The following figure shows an example of how an M580 controller is represented in a physical view.



Item	Description
1	The controller represented by the <i>Standalone_Controller_1</i> node in the SYSTEM PROJECT pane, page 527.
2	The local rack.
3	The drop.

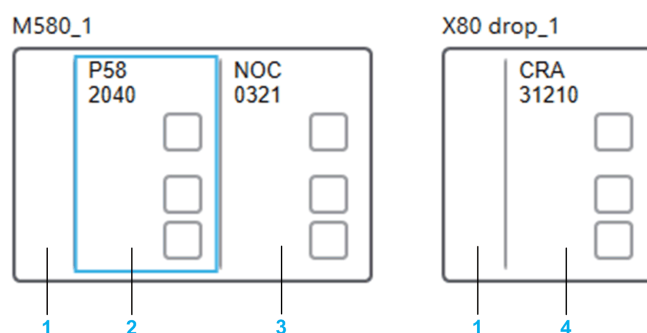
A rack or drop is composed of several graphical areas that you can select individually. Areas represent modules that have Ethernet connectivity (for example, the communication module in a local main rack, the adapter module in a remote rack). The position of modules shown in the physical view may differ from their position on the physical backplane. Modules let you open a menu containing commands that are specific to the selected module.

To view the modules and their position, open the rack or drop by using the **Rack Viewer**, page 559.

If you change the configuration, page 591, the representation of the controller is updated in the physical view depending on the type of module (for example, extended racks and their modules are not shown).

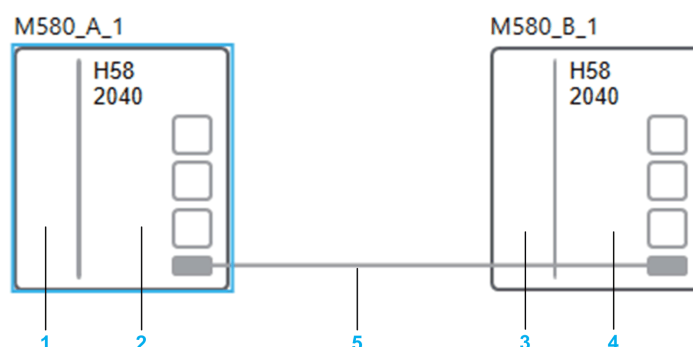
If you add a drop to a controller whose local rack already appears in a physical view, the drop is automatically added to the same physical view. It overlaps the local rack and you can reposition it.

The following figure shows an example of the local rack and drop that are displayed when you drag a standalone M580 controller node to a physical view. The drop was manually repositioned.



Item	Description
1	Handle, which lets you move the rack and open a context menu.
2	Area that represents the controller module. The highlighted outline indicates that only the controller module is selected. Right-click it to open the context menu of the module and the controller.
3	Area that represents a communication module that was added by configuring the controller.
4	Area that represents the adapter module of the drop.

The following figure shows an example of the two local racks and the high-speed cable link that are displayed when you drag a redundant M580 controller node to a physical view. If drops are configured, they are displayed as well.



Item	Description
1	Handle of local rack A. The rack is shown selected.
2	Controller module.
3	Handle of local rack B. The rack is automatically positioned at a distance and you can move either rack individually.
4	Controller module.
5	High-speed cable link represented by a solid line. You can enter a label or modify the routing, page 608, but you cannot delete the link.

Representation of Ports of Entities in Physical Views

Overview

Most entities have at least one port that you can connect to the port of another entity.

Only ports that use a communication protocol are shown (for example, Modbus TCP or EtherNet/IP).

This topic describes how ports are represented in a physical view.

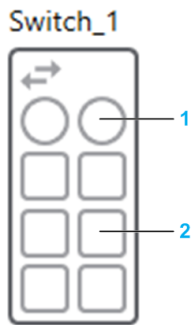
For details on the representation of the status of ports and on port actions, refer to the topic describing how to connect ports of devices, page 608.

NOTE: You can identify port types graphically, page 536.

NOTE: Select a port to view and/or modify its properties by using the **PROPERTIES** pane.

Types of Connectors

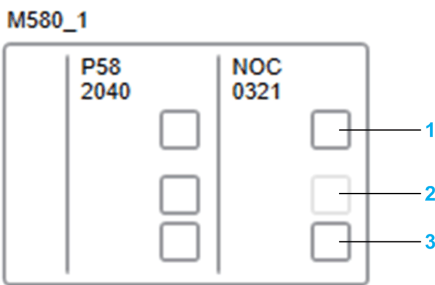
The following figure shows a device that features ports having two different types of connectors.



Item	Description
1	Fiber optic connectors are represented by a circle.
2	RJ45 connectors are represented by a square.

Ports of Modules of Controllers

The following figure shows an example of a local rack containing a controller and a communication module.



Item	Description
1	Service port
2	ETH1 shown disabled
3	ETH2

The backplane port is not shown.

Ports of Communication Cards

Ports of the following cards are managed separately from the entity:

- Network interface cards (NICs)
- Communication cards of Altivar Process devices

The following figure shows an example of the representation of ports of the NICs of a workstation.

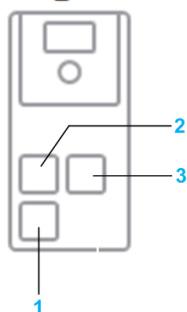
OperatorWorkstation_1



Item	Description
1	Area where Ethernet communication ports can be shown depending on the configuration of the device.
2	Port 1 of NIC1.
3	Redundant port 2 of NIC1.
4	Port 1 of NIC2.

The following figure shows an example of the representation of ports of an ATV6xx device of the **Altivar Variable Speed Drives** product range.

ATV_1



Item	Description
1	Embedded port 1.
2	Port 1 of the communication card if the card is present.
3	Port 2 of the communication card.

NOTE: Ports of devices that use non-Ethernet communication protocols are not shown.

Configuring Topological Entities

Selecting Topological Templates

Overview

The **Select Template** dialog box opens if more than one topological template is available to model an entity, page 565, a component, or a service that you create or modify.

It allows you to select the topological template and version that you want to use among the compatible ones that are available in the Foundation Library.

The template selection is available for the following entities, components, and services:

- Modbus TCP and EtherNet/IP devices of the **EPE Managed Devices** product range.
- Network Interface Cards (NICs) of workstations of the **Workstations & Servers** product type.
- *Control*, *Supervision*, and *OFS* services of workstations.

For other entities, the default template selection is automatically applied.

NOTE: The **Update Template** command, page 622 is available in the context menu of entities after you create them. For certain devices, the **Replace Template** command is also available.

Default Template Selection

By default, EcoStruxure Process Expert selects the topological template that has the highest version number with the usability state (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) **Approved** or **Deprecated** among those that have a matching **Hardware Reference** attribute.

Template Selection for Redundant Controllers

When you select a topological template for a module in one of the local or remote racks of a redundant controller, the same template is used for the module in the counterpart rack.

Select Template Dialog Box

For a description of the **Select Template** dialog box, refer to the topic describing the selection of topological device templates, page 603.

Selecting Topological Templates

To select a template, in the **Select Template** dialog box, click the template and then, **OK**. The corresponding device is created in the topology of the system.

Using Configuration Change On The Fly (CCOTF)

Before Performing CCOTF Modifications

The CCOTF functionality of the Control Participant is enabled by the *Online modification in RUN or STOP* parameter of the controller module when you create a controller of the M580 platform in the **Topology Explorer**.

It allows you to deploy compatible hardware configuration and parameter changes to the controller while it is running without impacting other modules or ongoing tasks.

Before performing a CCOTF-compatible modification, refer to *Modicon M580, Change Configuration on the Fly, User Guide*.

CCOTF With EcoStruxure Process Expert

In the EcoStruxure Process Expert **Topology Explorer**, CCOTF can be used when you configure a controller of the M580 platform offline with the Control Participant if you already deployed a built Control project to the controller.

When CCOTF is enabled, EcoStruxure Process Expert alerts you if a configuration or parameter change that you are about to make will require stopping the controller when you deploy changes to the controller during the deployment stage because this type of change must be built offline.

NOTE: If you configure drops of the S908 architecture or change an existing drop of this architecture during configuration of the controller offline, you must use the **Build All**, page 781 and **Deploy Built Project** commands thereafter, which require stopping the controller. To perform CCOTF-compatible configurations of drops of the S908 architecture, use the **Refine Online** command.

NOTE: CCOTF is not supported for controllers of the M340 and Quantum platforms.

Enabling and Disabling CCOTF

You can disable or enable the *Online modification in RUN or STOP* parameter in the Control Participant **Configuration** section of the controller module when configuring offline.

Disabling this parameter disables the CCOTF functionality.

Enabling or disabling the parameter requires that you deploy the project to the controller by using the **Deploy Built Project** command, which stops the controller.

Configuration Changes not Compatible With CCOTF

The following changes are incompatible with CCOTF and require stopping the controller:

- Changes made in the properties of the **Topology Explorer**.
- The following types of changes made in the Control Participant:
 - A CCOTF-incompatible change that is saved during controller configuration.

For example, adding a BMENOC**** communication module in the local rack of an M580 controller.

- A series of CCOTF changes of the same type that exceed the number of allowed consecutive changes in a same transaction.

For example, adding five digital input modules to the local rack of an M580 controller (the limit for CCOTF is four modules).

NOTE: Adding a drop with a BMECRA****/BMXCRA**** adapter module to an M580 controller is incompatible with CCOTF because you must modify the IP address and/or the logical network of the adapter module in the properties of the **Topology Explorer** after the drop is added.

For details on the other limitations of CCOTF, refer to the Control Participant help on CCOTF with the controller.

Working Principle

When you perform a series of CCOTF-compatible changes in the Control Participant of the **Topology Explorer**, you can build the executable of the associated Control project in the **Project Explorer** between changes. The changes are considered to be part of the same transaction. A transaction is executed and the CCOTF counter is reset if you use the **Deploy Changes/Undo Online Changes** or **Deploy Built Project** command on the controller.

If a change that you make in the Control Participant requires stopping the controller, a dialog box opens giving you the choice to proceed with the change or cancel it. The following table describes the impact of either choice.

Action selected by the user	Impact on CCOTF	Impact on continued configuration changes
Yes	<p>The configuration change is performed. If you save the change when closing the Participant configuration window, the Deploy Changes/Undo Online Changes command is disabled and deploying the change by using the Deploy Built Project command requires stopping the controller.</p> <p>NOTE: Closing the Participant configuration window without saving discards the change and CCOTF remains available.</p> <p>NOTE: Undoing the CCOTF-incompatible configuration change does not revoke the controller stop if you save your changes.</p>	<p>You are not informed anymore if you make CCOTF-incompatible configuration changes until you deploy to the controller.</p> <p>The Build command in the context menu of the executable is disabled. Only Build All can be used.</p>
Cancel	The configuration change is discarded and you can continue with the configuration of the controller.	You continue to be informed about CCOTF-incompatible configuration changes.

CCOTF in the System Engineering Life Cycle

The following table summarizes the main impact of CCOTF on the other stages of the system engineering life cycle. For details, refer to the description of each stage.

Stage	Impact of CCOTF
Refinement (offline)	If you perform a change in the Control project, page 446 that is not CCOTF-compatible, you are alerted in advance that it will require stopping the controller when you deploy the change.
Build	CCOTF-compatible changes related to the controller configuration can be built, page 781 by using the Build command.
Deployment	<p>CCOTF-compatible changes related to the topology can be deployed, page 830 by using the Deploy Changes/Undo Online Changes command without stopping the controller.</p> <p>If you made a CCOTF-incompatible change, only the Deploy Built Project command is available, which requires stopping the controller.</p>
Execution	<p>You must update the project, page 897 with configuration changes that you made online to use the Deploy Changes/Undo Online Changes command.</p> <p>NOTE: Adding a drop with a BMXCRA..... adapter module to an M580 controller is incompatible with CCOTF.</p>

Setting the Webuser Password for M580 Controllers

Overview

The **Webuser Password** attribute in the **System Project** section of the **User Preferences** window, page 524 allows you to change the **Web Diagnostic/Data Storage** password for the M580 controllers of the system topology.

For details on the **Web Diagnostic/Data Storage (Project > Properties > Project & Controller Protection > Data Storage** (or **Web Diagnostic/Data Storage**), refer to the topic describing data storage/Web protection (see *EcoStruxure™ Control Expert, Operating Modes*).

Scope of the Password Protection

The **Web Diagnostic/Data Storage** password helps prevent unauthorized access to Web diagnostics and/or to the data storage zone of the SD memory card of an M580 controller, if a valid memory card is inserted in the controller module.

By default, the password is *webuser* and it must be changed if the topology contains an M580 controller as of version 4.00 or greater; otherwise, an analysis of the topology detects an *incorrect configuration*, page 1091 and you cannot configure the M580 controller.

You must change the password the first time you open the **Rack Viewer**, page 559 for an M580 controller as of version 4.00 or greater while system monitoring is active. Thereafter, the password is not requested anymore until you change it, log out, or close the engineering client.

NOTE: The **Web Diagnostic/Data Storage** section in the properties of a control project in the Control Participant is disabled and the password can only be managed in the **User Preferences** window.

Password Requirements

The password must contain 8 to 16 characters with at least one character from each of the following categories:

- Uppercase characters from the classical Latin alphabet (A...Z)
- Lowercase characters from the classical Latin alphabet (a...z)
- Base-10 digits (0...9)
- Special characters (~, !, @, \$, %, ^, &, *, _, +, -, =, ` , |, \, (,), [,] , : , " , ' , < , > , { , } , ; , #)

Changing the Password

To change the **Webuser Password**, proceed as follows.

Step	Action
1	Open the User Preferences window in the Topology Explorer of the system and select the System Project section.
2	Click Define Password... next to the Webuser Password attribute.
3	In the dialog box that opens, enter the password that is set (<i>webuser</i> , by default), enter the new password, confirm it, and click OK . Result: If the passwords are valid, a confirmation message is shown and the new password is set.
4	Configure each M580 controller of the topology to apply the change to their configuration. NOTE: If the controller is open in the Control Participant while you change the password, use the Synchronize Changes command, page 590 or close the Control Participant and save changes.

Recovering a Forgotten Password

To receive a temporary password, proceed as follows.

Step	Action
1	In the dialog box prompting you to enter the password that is set, click I forgot my password . Result: An unlock code is displayed in the dialog box.
2	Contact your local Schneider Electric service representative and provide the unlock code. Result: You receive a temporary password.
3	Enter the temporary password in the Password text box.
4	Follow the procedure describing how to change the password and use the temporary password in place of the password that is set.

Synchronizing Changes

Overview

The **Synchronize Changes** command lets you apply to the controller configuration changes that you make to or from the properties of an M580 controller in the **Topology Explorer** and vice versa, in a single step, while the Control Participant window is open.

When you use the command, changes made to the configuration in the Control Participant are saved.

Change Indicator

An asterisk is displayed next to the name of the controller in the **SYSTEM PROJECT** pane to indicate that a modification to its properties in the **Topology Explorer** was not applied to or from the configuration in the Control Participant and/or vice versa.

The asterisk appears only if a modification is made while the Control Participant is open and it is only visible to the user who made the modification.

The modification that you make in the **Topology Explorer** must have an impact on the controller configuration. Changing the **Under Construction** attribute, for example, does not display the change indicator.

When the asterisk is displayed and the Control Participant is open, the **Synchronize Changes** command is available.

NOTE: The asterisk also appears if you change Control Participant data that is not managed in the **Topology Explorer**, page 519 (for example, changes in the **Programs** directory).

Working Principle

The following table presents the actions that are performed if you use the **Synchronize Changes** command and changes were made to a controller in both the **Topology Explorer** and the configuration in the Control Participant.

Step	Description
1	The controller is analyzed, page 1090. If incorrect configurations are detected, you can choose to ignore them and proceed with the synchronization or cancel it. NOTE: Proceeding further in the system engineering life cycle when incorrect configurations are present may prevent you from building in the Control executable.
2	If you proceed, changes to the controller configuration in the Control Participant are applied to the properties of the controller in the Topology Explorer .
3	Changes to the properties of the controller are applied to the controller configuration.
4	The configuration is saved.
5	When the process is completed, the Synchronize Changes command is disabled. The Control Participant remains open.

Canceling the Synchronize Changes Command

If you close the Control Participant after using the **Synchronize Changes** command and discard changes before the save operation completes, the synchronization of changes is canceled.

Configuring Controllers

Overview

The controller topological entity models a physical controller. It acts as an engine for executing Control projects that are created in the **Projects Explorer** and deployed.

You configure controllers in the Control Participant. Closing the Participant transfers the configuration changes to the **Topology Explorer** where the controller representation is updated.

If a topological template for a module is not available in the Foundation Library, the *\$UnknownModule* template is used. Hardware modules that are modeled with this template do not appear in the **Hardware Mapping Editor**.

NOTE: For additional information on configuring M580 safety controllers, refer to the topic describing the engineering of M580 safety systems, page 908.

NOTE: Before using another command for an entity, wait for the task in progress to complete. However, the same command can be executed on separate entities simultaneously.

Simultaneous Configuration of Controllers

Depending on the controller platform, the maximum number of controllers that you can configure simultaneously on each computer is defined as follows:

- M580 controllers: Four per **Topology Explorer** instance or system with a maximum of four **Topology Explorer** instances per engineering client.
- Other controller platforms: The number depends on the value configured for **Control Participant Max Instances** in the **System Server Configuration Wizard** and the number of other Control Participant windows (see *EcoStruxure Process Expert, Installation and Configuration Guide*) open at the same time.

Device DDTs of Controllers

If you add a module to a controller that uses a Device Type Manager (DTM) (for example, an M580 controller module, a BMENOC•••• communication module, or a HART analog I/O module), corresponding device Derived Data Type (DDT) is automatically created. This type of variable exists only in the configuration that you can open from the **Topology Explorer (Configure command)** and is integrated to the Control project during build.

However, you can already create certain device DDTs, page 455 in the Control project during refinement.

Configuring Redundant Controllers

For redundant M580 controllers, you can open the configuration from the controller node or one of the two local racks.

When you modify the properties of a module in one of the local racks or modify the configuration of a rack (for example, add or delete a module), the modifications are automatically applied to the counterpart module in the other rack or the counterpart rack. Rules that are specific to the configuration of redundant controllers may apply.

For example, if you disable the service port of the controller module of rack M580_A_1, the service port of the controller module of rack M580_B_1 is also disabled.

Creating a Controller

To create a controller, refer to the topic describing how to add topological entities to a system, page 565.

Controller Actions

For a description of actions that you can perform on controllers, refer to the topic describing entity context menu commands, page 570.

Controller Properties

To configure the properties of a controller or one of its modules, select it and use the **PROPERTIES** pane, page 673. Other properties can be configured by using the Control Participant.

To apply changes to properties made in the **PROPERTIES** pane, either use the **Configure** command or the **Synchronize Changes** command, page 590 if the controller is open in the Participant while the changes are made.

NOTE: If the controller is acting as server for peer to peer communication, page 776, the full or partial name of the controller is used to create the name of the DTM when it is added during build, page 776.

Configuring the Controller

To configure an M580 controller, the analysis must not detect incorrect configurations, page 1091 for the controller and its dependencies.

To configure a controller, proceed as follows.

Step	Action
1	<p>Select the controller and, if required, configure the properties of the controller and its dependencies to enable the configuration in the PROPERTIES pane (for example, for an M580 controller, IP addresses and communication security settings).</p> <p>NOTE: For M580 controllers with firmware version 4.00 and greater, you must set a Web Diagnostic/Data Storage password, page 588 in the User Preferences window that opens from the toolbar of the Topology Explorer.</p>
2	<p>Right-click the controller and select Configure.</p> <p>Result: The values of controller properties configured in the PROPERTIES pane are applied to the controller configuration and the Control Participant opens.</p> <p>NOTE: Click the Full Screen button in the toolbar of the Topology Explorer to maximize the window.</p>
3	<p>Configure the controller.</p> <p>For more information on configuring the controller by using the Control Participant, see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>.</p>
4	<p>Depending on the controller platform, save your changes as follows:</p> <ul style="list-style-type: none"> M580: Click Synchronize Changes, page 590 in the controller context menu or close the Control Participant window and save changes. M340 and Quantum: Click the Save button in the title bar of the Control Participant window. <p>Result: The representation, page 578 of the controller and, if applicable, its properties are updated in the Topology Explorer.</p> <p>NOTE: If EcoStruxure Process Expert detects incorrect configurations, it displays a notification and you can modify the controller configuration. You can also ignore incorrect configurations and save changes. However, proceeding further in the system engineering life cycle when incorrect configurations are present may prevent you from building in the Control executable.</p>
5	<p>Select the controller and/or a module in the Topology Explorer and configure its properties, page 673.</p>

IP Address Management

The default IP addresses of modules that you add to the controller may be replaced when you save changes because the consistency of communication parameters of the topology is managed by the **Topology Explorer**.

Adding a module that manages an IP address may create an additional logical network, page 576 when you save changes. This is the case, for example, if the IP address of the embedded interface of the module must belong to a different logical network than the IP address of the controller module.

Changing the Controller Configuration

To change the hardware configuration of a controller, follow the procedure to configure a controller and make changes in the Control Participant.

If the controller is mapped to, at least, one executable, saving changes to communication functions in the Control Participant deletes the existing communication mapping, page 725 and hardware mapping, page 747 of this executable.

If this is the case, select one of the following in the confirmation dialog box:

- **OK:** To save the change to the controller configuration and delete the associated communication and hardware mappings.
- **Cancel:** To keep the Control Participant window open without discarding changes.

For more information, refer to Configuring the Controller (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

Impact of Controller Module Replacement on the Topology

When you save or synchronize changes after replacing an M580 controller module, the necessary changes are automatically made.

The table describes some of the impacts of a controller module replacement on the topology depending on the replacement scenario.

Object or data	From stand-alone to stand-alone	From redundant to redundant	From standalone to redundant	From redundant to standalone
Controller name	No change.		Controllers that have a default name are renamed <i>Redundant_Controller_x</i> where <i>x</i> is an integer suffix.	Controllers that have a default name are renamed <i>Standalone_Controller_x</i> .
Local racks			The name is replaced by <i>M580_A_x</i> . An identical local rack B is added with name <i>M580_B_x</i> .	Local rack B is deleted. The name of the former local rack A is kept.
IP addresses of modules			IP addresses may be reassigned depending on the availability of specific addresses in the same logical network (for example, MainIP+1).	No change.
Scanner services	The Communication Mapping Views are updated depending on the scanner services of the controller module reference that you selected.			
			Scanner services of modules in local rack A are used.	Scanner services of modules in former local rack A are used.
DHCP servers	No change.		Modules in local rack A become the DHCP server. The name of the DHCP server is updated in clients.	Modules in former local rack A become the DHCP server. The name of the DHCP server is updated in clients.
Modules in local racks			Modules present in local rack A are automatically added to local rack B.	Modules present in former local rack A are kept.
Physical links			Only the high-speed cable link with local rack B is created.	Physical links to and from local rack B are removed.

Configuring the Controller I/O Scanner Parameters

To use the I/O scanner of the controller and define communication channels between the controller and/or communication modules and devices connected to the same logical network, configure the read/write zones once the configuration process to transfer the controller configuration to the **Topology Explorer** is completed.

The related parameters of the controller/communication modules are configured and the data is exposed in the **Communication Mapping Editor** window during the communication mapping, page 725.

If you configure overlapping client and server memory areas for a controller and map variables to these overlapping addresses, build does not complete successfully and a message informs you.

The following settings are an example of values entered in the **Properties** window of a controller that result in overlapping memory areas:

- **ServerMemoryStart:** 1
- **ServerMemoryLength:** 100
- **ClientMemoryStart:** 51
- **ClientMemoryLength:** 150

For more information on configuring parameters for I/O scanning, refer to the Control Participant help.

NOTE: If you reduce the value of the size, page 725 that is reserved in the client and/or server memory while peer to peer communication channels exist in the controller and you already built the associated logical Control Participant project, ensure that the new register size fulfills the requirements of the project. You cannot build the project if the free memory value becomes negative.

NOTE: If you modify a parameter of the client and/or server memory reserved for communication channels and you already deployed the associated built Control Participant project, you can apply the change to the engine only by using the **Deploy Built Project**, page 834 command, which requires stopping the controller. You cannot use the **Deploy changes** command.

To configure the I/O scanner parameters of the controller, proceed as follows.

Step	Action
1	In the Topology Explorer , select the module of the controller that you want to use for I/O scanning.
2	In the memory categories, page 685 of the PROPERTIES pane (CONFIGURATION tab), enter a value in the appropriate fields.

NOTE: For controllers of the Quantum platform configured with a 140NOC.... communication module and acting as client, configure the memory size of the communication module (see *EcoStruxure Process Expert, Control Participant Services, User Guide*) by using the Control Participant.

Controller Representation

Refer to the topics describing the representation of controllers in physical views, page 536 and the **SYSTEM PROJECT** pane, page 527 for more information.

Deleting a Controller

When you delete a controller, EcoStruxure Process Expert notifies you if it detects mappings.

After you confirm the command, the following are deleted:

- Physical links.
- The controller entity.
- The corresponding Control Participant configuration project, including:
 - Mapping information.
 - Peer to peer communication channels, page 725.
 - Deployed Control Participant projects related to the engine.

To delete the controller, proceed as follows.

Step	Action
1	Right-click the controller in the Topology Explorer and select Delete . Result: A dialog box opens.
2	Verify the information that is displayed and click OK . Result: The controller is deleted and a message is displayed in the notification panel. NOTE: For M580 safety controllers, it also deletes the associated M580 Control project, page 912.

Deleting Modules

To delete a module from the configuration immediately after closing the Control Participant window or using the **Synchronize Changes** command, you may need to fix incorrect configurations, page 1091 in the **Topology Explorer** first.

Otherwise, you cannot reopen the Control Participant to delete the module or close the Control Participant window to apply the deletion if the window is open.

Configuring Workstations

Overview

The workstation topological entity models the hardware (for example, network interface cards (NICs)) and software (for example, the Controller simulator, or Supervision servers and clients) that are installed on a computer.

Workstations can be operator stations, operation servers or engineering stations acting as engines for executing:

- Deployed Supervision projects.
- Control projects deployed to the Controller simulator, page 799.

You create workstations in the **Topology Explorer** and define their Network Interface Cards (NICs) and services in the **PROPERTIES** pane.

Creating a Workstation

To create a workstation, refer to the topic describing how to add topological entities to a system, page 565.

Creating Workstation Services

You can create the following services on a workstation by using the **SERVICES** tab of the **PROPERTIES** pane. Some services are configurable, page 708.

Service	Description
Control Services	<p>Creates a Control Expert service instance, which allows you to:</p> <ul style="list-style-type: none"> • Deploy, page 794 a built Control Participant project to the Controller simulator that is running on the computer. • Select the workstation/Control Expert service to which you deployed a Control project in the communication mapping, page 769. It lets you use it as data source (I/O device) for simulating Supervision runtime services. <p>You can create several Control Expert services per workstation. Each service corresponds to a Controller simulator instance that uses a unique port value that you must enter when you create a service.</p>
Supervision Service	<p>Creates a Supervision service instance.</p> <p>It lets you select the workstation in the service mapping, page 767 to deploy a Supervision project to it.</p> <p>Create the service in the workstations on which the I/O⁽¹⁾, Alarm, Trend, and Report servers as well as the Supervision clients are installed.</p> <p>You can create one Supervision service per workstation.</p>
OFS⁽²⁾	<p>Creates a protocol-independent OFS service instance.</p> <p>Create the service in the workstation on which the I/O server and OPC Factory Server or OPC UA Server Expert are installed. It is required to map the workstation to the <i>IOServer</i> service of the Supervision project in the service mapping, page 767 and to deploy the corresponding configuration files, page 788 to the workstation.</p> <p>You can create one OFS service per workstation.</p> <p>NOTE: The OFS service is not required to perform the service mapping with the <i>IOServer</i> service if the OPCUA Embedded protocol is selected, page 354 for the tag containers associated to the same I/O server.</p>
(1) Also requires an OFS service on the workstation.	
(2) Also requires a Supervision service on the workstation.	

Workstation Actions

For a description of actions that you can perform on workstations, refer to the topic describing entity context menu commands, page 570.

Workstation Properties

To configure the properties of a workstation or its NICs and services, select it and use the **PROPERTIES** pane, page 673.

Creating Network Interface Cards (NICs)

To create an NIC, proceed as follows.

Step	Action
1	Select the workstation in the Topology Explorer and click the Plus Sign (+) in the Communication Card category of the CONFIGURATION tab in the PROPERTIES pane. Result: An NIC is created.
2	Configure its properties, page 673.

NIC Actions

For a description of actions that you can perform on NICs, refer to the topic describing entity context menu commands, page 570.

NIC Properties

To configure the properties of an NIC, select the NIC or the workstation and use the **PROPERTIES** pane, page 673.

Configuring STB Islands

Overview

The **STB Island** topological entity of the **EPE Managed Devices** product range models a physical STB island that communicates with the controller through I/O scanning. It contains:

- Modules
- CANopen extensions
- HART devices

The STB island acts as a server.

You configure STB islands in the Control Participant. Closing the Participant transfers the configuration changes to the **Topology Explorer**.

If a topological template for a module is not available in the Foundation Library, the *\$UnknownModule* template is used. Hardware modules that are modeled with this template do not appear in the **Hardware Mapping Editor**.

Creating an STB Island

To create an STB island, refer to the topic describing how to add topological entities to a system, page 565.

STB Island Actions

For a description of actions that you can perform on STB islands, refer to the topic describing entity context menu commands, page 570.

STB Island Properties

To configure the properties of an STB island, select it and use the **PROPERTIES** pane, page 673. Some properties must be configured by using the Control Participant.

Configuring the STB Island

To configure the STB island, proceed as follows.

Step	Action
1	Right-click the STB island in the Topology Explorer and select Configure . Result: The Control Participant opens.
2	Configure the STB island. For more information on configuring the STB island by using the Control Participant, see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i> .
3	Click the close button in the toolbar of the Participant window and save changes. NOTE: If EcoStruxure Process Expert detects an incorrect configuration, it displays a notification, which allows you to change the configuration.
4	Select the STB island in the Topology Explorer and configure its properties, page 673.

Modifying STB Island Properties

Modifying the IP address or subnet mask of an STB island deletes the communication mapping, page 725 of this STB island.

If this is the case, select one of the following in the confirmation dialog box:

- **Yes:** To save your changes to the properties of the STB island and delete the associated communication mapping.
- **No:** To discard your changes.

Configuring Communication Between Controller and STB Island

Refer to *Communicating with STB Islands Through Modbus TCP*, page 940 for details about implementing communication by using the I/O scanner of the controller.

STB Island Representation

After you configure an STB island, save changes, and close the Control Participant, the representation of the entity in the **Topology Explorer** remains unchanged.

For more information, refer to the topics describing the representation of entities in physical views, page 536 and the **SYSTEM PROJECT** pane, page 527 .

Deleting an STB Island Entity

When you delete an STB island, EcoStruxure Process Expert notifies you if it detects mapping interface links with a device.

After you confirm the command, the following are deleted:

- Physical links with the STB island.
- The STB island entity.
- The associated STB island profile.

To delete the STB island, proceed as follows.

Step	Action
1	Right-click the STB island in the Topology Explorer and select Delete . Result: A notification dialog box opens.
2	Verify the information that is displayed and click OK to proceed. Result: The STB island is deleted and a message is displayed in the notification panel.

Configuring PROFIBUS Remote Masters and Decentralized Peripherals

Overview

The **PRM Profibus DP** topological entity of the **EPE Managed Devices** product range models PROFIBUS remote masters (PRMs) and their decentralized peripherals (DPs), which communicate with controllers by using I/O scanning.

You configure PRM and DPs from in the Control Participant by using the default device type managers (DTMs) or generic station description (GSD) files that are installed. Closing the Participant transfers the configuration changes to the **Topology Explorer**.

If a topological template for a DP is not available in the Foundation Library, the *\$GenericPBSlave* template is used. DPs that are modeled with this template use a generic mapping interface that is available in the **Hardware Mapping Editor**, page 747.

NOTE: You can install additional DTMs and GSD files (see *EcoStruxure Process Expert, Installation and Configuration Guide*) in the Control Participant.

NOTE: To configure a PRM gateway of the **Network Fieldbus Devices** product type, refer to the topic describing the **Fieldbus Editor**, page 553.

Creating a PRM Entity

To create a PRM entity, which will contain the PRM and DP configuration (devices), refer to the topic describing how to add topological entities to a system, page 565.

PRM Actions

For a description of actions that you can perform on PRMs, refer to the topic describing entity context menu commands, page 570.

PRM Properties

To configure the properties of a PRM, select it and use the **PROPERTIES** pane, page 673.


NOTE: The full or partial name of the PRM is also used to create the name of the DTM when it is added during build, page 776 but the name is not propagated to the PRM DTM (**Alias name**) in the Control Participant.

Simultaneous Configuration of PRMs

The maximum number of PRMs that you can configure simultaneously on each computer depends on the value configured for **Control Participant Max Instances** in the **System Server Configuration Wizard** and the number of other Control Participant windows (see *EcoStruxure Process Expert, Installation and Configuration Guide*) open at the same time.

Configuring the PRM and its DPs

To configure the PRM and its DPs, proceed as follows.

Step	Action
1	Select the PRM and configure Controller Family in the PROPERTIES pane (for example, for an M580 controller, IP addresses and communication security settings).
2	Right-click the PRM and select Configure . Result: The Control Participant opens.
3	Configure the PRM and its DPs by using the Control Participant. For more information, refer to the topic describing the configuration of the PRM and DPs (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>).
4	Click the save button in the toolbar of the Participant window.  Result: The configuration is transferred to the Topology Explorer . NOTE: If EcoStruxure Process Expert detects an incorrect configuration, it displays a notification, which allows you to modify the configuration.
5	Close the Control Participant window.

Changing the PRM Configuration

To change the configuration of a PRM, follow the procedure to configure a PRM and make changes in the Control Participant.

Saving changes to communication functions in the Control Participant deletes existing communication mappings, page 725 of the PRM.

If this is the case, select one of the following in the confirmation dialog box:

- **OK:** To save the change to the configuration file of the PRM and delete the associated communication mapping.
- **Cancel:** To keep the Control Participant window open without discarding changes.

PRM Representation

After you configure the PRM, save changes, and close the Control Participant, the representation of the entity in the **Topology Explorer** remains unchanged.

Refer to the topics describing the representation of entities in physical views, page 536 and the **SYSTEM PROJECT** pane, page 527 for more information.

Deleting a PRM Entity

After you confirm the delete command, the following are deleted:

- Physical links.
- The PRM.
- The corresponding Control Participant configuration project.

To delete the PRM, proceed as follows.

Step	Action
1	Right-click the PRM in the Topology Explorer and select Delete . Result: A dialog box opens.
2	Verify the information that is displayed and click OK to proceed. Result: The PRM is deleted and a message is displayed in the notification panel.

Configuring Modbus TCP and EtherNet/IP Devices

Overview

The **Modbus TCP Device** and **EtherNet/IP TCP Device** devices of the **EPE Managed Devices** product range model I/O devices that communicate with the controller by using the following communication protocols:

- Modbus TCP implicit (I/O scanning) or explicit messaging.
- EtherNet/IP explicit messaging.

Each topological entity requires selecting a topological (hardware) template to model either of the following:

- A specific Schneider Electric device (see *EcoStruxure Process Expert, General Purpose Library, Device Templates, Reference Manual*). For example, the *\$EATV6xxHW* template that models an Altivar Process ATV630 speed drive).
- A generic device for which no specific template is available. For example, the *\$EIPGenericDeviceHW* template. Several generic Modbus TCP templates exist to create the required number of I/O scanner lines.

NOTE: Devices created with the *\$EGenericDeviceHW* template can function as gateways if you implement monitoring of Modbus serial device data directly from the Supervision infrastructure, page 984.

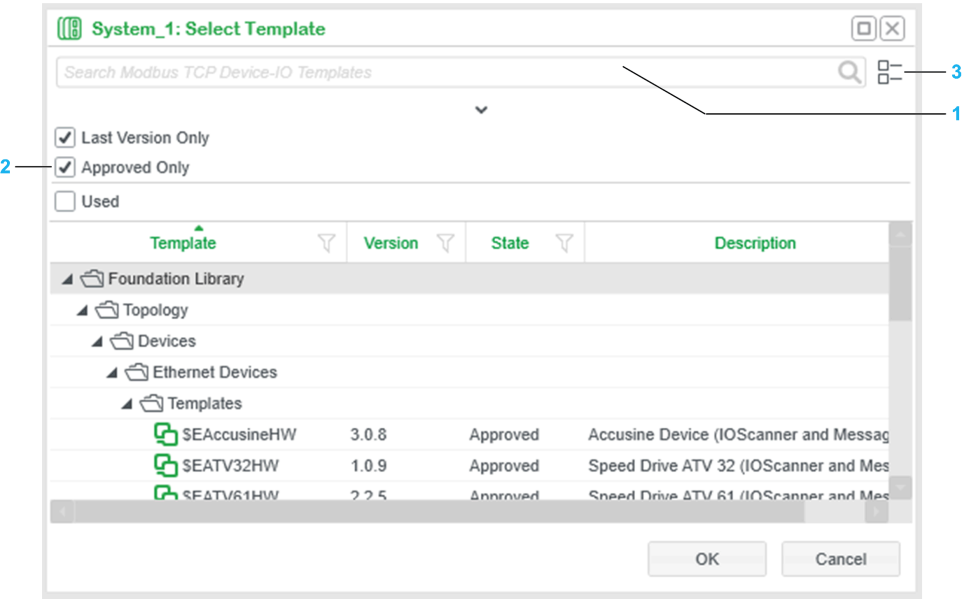
NOTE: To configure the devices of the other product ranges of the **Automation Devices** product type, configure their properties in the **PROPERTIES** pane and use the commands of their context menu.

Selecting Topological Device Templates

The **Select Template** dialog box displays templates of the Global Templates Library that fulfill the following criteria:

- **Type:** Topological
- **SubType:** Device-I/O
- **Usability State** (see *EcoStruxure Process Expert, Global Templates, Reference Manual*): **Approved** or **Deprecated**

The following figure shows an example of the **Select Template** dialog box that opens when you create a device with **Modbus TCP Device**.



Item	Description
1	Search field, page 164 to search for topological device templates.
2	Filters, page 165. The check mark indicates that the filter is applied. The Used filter applies to templates of topological entities.
3	Button to toggle between grid view and tree view.

Column header	Description
Template	Identifier of the template. In tree view mode, the path to the template in the Global Templates Library is shown.
Version	Version of the template.
State	Usability state (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) of the template.
Description	Description of the template.

Creating a Device Entity

To create an I/O device entity, refer to the topic describing how to add topological entities to a system, page 565.

Modbus TCP and EtherNet/IP Device Actions

For a description of actions that you can perform on Modbus TCP and EtherNet/IP I/O devices, refer to the topic describing entity context menu commands, page 570.

I/O Device Properties

To configure the properties of a device, select it and use the **PROPERTIES** pane, page 673. Other properties can be configured by using the Control Participant.

The default device name is *xHW_n* where *x* is the identifier of the corresponding application template of the device and *n* an incremental number starting at 1.

The full or partial device name is used to create the name of the DTM if it is added during build, page 776.

Changing I/O Device Properties

Modifying the IP address or subnet mask of an I/O device entity may delete the communication mapping, page 725 that exists with this device.

In this case, select one of the following in the confirmation dialog box:

- **Yes:** To save your changes to the properties of the I/O device and delete the associated communication mapping.
- **No:** To discard your changes.

Implementing Communication Between Controllers and I/O Devices

For information on how to implement communication between controllers and I/O devices by using Modbus TCP or EtherNet/IP protocols, refer to the topic describing how to manage communication, page 935.

Configuring HART Modules

Overview

The HART communication protocol can be addressed by using either of these Modicon eX80 modules:

- BMEAHI0812 HART analog input module.
- BMEAHO0412 HART analog output module.

You can add these HART analog I/O modules to local racks and/or remote drops of controllers of the Quantum and M580 platform. With the Quantum platform, a 140NOC.... communication module is required.

You configure HART modules in the Control Participant. Closing the Participant transfers the configuration changes to the **Topology Explorer** where the controller representation is updated.

NOTE: You can address the HART protocol either:

- With a Modicon STB island by using the STBAHI8321 HART interface module, page 599.
- With a gateway of the **Network Fieldbus Devices** product type by using the **Fieldbus Editor**, page 553.

Managed Data

The **Primary Variable** (PV) of HART analog I/O modules is the only variable that is available in the **Hardware Mapping Editor** during the mapping stage. There, you can map facets of analog I/O templates (for example, *\$AnalogInput1*) instantiated in the application, which represent PV, to the inputs and outputs of the HART modules.

The other variables and data of the HART modules that you configure are retained during build and can be deployed to a controller but are not usable at the system level.

NOTE: Use an Asset Management System and manually perform additional steps (see *EcoStruxure Process Expert*, *Foundation Application Templates*, *User Guide*) to take advantage of the advanced functionality of HART analog I/O modules.

HART Module Device DDT Variables

When you add a HART analog I/O module to a controller, the Participant creates a Device DDT (DDDT) variable associated to the I/O the module.

When you add the corresponding HART Device Type Manager (DTM) that is preinstalled, it creates a second DDDT variable associated to the DTM to manage the extended data.

Both variables exist only in the controller configuration that you can open from the **Topology Explorer** and are integrated to the Control Participant project when you build it, page 775. However, you can create the two DDDTs, page 455 to use them in a Control Participant project during the refinement stage already.

Configuring HART Modules

To configure HART analog I/O modules, proceed as follows.

Step	Action
1	Right-click the controller entity in the Topology Explorer and select Configure , page 591. Result: The Control Participant opens.
2	Configure the HART module by using the Control Participant. For more information about configuring HART analog I/O module and working inside the Control Participant, refer to the topic describing the configuration of the controller (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>).
3	Depending on the controller platform, save your changes as follows: <ul style="list-style-type: none"> • M580: Click Synchronize Changes, page 590 in the controller context menu or close the Control Participant window and save changes. • Quantum: Click the Save button in the title bar of the Control Participant window. Result: The representation, page 578 of the controller and, if applicable, its properties are updated in the Topology Explorer . NOTE: If EcoStruxure Process Expert detects an incorrect configuration, it displays a notification, which allows you to modify the configuration.

Changing the HART Module Configuration

To change the configuration of a HART analog I/O module, follow the procedure to configure a HART module and make changes in the Control Participant.

HART Module Representation

After you add HART modules to a controller, save changes, and close the Control Participant, HART modules do not appear in the **SYSTEM PROJECT** pane or physical views.

Refer to the topics describing the representation of controllers in physical views, page 536 and the **SYSTEM PROJECT** pane, page 527 for more information.

Connecting Ports of Entities by Physical Links

Overview

After you added at least two entities to a physical view of your system topology, you can connect their communication ports with a physical link. This link models a cable connection. Color coding can be used to indicate the type of network, page 625.

You can connect ports of entities that appear in the same or in different physical views.

Once you have created the connection, the path of the physical link is automatically drawn. You can change the routing of the link manually.

If you reposition an entity, you may need to move links that you routed manually.

The high-speed cable link that connects the local racks of a redundant controller is considered a physical link but some of the functionality described in this topic may not apply to it.

NOTE: A physical link connection does not connect an entity to a logical network, page 576.

NOTE: To connect Quantum and M340 controllers to a logical network, refer to the topic describing how to connect to an Ethernet network, page 615.

NOTE: You cannot connect ports of PROFIBUS Remote Masters of the **EPE Managed Devices** product range.

Connection Rules

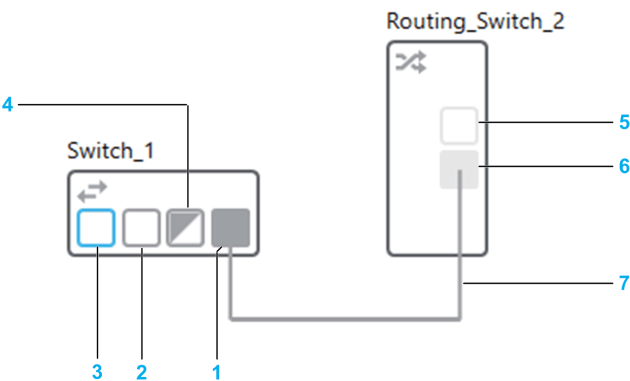
The following rules apply when you connect entities by a physical link:

- A link can create only a one-to-one connection between two ports of the same or distinct entities.
- Each port can be connected only once.
- To connect entities that appear in different physical views:
 - One of the ports to be connected by a physical link must be tagged for connection, page 611.
 - The two physical views must be open.

NOTE: You can connect a port that is disabled but it is reported when you analyze, page 1090 the topology.

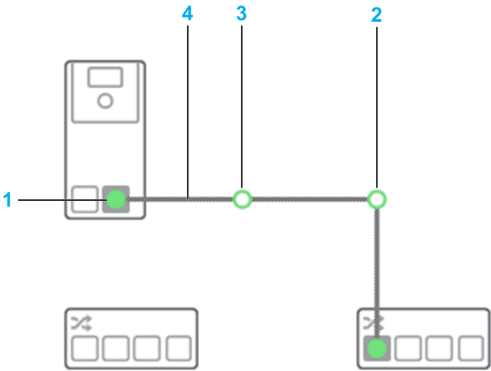
Port Status and Physical Link Representation

The following figure shows an example of the representation of two devices, page 578 with various port configurations in a physical view. The devices are connected by a physical link.



Item	Description
1	Linked port
2	Port available for connection (port enabled)
3	Port that is tagged for connection, page 611
4	Port that is connected to an entity that was removed from the physical view. The port is not available for connection anymore.
5	Port available for connection (port disabled)
6	Disabled and linked port
7	Physical link

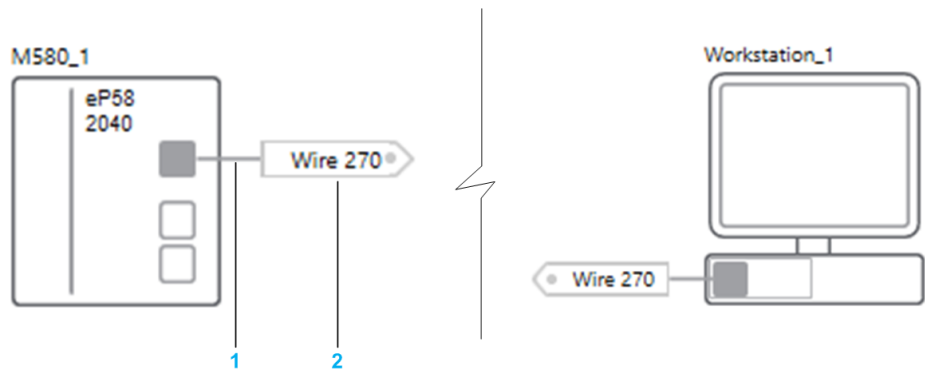
The following figure shows an example of a physical link that is selected.



Item	Description
1	Link endpoint
2	Joint created automatically
3	User-created joint
4	Segment of the link (portion of the link that is between two joints or a joint and an endpoint)

Remote Link Representation

The following figure shows an example of a remote link between a controller and a workstation that are located in the same or different physical views. If entities are located in the same physical view, showing the link as a remote link, page 613 is optional.



Item	Description
1	Physical link stub
2	Connector. The same connector is shown for each of the two connected ports. To rename it, edit its Label attribute in the PROPERTIES pane.

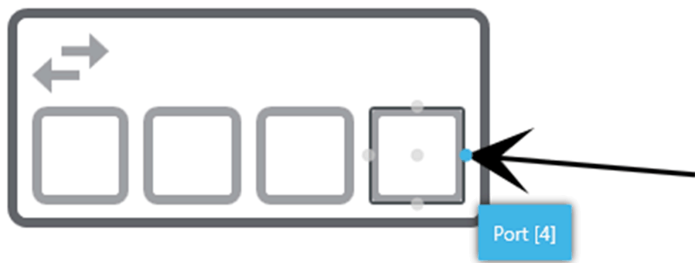
A tooltip indicates:

- The value of the **Label** attribute. If it is blank, a numerical ID is shown.
- The destination of the link in the format *physical view name\entity name\port name* where:
 - *physical view name* is shown only if the entity is located in a different physical view.
 - *entity name* can be followed by the name of one or more child objects in case of racks. For example, *Standalone_Controller_2\M580_1\BMEXBP0800 #0\BMEP582040 #0* where *Standalone_Controller_2* is the name of the controller and *M580_1* the local rack.

Port Name, Anchors, and Link Direction

The name of a port is indicated by a tooltip when you hover with the pointer over the port.

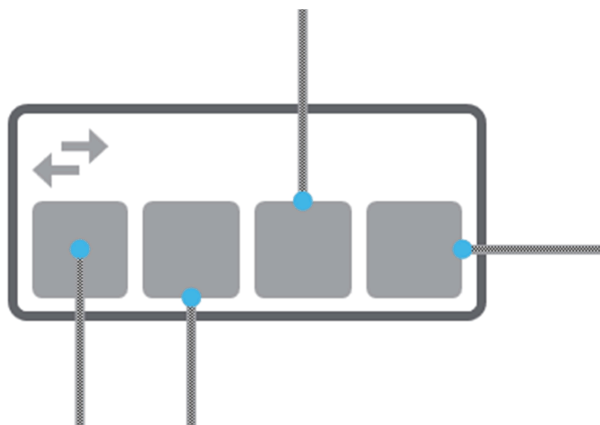
Each port has five anchors from which you can start a link or to which you can connect a link.



The anchors appear on the target port only when you position the pointer over it.

The anchor that you select to start or connect the link dictates the initial direction of the link. If you select the center anchor, the links are automatically routed.

The following figure shows four links connected to different anchors of ports and the resulting link direction as an example.



NOTE: You can move an existing physical link to a different anchor of the same port to modify the link direction.

Port Actions

Right-click the available port of an entity to open a context menu containing the following commands (the port can be in disabled state).

Command	Description
Tag for Connection	<p>When selected, lets you connect the port, page 612 from an entity located in another open physical view. The name of the port appears in a submenu of the Connect to command when you select the command on the other entity.</p> <p>NOTE: Closing a physical view clears the tags for connection for the ports of entities appearing in this physical view.</p> <p>NOTE: Deleting or removing a remote link removes tag for connection if it was set for the port.</p>
Connect to	<p>Lists the ports of entities that appear on the physical view that is open and that are tagged for connection.</p> <p>If Tag for Connection was selected for a port, the command is unavailable.</p> <p>Ports are shown by using the format <i>physical view name\entity name\port name</i> where:</p> <ul style="list-style-type: none"> <i>physical view name</i> is shown only if the entity is located in a different physical view. <i>entity name</i> can be followed by the name of one or more child objects in case of racks. For example, <i>Standalone_Controller_2\M580_1\BMEXBP0800 #0\BMEP582040 #0</i> where <i>Standalone_Controller_2</i> is the name of the controller and <i>M580_1</i> the local rack. <p>NOTE: Once you have connected one of the listed ports, it is not tagged for connection anymore.</p>

Connecting Ports of Entities in the Same Physical View

To connect two ports by a physical link, proceed as follows.

Step	Action
1	In a physical view, position the pointer over the center or one of the edges of a port that is available for connection. Result: A link symbol appears next to the pointer.
2	Press and hold the left mouse button and drag the pointer away from the port. Result: A line with an arrow end is drawn.
3	You can release the mouse button and move the arrow over the port that is the target for the connection. Result: The anchors, page 610 of the port are shown. NOTE: To cancel the link drawing, press Esc .
4	Move the arrow over one of the anchors, page 610 of the port so that it is highlighted and click it. Result: The port is connected and the link between the entities is drawn. The link is shown selected, which lets you view endpoints and joints.

Connecting Ports of Entities Appearing in Different Physical Views

You can also use the following procedure to connect entities within the same physical view. In this case, a physical link is drawn between the ports.

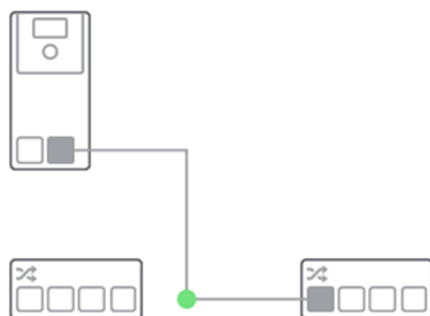
To connect two ports by a physical link when entities are located in different physical views, proceed as follows.

Step	Action
1	In one of the physical views, right-click the port of an entity and select Tag for Connection . Result: The port is outlined.
2	In the other physical view, right-click the port to be connected and select Connect to . Result: The name of the port that is tagged for connection is shown. (Any other port that is tagged for connection appears as well.)
3	Select the port. Result: A remote link is created between both ports in each physical view. The anchors, page 610 are automatically selected.

Changing the Connected Port or Routing of an Existing Physical Link

To connect either endpoint of an existing physical link to another available port, select the endpoint and drag it to the other port. The physical link is rerouted automatically. Manual changes to the route and joints that you added are discarded.

You can also modify the route of a link. The following figure shows the automatically routed link from the previous example that was rerouted manually by dragging the left-hand joint to a new position. The joint is shown selected. This action disables the automatic routing, page 613 feature of the link.



For remote links, you can drag the connector, page 610 to a new position.

NOTE: If you route links manually, a soft snapping is applied and guidelines appear to help you align with graphical elements and maintain distances.

NOTE: You can delete both automatically and user-created joints by right-clicking them and selecting **Delete**.

Physical Link Actions

Right-click a physical link or the stub of a remote link, page 610 to open a context menu containing the following commands.

Command	Description
Add Point	<p>Inserts a joint where you right-clicked the link.</p> <p>It creates two segments and lets you:</p> <ul style="list-style-type: none"> • Move either segment by dragging it. • Modify the direction of either segment to change the routing by dragging the joint. <p>NOTE: Deleting a joint realigns the two segments.</p>
Source	<p>Opens a submenu that lets you configure in which direction the link exits the port from which it originates. You can use it, for example, to change the default routing of links.</p> <p>For a description of the submenu entries, refer to the topic describing physical link properties.</p> <p>NOTE: You can also use the port anchors, page 610.</p>
Destination	<p>Opens a submenu that lets you configure in which direction the link enters the port to which it is connected. You can use it, for example, to change the default routing of links.</p> <p>For a description of the submenu entries, refer to the topic describing physical link properties.</p> <p>NOTE: You can also use the port anchors, page 610.</p>
Auto Routing	<p>Selected by default after you create the link.</p> <p>It is cleared if you add a joint or change the routing of the link.</p> <p>Selecting it again discards changes and routes the link automatically.</p>
Show as remote link	<p>When selected, shows only both ends of the physical link. Each stub ends with a connector that displays the five first characters of the label.</p> <p>If you clear the check box, a solid link is redrawn and the tag removed. Other existing links may be repositioned as needed.</p> <p>NOTE: The command is selected and disabled for remote links connecting entities that are located in different physical views.</p>
Delete	<p>Deletes the link and makes the ports that were connected available again.</p> <p>You must confirm the command.</p> <p>For remote links, it also deletes the counterpart link in the other physical view.</p>

Remote Link Connector Actions

Right-click the [connector](#), [page 610](#) of a remote link to open a context menu containing the following commands.

Command	Description
Navigate	Shows the entity that is connected at the other end and highlights the counterpart connector in the physical view. If it is located in another physical view, it is opened and/or made active. You can also double-click the connector.
Horizontal/Vertical	Defines the orientation of the connector. NOTE: Corresponds to the Orientation attribute of the connector.
Delete	Deletes the connector, the link, and their counterparts in the other physical view after you confirm the command.

Physical Link Properties

Click a physical link or the [stub of a remote link](#), [page 610](#) to display its properties in the **CONFIGURATION** tab of the **PROPERTIES** pane.

You can view and/or edit the following properties of the **General** category.

Attribute	Description
Label	Editable name. The name does not need to be unique in the topology. NOTE: The name is used as label for the link in the physical view. You can show or hide the label by using the command in the toolbar of the physical view.
Source	The following settings let you configure how the link exits the port from which it originates. The link is connected to the corresponding anchor , page 610 . <ul style="list-style-type: none"> • Default: The direction that is defined by default for this type of entity is used. • Top: The link exits the port from the top. • Bottom: The link exits the port from the bottom. • Right: The link exits the port from the right. • Left: The link exits the port from the left.
Destination	The following settings let you configure how the link enters the port to which it is connected. The link is connected to the corresponding anchor , page 610 . <ul style="list-style-type: none"> • Default: The direction that is defined by default for this type of entity is used. • Top: The link enters the port from the top. • Bottom: The link enters the port from the bottom. • Right: The link enters the port from the right. • Left: The link enters the port from the left.

Connecting Quantum and M340 Controllers and PRMs to Logical Networks

Overview

The **Physical Connections** command allows you to connect the following topological entities that support Modbus TCP or EtherNet/IP connectivity to logical networks, page 576 of the topology:

- Controllers of the M340 and Quantum platforms.
- PROFIBUS Remote Masters (PRMs) of the **EPE Managed Devices** product range.

The connection models the physical connection of their communication modules to a network.

You must perform the communication mapping, page 719 in Control and Supervision executables. A connection is not required to perform the service mapping or deployment and execution tasks.

You can also use the command to view existing connections of communication modules and disconnect them from these networks.

IP Address Uniqueness Requirement

The IP address of a topological entity on a logical network must be unique to connect the entity to the network by using the **Physical Connections** command. This rule applies to PRMs and to the following modules of M340 and Quantum controllers:

- Controller modules (including the main IP address and IP address A depending on the controller platform)
- Communication modules, such as:
 - BMXNOE••••
 - BMXNOC•••• and 140NOC•••• (for 140NOC78•00 modules, main IP address only, not IP address A)

The IP addresses configured for the above entities and modules are verified against the IP addresses of the other entities of the topology. For M580 controllers, you must have configured the controller or synchronized changes after setting its IP address so that it is taken into account.

NOTE: Also refer to rules for redundant controllers, page 616.

Availability of the Physical Connections Dialog Box

If the Control Participant, the **Communication Mapping Editor**, or the **Hardware Mapping Editor** is open for the topological entity, the **Physical Connections** dialog box does not open.

Managing Conflicting Connections to Logical Networks

If the topology contains at least two topological entities that are either an M340 or a Quantum Controller, or a PRM, each having a communication module connected to the same logical network (**Physical Connections**), if you configure a duplicate IP address for a module of one of these entities and save the configuration, a dialog box opens. Click either of the following:

- **OK:** Disconnects the module from the logical network. This deletes all or part of the existing communication and hardware mapping of the associated Control executable.
- **Cancel:** Keeps the Control Participant open without saving and allows you to change the IP address to retain the connection to the logical network.

For more information, refer to Considerations When Updating an Existing Configuration (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

Connections With Redundant Controllers

For redundant controllers, the rule that requires an IP address to be unique, page 615 on a logical network to allow connecting the controller to this network by using the **Physical Connections** command applies to the following:

- For controller modules: (Main) *IP address +1* (where *IP address* is the address of the controller module of the primary controller).
- For BMXNOE**** and 140NOC78*00 communication modules: *IP address +1* (where *IP address* is the address of the communication module of the primary controller).

If a standalone controller is connected to a logical network and you convert it into a redundant controller, connected modules of the controller that becomes the primary controller may be disconnected. This affects existing mappings, page 593.

The following table describes the various scenarios if you convert a standalone controller to a redundant controller depending on the status of the IP address of counterpart modules in the standby controller. The standalone controller becomes the primary controller of the redundant configuration.

Module connection status in the standalone configuration	IP address ⁽¹⁾ of counterpart module in the standby controller after changing to a redundant configuration	Impact on the connection of both modules to the logical network
Connected	Unique	Both modules are connected.
Connected	Already exists	The module of the primary controller is disconnected. The module of the standby controller is not connected.
Not connected	irrelevant	Both modules are not connected.
(1) If the module has two IP addresses both are considered unless otherwise mentioned.		

Connecting an M340 or Quantum Controller to a Logical Network

To connect an M340 or Quantum controller to a logical network, proceed as follows.

Step	Action
1	<p>Right-click the controller in the Topology Explorer and select Physical Connections.</p> <p>Result: The Physical Connections dialog box opens and displays the identifier of the communication modules of the controller in the Communication Modules column.</p> <p>NOTE: The identifier only appears if you assigned an IP address to the communication module in the configuration.</p> <p>NOTE: When connecting a redundant controller, only communication modules of the primary controller are displayed.</p> <p>NOTE: When connecting a controller that has a 140CRP..... module, the EthRIO bus device that is displayed in the Physical Connections dialog box represents the connection with the remote I/O network and the corresponding communication modules.</p>
2	<p>Click the Allowed Network(s) list next to the communication module that you want to connect.</p> <p>Result: The logical networks to which a connection is possible are displayed.</p>
3	Select a network.
4	<p>Click OK to save changes and close the Physical Connections dialog box.</p> <p>Result: The communication module is connected to the selected logical network.</p> <p>NOTE: For a redundant controller, EcoStruxure Process Expert automatically connects the counterpart module of the standby controller to the same logical network.</p>

Disconnecting an M340 or Quantum Controller From a Logical Network

To disconnect an M340 or Quantum Controller from a logical network, proceed as follows.

Step	Action
1	<p>Right-click the controller in the Topology Explorer and select Physical Connections.</p> <p>Result: The Physical Connections dialog box opens and displays the name of:</p> <ul style="list-style-type: none"> The communication modules of the controller in the Communication Modules column. The logical network to which a module is connected in the Allowed Network(s) column. <p>NOTE: For a redundant controller, only the communication modules of the primary controller are displayed.</p> <p>NOTE: When disconnecting a controller that has 140CRP..... module, the EthRIO bus device that is displayed in the Physical Connections dialog box represents the connection with the remote I/O network and disconnects the corresponding communication modules.</p>
2	Click the Allowed Network(s) list next to the communication module that you want to disconnect.
3	<p>Select Not Assigned.</p> <p>NOTE: If you select another logical network from the list instead (if available) the network connection of the module is changed.</p>
4	<p>Click OK.</p> <p>Result: If a communication mapping exists for the executable to which the controller is mapped, a dialog box opens prompting you to confirm the disconnection.</p> <p>Otherwise, the communication module is disconnected from the logical network and the Physical Connections dialog box closes.</p> <p>NOTE: For a redundant controller, EcoStruxure Process Expert automatically disconnects the counterpart module of the standby controller from this logical network.</p>
5	<p>In the dialog box, click Yes to confirm the disconnection.</p> <p>Result: The communication module is disconnected from the logical network, existing mappings, page 593 of the executable to which the controller is mapped are deleted, and the status of the executable changes to Out Of Date if it is built.</p> <p>NOTE: Click No to close the Physical Connections dialog box without disconnecting the communication module from the logical network.</p> <p>NOTE: For a redundant controller, EcoStruxure Process Expert automatically disconnects the counterpart module of the standby controller from the same logical network.</p>

Connecting a PRM to a Logical Network

To connect a PRM of the **EPE Managed Devices** product range to a logical network, proceed as follows.

Step	Action
1	<p>Right-click the PRM in the Topology Explorer and select Physical Connections.</p> <p>Result: The Physical Connections dialog box opens and displays the PRM in the Communication Modules column.</p> <p>NOTE: The PRM is displayed only if you assigned an IP address, page 601 to it.</p>
2	<p>Click the Allowed Network(s) list next to the PRM.</p> <p>Result: The logical networks to which a connection is possible are displayed.</p>
3	Select a network.
4	<p>Click OK to save changes and close the Physical Connections window.</p> <p>Result: The PRM is connected to the selected logical network.</p>

Disconnecting a PRM from a Logical Network

To disconnect a PRM of the **EPE Managed Devices** product range from a logical network, proceed as follows.

Step	Action
1	<p>Right-click the PRM in the Topology Explorer and select Physical Connections.</p> <p>Result: The Physical Connections dialog box opens and displays the name of:</p> <ul style="list-style-type: none"> The PRM in the Communication Modules column. The logical network to which the PRM is connected in the Allowed Network(s) column.
2	Click the the Allowed Network(s) list next to the PRM.
3	<p>Select Not Assigned in the Allowed Network(s) list.</p> <p>NOTE: If you select another logical network from the list instead (if available) the network connection of the PRM is changed.</p>
4	<p>Click OK.</p> <p>Result: If a hardware or communication mapping exists for the executable of the client controller communicating with the PRM, a dialog box opens prompting you to confirm the disconnection.</p> <p>Otherwise, the PRM is disconnected from the selected logical network and the Physical Connections dialog box closes.</p>
5	<p>In the dialog box, click Yes to confirm the disconnection.</p> <p>Result: The PRM is disconnected from the selected logical network, existing hardware mapping, page 747 and communication mapping, page 725 of the executable of the client controller communicating with the PRM are deleted, and the status of the executable changes to Out Of Date if it is built.</p> <p>NOTE: Click No to close the Physical Connections dialog box without disconnecting the PRM from the logical network.</p>

Configuring Fieldbuses

Managing Field Devices on Fieldbuses

Overview

You can edit the fieldbus of the following types of gateways:

- PROFIBUS remote master (PRM) gateways of the **Network Fieldbus Devices** product type, **Gateway** product range. It lets you add child devices and INTERBUS network coupling devices.
- HART modules of a controller. Lets you add field devices to a channel.

After you add field devices from the **TOOLBOX**, you can configure their parameters, page 673 to your requirements.

You can perform these tasks by using the **Fieldbus Editor**.

While the **Fieldbus Editor** or the device type manager (DTM) of a device is open, you cannot edit the attributes of the device and its child devices by using the **PROPERTIES** pane.

The representation of the gateway in the **SYSTEM PROJECT** pane is updated after you close the **Fieldbus Editor** and save changes.

NOTE: In a physical view, field devices of gateways are not shown.

Adding and Configuring Field Devices

To add a field device to the fieldbus of an existing gateway device and configure it, proceed as follows.

Step	Action
1	Open the Fieldbus Editor of the gateway.
2	In the TOOLBOX pane, select a field device and drag it onto the gateway or along the fieldbus. Result: The Select DTM Reference dialog box opens.
3	Select a reference, page 629 from the list and click OK . You can sort and filter DTMs. Result: Independently of where you dragged the field device, a logical number, page 687 is automatically assigned to it. The newly added field device is shown selected. NOTE: If the field device is a network coupler, such as a PROFIBUS-to-INTERBUS gateway, you can add compatible child devices to it in the same way.
4	To modify the configuration of the field device, open its DTM by using the context menu of the field device.
5	To modify the configuration of the gateway, open its DTM by using the context menu of the gateway. NOTE: This lets you also change the station address of PROFIBUS child devices.
6	Close the Fieldbus Editor . Result: A confirmation dialog box opens.
7	Click Yes . Result: Your changes are saved and the Fieldbus Editor closes. This operation also saves other changes you made, such as editing values in the DTM of the gateway or field devices.

Changing the Channel Number of a HART Field Device

To assign a HART field device to a different channel of the gateway, proceed as follows.

Step	Action
1	Open the Fieldbus Editor of the gateway and select the HART field device.
2	<p>In the PROPERTIES pane, enter a valid channel number in Logical Number.</p> <p>Result: The value is applied to the DTM of the gateway and the field device is repositioned in the Fieldbus Editor.</p> <p>NOTE: If you entered a channel number that is already used, both field devices swap channels.</p>
3	Close the Fieldbus Editor and save changes.

Managing the Topology

Overview

This section describes how to manage the topology of the system.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Updating Templates of Topological Entities

Overview

By using the **Update Template** entity context menu command in the **Topology Explorer**, you can change the version of templates that are used by the following topological entities:

- Controllers.
- Devices of the **EPE Managed Devices** product range.
- Workstations, their Network Interface Cards (NICs) and services.

If the update of one template cannot be completed for an entity, the entire update process is rolled back.

NOTE: The update functionality impacts:

- Existing service, communication, and hardware mapping interface links.
- Deployment information related to the executable to which the entity is mapped.

NOTE: For Modbus TCP and EtherNet/IP devices of the **EPE Managed Devices** product range, the **Replace Template** command is available and allows you to select a different template to be used by the device.

Default Template Selection

For the update, by default, EcoStruxure Process Expert selects the topological template with the same identifier that has the highest version number and the **Usability State** of **Approved**.

Description of the Update Functionality

The following table describes the result on various item configurations when you update the present template that is used by a topological entity with a higher version.

Item	Identifier in higher version template	Result
Parameter values	Is the same as in the present template	Explicitly defined parameters of the present template keep their respective value in the higher version template.
	Is not in the present template	New parameters that are not part of the present template keep their default values.
Mappings, page 720	–	EcoStruxure Process Expert maintains the existing service, communication, and hardware mappings of the Control and Supervision executables to which the entity is mapped if the necessary mapping interfaces are available and satisfy applicable rules.
Deployment information	–	If you deployed a Control project to a controller or controller simulator (workstation), the corresponding deployment information is unchanged.

NOTE: If you update the template of a redundant controller, the update is performed for the primary and standby controllers.

Updating Templates

To update the templates that are used by a topological entity, proceed as follows.

Step	Action
1	<p>In the Topology Explorer, right-click the entity and select Update Template.</p> <p>Result: The Update Device Template dialog box opens. It shows for each device template that can be updated, the version that will be used.</p> <p>NOTE: A submenu may open, which allows you to update only the templates of subcomponents of an entity, such as NICs, or update the entity entirely.</p>
2	<p>Clear the check boxes of templates that you want to exclude from the update, if required, and click OK.</p> <p>Result: The selected templates are updated. Existing mappings of the corresponding executable of the associated Control and/or Supervision project may be deleted.</p>

NOTE: If you use a filter in the **Name** column of the **Update Device Template** dialog box, the higher versions of templates that are hidden but selected by default are also used for the update operation.

Assigning Devices and Modules to DHCP Servers

Overview

This topic describes how you can assign a device or module that features a DHCP client service to a DHCP server by using the **DHCP VIEW**, page 550. It also describes how to unsubscribe a device or module from the DHCP service.

For certain devices or modules, the DHCP client/server relation cannot be modified (for example, BMECRA•••• adapter modules).

You can perform an analysis, page 1091 of the topology to detect inconsistencies.

NOTE: Alternatively, you can add a device or module to a DHCP server or unsubscribe it by configuring its DHCP client attributes, page 694 in the **Services** tab of the **PROPERTIES** pane.

Adding Devices and Modules to a DHCP Server

To add devices or modules to a DHCP server, proceed as follows.

Step	Action
1	Select one or more devices and modules in the Not Assigned node. NOTE: Your selection can also include some that are already assigned to a DHCP server if you want to reassign them to another server.
2	Drag your selection to a DHCP server node. Result: <ul style="list-style-type: none">• The selected devices and modules are added to the list of clients of the DHCP server.• The DHCP Client setting, page 694 of the devices and modules is updated. NOTE: Adding a device or module to a DHCP server does not imply that the DHCP service is functional. Services may be disabled and/or protocols incompatible.

Unsubscribing Devices and Modules From the DHCP Service

To unsubscribe devices or modules from the DHCP service, proceed as follows.

Step	Action
1	Select one or more devices and modules.
2	Drag your selection to the Not Assigned node. Result: <ul style="list-style-type: none">• The selected devices and modules do not subscribed to a DHCP service anymore.• The DHCP Client setting, page 694 of the devices and modules is updated.

Managing Logical Networks and IP Addresses

Highlighting Logical Networks in Physical Views

Overview

In a physical view, by using the **Select Logical Networks** list, you can highlight the ports whose interface belongs to a given logical network as well as the physical links connecting them. You can highlight more than one network at a time.

The functionality lets you view, in a graphical way, which entities a logical network encompasses and how it is laid out.

Working Principle

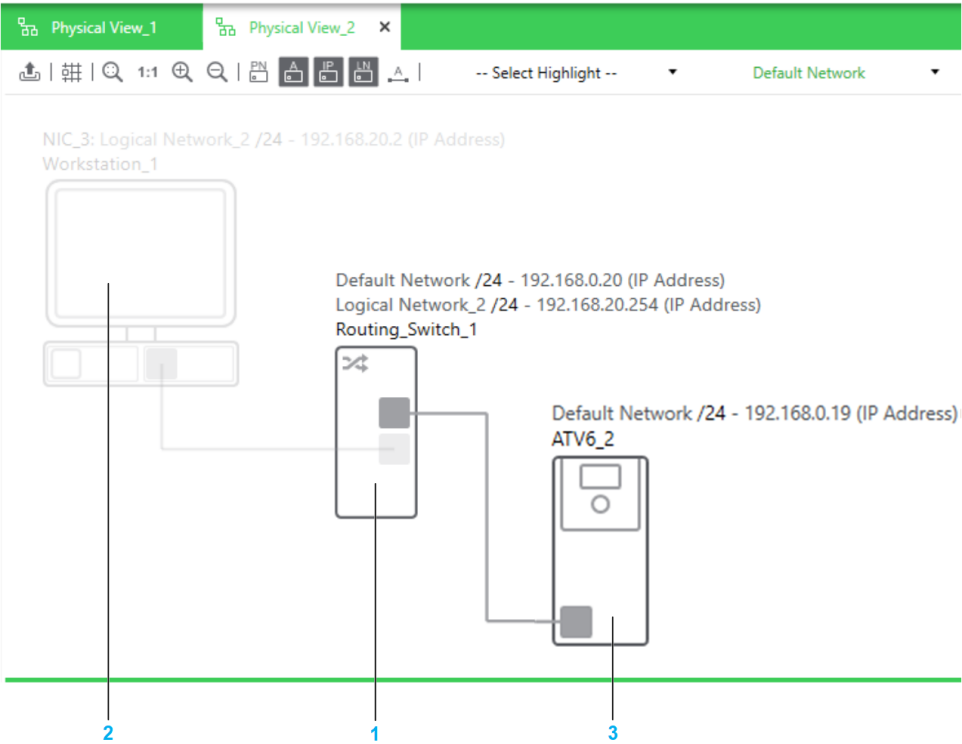
The following table describes the working principles of the logical network highlighting functionality.

Scope	The functionality highlights objects that appear in the physical view where you make the logical network selection. NOTE: In this context, highlight means the emphasis of the object, while non-highlighted means the object is grayed-out.
Representation	Highlighted objects keep their original appearance. Non-highlighted objects appear in a lighter shade. As soon as one port of an entity is highlighted, the outline of the entire entity is highlighted. The physical link between two highlighted ports is also highlighted. NOTE: The enabled ports of unmanaged switches (Network Infrastructure Devices product type) are highlighted as soon as one of them is connected to a port that is highlighted.
Refresh	The highlighting is automatically updated to reflect changes that you make to the configuration of ports (for example, changing the Logical Network setting of a port from <i>NOCONF</i> to a logical network that is already highlighted).
Persistence	The network selection is cleared when you close the physical view.

NOTE: If you highlight several logical networks, the entities, their ports, and physical links that belong to them are highlighted without distinction of logical network.

Example

The following example shows several entities in a physical view that have their interfaces configured with different logical networks while *Default Network* is selected to be highlighted.



Item	Object	Description
1	Routing_Switch_1	<p>Port2 (bottom) belongs to <i>Logical Network_2</i> and is not highlighted.</p> <p>Port1 (top) belongs to <i>Default Network</i> and is highlighted. It also highlights the device outline.</p>
2	Workstation_1	Its first port (not connected) is not configured, the second belongs to <i>Logical Network_2</i> and none of them are highlighted; nor is the physical link.
3	ATV6_2	<p>Its port belongs to <i>Default Network</i> and is highlighted together with the entity outline.</p> <p>The physical link is also highlighted because it connects two ports that belong to <i>Default Network</i>.</p>

Changing IP Address Assignments to Logical Networks at Topology Level

Overview

This topic describes how you can manage the assignment of IPv4 addresses to logical networks, page 541 at the topology level by using the **LOGICAL NETWORK VIEW**.

Modifying IP Addresses

To modify an IP address, double-click it and enter the new value.

You cannot modify an IP address in the **LOGICAL NETWORK VIEW** if the corresponding attribute is read-only when accessed through the **PROPERTIES** pane.

NOTE: The syntax of the IP address and its validity in relation to the logical network are verified. If the status is invalid, you can automatically assign a new valid IP address to the interface by using the **Reassign IP Address** command. The uniqueness of the IP address is verified when you analyze, page 1091 the topology.

Rules for the Assignment of IP Addresses to Another Logical Network

The following table describes the rules that apply when you assign an interface to a new logical network in the **LOGICAL NETWORK VIEW**.

Scope	Description
Selection of entities	You can select one or more interfaces belonging to the same or different logical networks.
	When you select an interface that is associated to another one and assign the interface to a new logical network, the associated interface is automatically assigned to the same network. (For example, when you reassign only the MainIP address of a controller module, IPA is reassigned too. For a redundant controller, MainIP+1 and IPB are reassigned as well.)
Target logical network	You can drag your selection to the logical network group or onto the list of interfaces inside the group. The resulting IP address assignment is the same.
IP address assignment	<p>The interface that you assign to a new logical network is automatically assigned the next available IP address on this network. The values of the corresponding attributes of the interface are updated.</p> <p>If there are not enough IP addresses available in the target logical network, the action is canceled and a notification displayed.</p>

Modifying Logical Networks

When you modify the subnet address or subnet mask of a logical network to which interfaces are assigned, EcoStruxure Process Expert verifies the validity of each IP address in relation to the new logical network data and updates its status in the corresponding column of the **LOGICAL NETWORK VIEW**. If the status is invalid, you can automatically assign a new valid IP address to interfaces by using the **Reassign IP Address** command.

Assigning IP Addresses by Logical Network

To assign one or more IP addresses to a logical network, proceed as follows.

Step	Action
1	<p>In the LOGICAL NETWORK VIEW, select one or more interfaces in the NOCONF node.</p> <p>Result: Your selection is highlighted.</p> <p>NOTE: Your selection can also include one or more interfaces that are already assigned to a logical network if you want to reassign them to another logical network.</p>
2	<p>Drag your selection to a logical network</p> <p>Result: The interfaces belong to the logical network and each IP address is replaced by the next available address, page 627 on this network.</p>

Clearing Assignments to Logical Networks

To clear the logical network assignment for one or more interfaces, select them and drag your selection to the **NOCONF** node.

The logical network and IP address information is deleted.

Using Device Type Managers

Using DTMs

Overview

Devices other than those belonging to the **EPE Managed Devices** product range, can be configured further by using a device type manager (DTM).

These DTMs are managed in the **Topology Explorer** where they can be opened and configured.

You can add DTMs to the local DTM catalog.

You may be able to deploy and load the DTM configuration to and from a device.

To see which DTM version is in use and whether it can be updated, refer to the topics describing the analysis of DTMs, page 633.

NOTE: Certain concurrent actions or actions performed by other users may temporarily lock a DTM and you cannot save changes if you close it. A dialog box informs you of the lock when you open the DTM. In this case, close the DTM and reopen it once the lock has been released.

Prerequisites

To be used, the DTM of a device must be created on the local computer and the local DTM **Hardware Catalog** updated, page 513.

NOTE: Ensure that the DTM **Hardware Catalog** is identical on the computers connecting to a same Control Expert server.

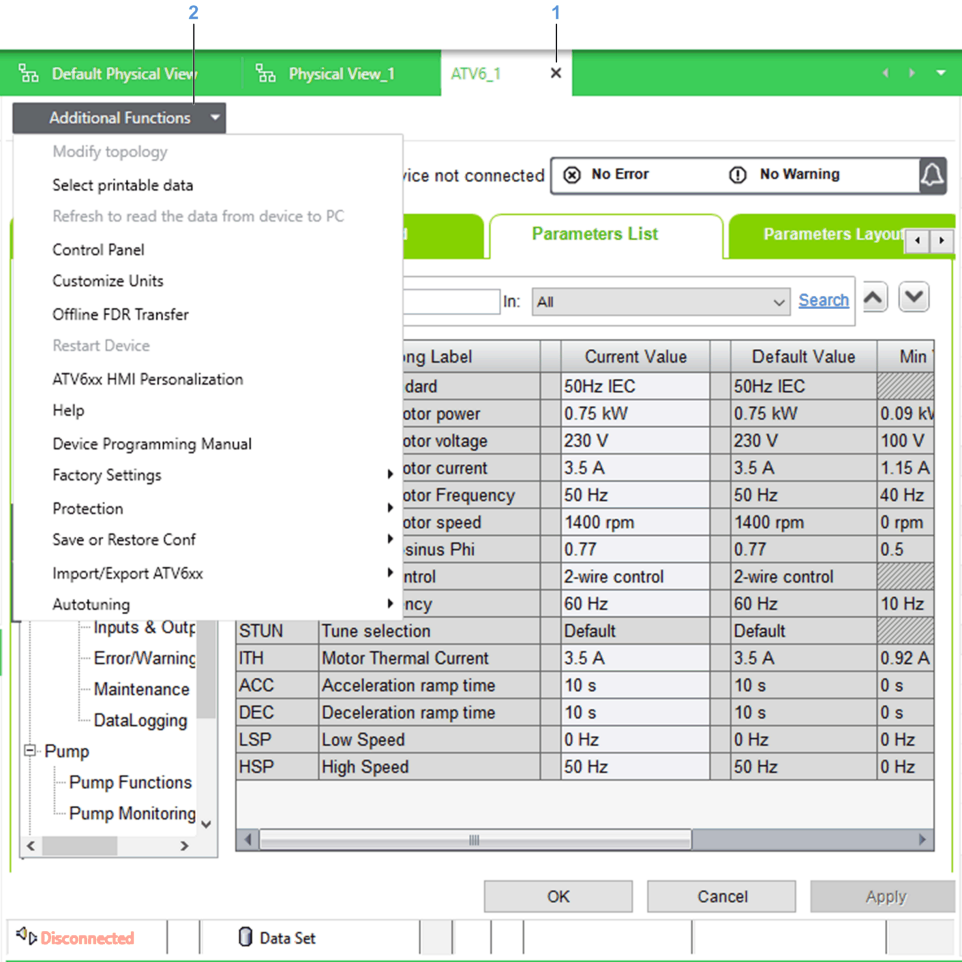
Automatic Update of DTMs

If you open the DTM of a device and a later version of this DTM is available in the local DTM **Hardware Catalog**, the device DTM is automatically updated.

NOTE: You can also manually update, at once, the DTM that is used by several devices by using the **MANAGE DTMs** command, page 513.

DTM Window

The following figure shows an example of DTM window that opens when you use the **Open** command of the device context menu. In the example, the **Additional Functions** menu is expanded to show the additional DTM commands that are available while the device is disconnected.



Item	Description
1	Button to close the DTM.
2	<p>Menu that lets you access various additional functions that are featured by the DTM. The content of the menu is DTM-specific.</p> <p>The availability of commands depends on the connection state of the device. If the device is disconnected, commands that require the device to be connected are shown disabled, and vice versa.</p>

File Types Used to Create DTMs

DTMs can be created by using the following types of files.

File type	Description
Executable	The installation method of the DTM is defined by the device manufacturer.
EDS	This file type is used for EtherNet/IP devices and requires using an installation tool.
GSD	This file type is used for PROFIBUS devices and requires using an installation tool.

FdtDTM Settings

The **FdtDtm Settings** section of the **User Preferences** window, page 524 lets you configure the following:

- Which network adapter of the computer is used to connect to a device.
- The path where the installation tool for EDS and GSD files is located.

Managing EDS Files

The following table describes the commands of the **Tools > Manage EDS Library** submenu in the toolbar of the **Topology Explorer**.

Command	Description
Add Device From EDS	Opens a wizard (see <i>EcoStruxure™ Control Expert, Operating Modes</i>) that lets you add a DTM to the Hardware Catalog by using an EDS file.
Remove Device From EDS	Opens a wizard that lets you remove a DTM from the Hardware Catalog by removing the corresponding EDS file.
Export EDS Library	Lets you create an export file (see <i>EcoStruxure™ Control Expert, Operating Modes</i>), which contains the EDS files of the local EDS Library.
Import EDS Library	Lets you import an EDS Library to the local computer.

Managing GSD Files

The following table describes the commands of the **Tools > Manage GSD Library** submenu in the toolbar of the **Topology Explorer**.

Command	Description
Add Device From GSD	Opens a wizard that lets you add a DTM to the Hardware Catalog by using a GSD file.
Remove Device From GSD	Opens a wizard that lets you remove a DTM from the Hardware Catalog by removing the corresponding GSD file.
Export GSD Library	Opens a wizard that lets you export to a Library file (.glb) the GSD files that are installed on the local computer at a path that you select. You can enter the name and location of the GSD Library export file.
Import GSD Library	Opens a wizard that lets you import the content of a GSD Library export file (.glb) to the local computer.

NOTE: Using the commands requires that the latest version of the PRM Gateway DTM is installed on the computer. Visit se.com to download it. Ensure that **GSD Path** in the **User Preferences** window, page 524 points to the folder where the PRM Gateway DTM installed the tool.

Device Properties to Be Configured Outside of the DTM

You must configure the values of the following attributes of the device by using the **PROPERTIES** pane.

The table describes which values are propagated to the DTM and their state depending on the type of DTM that the device uses.

Attribute configured in the PROPERTIES pane	Third-party DTM	Schneider Electric DTM
Device Name and Part Number	Values are propagated and are read-only.	Values are propagated and are read-only.
Communication card Reference (if applicable)		
Ethernet Protocol ⁽¹⁾	Not propagated.	Values are propagated if supported by the DTM and are read-only.
IP Address ⁽¹⁾	Value is propagated and remains editable ⁽²⁾ .	
Subnet Mask ⁽¹⁾	Values are not propagated.	
Gateway Address ⁽¹⁾		
DHCP configuration information ⁽¹⁾		
(1) For embedded interfaces of the device and its communication cards (if applicable).		
(2) The values in PROPERTIES pane and in the DTM must be identical.		

Importing Devices That Use a DTM

If you configure parameters inside the DTM of a device and export this device, the values that you configured are retained when you import the device.

Opening the DTM of a Device and Editing Values

To open the DTM and edit values, proceed as follows.

Step	Action
1	Right-click a compatible device and select Open . Result: The DTM opens in a new window.
2	Edit the values as required and apply them (if applicable).
3	Close the DTM from its tab. Result: Depending on the DTM, the DTM closes and changes are saved or a confirmation dialog box opens that prompts you to save changes before closing.

NOTE: Only one user can edit parameters in the DTM of a device at a time. Editing the attributes of a device in the **PROPERTIES** pane is not allowed while the DTM is open.

Viewing DTMs Used by Devices of the Topology

Overview

The **Manage DTMs** command, page 513 lets you compare the DTMs that are used by devices of the system topology with those installed on the local computer.

It does not apply to devices of the **EPE Managed Devices** product range.

The command is available from the **Tools** menu in the toolbar of the **Topology Explorer**.

Results of the analysis are shown in the **IN USE** tab of the **MANAGE DTMs** window.

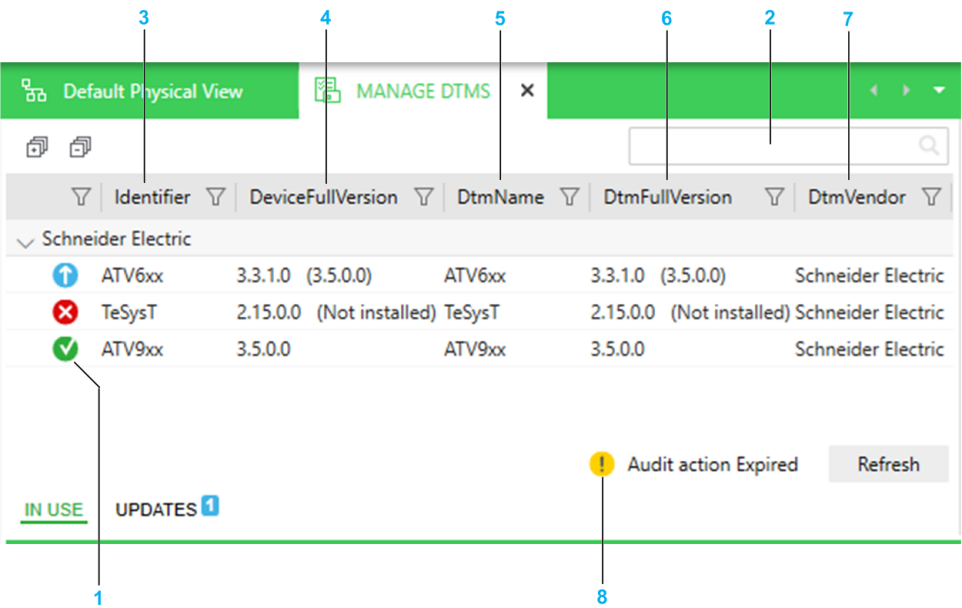
Prerequisites

To use the functionality, the following conditions must be fulfilled:

- The DTM **Hardware Catalog** is up-to-date.
- The device that is using the DTM must exist in the topology and the DTM was opened from the device once.

IN USE Tab Description

The following figure shows an example of the **IN USE** tab.



Item	Description
1	DTM status icons.
2	Search field. Enter one or more strings separated by a space and the window displays objects that contain any of these strings. In addition, you can sort and filter the contents of each column/category.
3	Device that uses the DTM. NOTE: If the DTM of an existing device has not been opened yet, it does not appear in this window.
4	Version of the DTM that is required by the device.
5	Name of the DTM.
6	Version of the DTM in the local Hardware Catalog .
7	DTM vendor. By default, DTMs are shown grouped by DTM vendor.
8	Notification indicating that the view is out of date and must be refreshed, for example, because you have created a device that uses a DTM while the MANAGE DTMs window was open. NOTE: The notification does not appear if you open the DTM of a device the first time while the window is open. In this case, reopen the window to refresh the view.

DTM Status Icons

The table describes the meaning of the icons that appear in the **MANAGE DTMs** window.

Icon	Description
	The DTM used by the device is compatible with a DTM installed on the computer. NOTE: This icon is not available in the UPDATES tab.
	A later version of the DTM is available in the local Hardware Catalog . The version that is used by the device can be updated, page 636.
	The DTM used by the device is not installed on the computer or, the DTM version installed on the computer is earlier than the version used by the device. NOTE: This icon is not available in the UPDATES tab.

Updating DTMs Used by Devices

Overview

The **Manage DTMs** command, [page 513](#) lets you see which devices of the system topology have a later version of their DTM installed on the local computer and updated in the local **Hardware Catalog**.

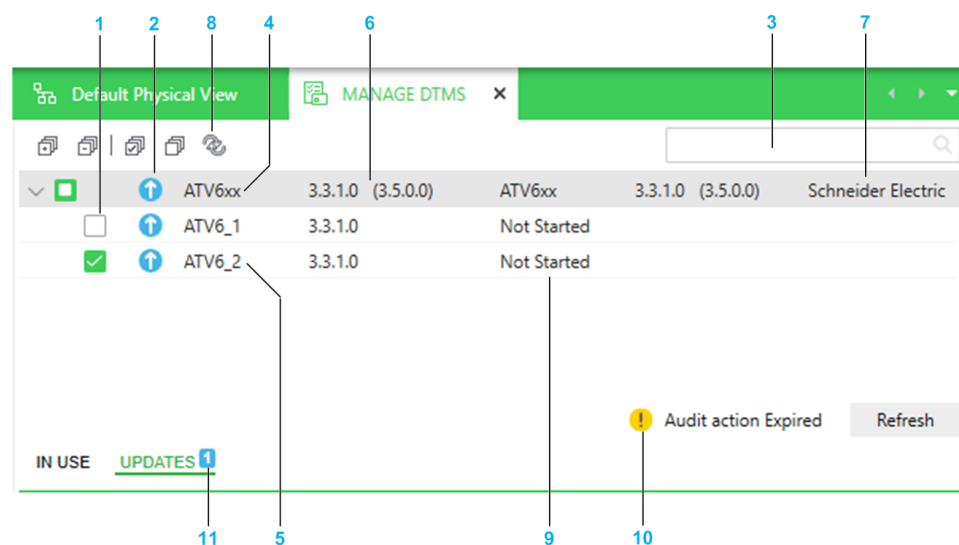
It does not apply to devices of the **EPE Managed Devices** product range.

The **UPDATES** tab lets you select which devices you want to update.

Information about the result of the DTM update process is available in the notification panel.

UPDATES Tab Description

The following figure shows an example of the **UPDATES** tab in which one device was selected to update the DTM that it uses.



Item	Description
1	Check boxes to select a DTM and all devices using it or individual devices.
2	DTM status icons, page 634.
3	Search field. Enter one or more terms separated by a space and the window displays objects that contain any of these terms.
4	Name of the DTM.
5	Name of the device that uses the DTM. NOTE: If the DTM of an existing device has not been opened yet, it does not appear in this window.
6	Version of the DTM that is used by the device and, in parenthesis, the later version of the DTM in the local Hardware Catalog .
7	DTM vendor.
8	Button to start the DTM update process for the selected devices after you confirm the command.
9	Status of the DTM update operation for each device. Possible values: <ul style="list-style-type: none"> • Not Started • In Queue • Updating • Updated (updated devices are automatically removed from the view.)
10	Notification indicating that the view is out of date and must be refreshed, for example, because you created a device that uses a DTM while the MANAGE DTMS window was open.
11	Number of DTMs that can be updated.

Implementing Communication With Non-EPE Managed Devices and Controllers

Overview

This topic describes how to configure I/O profiles of devices and controllers to implement communication by using implicit messaging (I/O scanning).

The information in this topic does not apply to devices of the **EPE Managed Devices** product range.

Configuring I/O Profiles of Devices

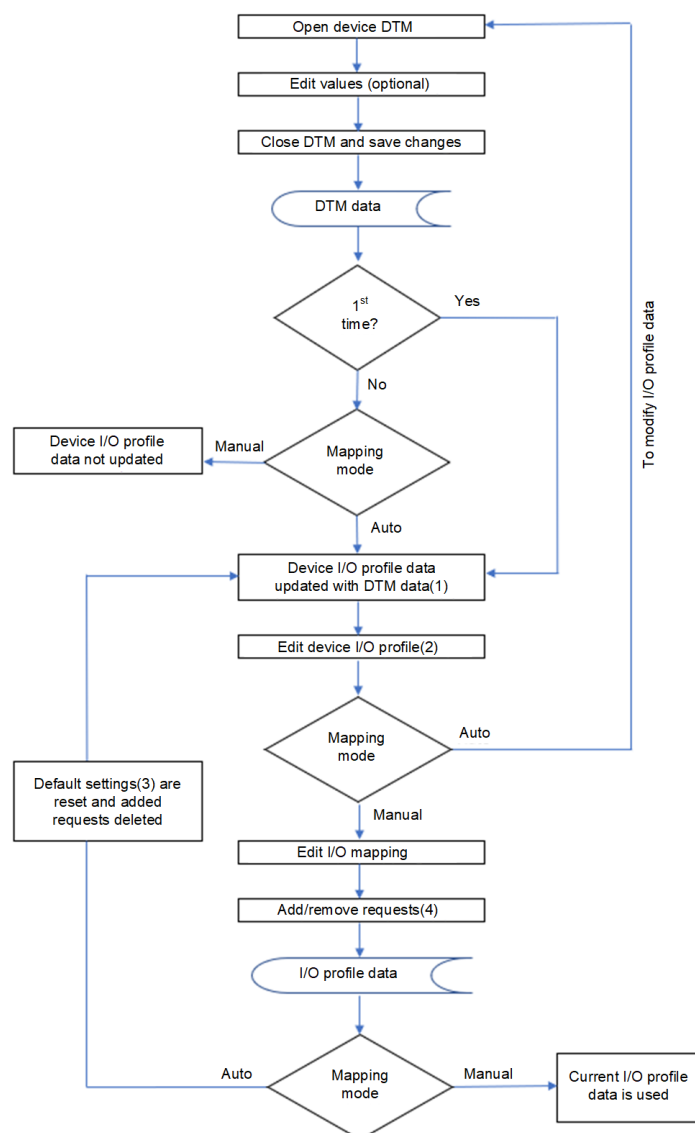
Overview

For devices that are configured by using a DTM, [page 629](#), the mapping mode, [page 709](#) is set to **Automatic** and properties of the device I/O profile are set by the DTM.

Depending on the device, you may be able to modify the properties by using the manual mapping mode.

Configuration Workflow

The following workflow shows the relation between the DTM of a device and its I/O profile depending on the setting of the mapping mode.



- (1) The first time, it may create one request or connection that cannot be deleted.
- (2) Properties that can be edited independently of the mapping mode. This includes editing **attributes**, page 709 of requests/connections that are indicated as **Default**, editing profile and instance name, adding and removing I/O profiles, and changing the mapping mode.
- (3) The values of the attributes of requests/connections that are indicated as **Default** are reset to DTM/default values if you switch the mapping mode from manual to automatic.
- (4) If adding requests/connections from the **PROPERTIES** pane is supported by the device.

Editing Properties of Active I/O Profiles

If an I/O profile is active, page 641 and you select it in the Communication Mapping View, you can edit the values of the following attributes, page 709 in the **PROPERTIES** pane to customize the requests or connections of this active profile and the associated device DDT. This does not modify the default values of the I/O profile of the device:

- For Modbus TCP:
 - **UnitID**
 - **Health Timeout (ms)**
 - **Repetitive Rate (ms)**
 - **Last Value**
 - **Gateway/Bridge Device**
- For EtherNet/IP:
 - **Timeout Multiplier**

In an active I/O profile, the values of the other attributes are read-only and you cannot create new requests or connections.

Conversely, editing the attributes of the I/O profile of a device (including requests and connections that you add or remove) automatically modifies the active I/O profile except for the protocol-specific attributes listed above. To apply the values of these attributes to the active I/O profile, use the **Reset To Default** command, page 544.

Configuring I/O Scanning of Devices and Local Slaves

Overview

This topic describes how to add one I/O profile, page 709 of a device (except devices of the **EPE Managed Devices** product range) to an I/O scanner of an M580 controller by using the Communication Mapping View, page 544 that corresponds to the communication protocol, page 679 of the device.

Adding the I/O profile of a device to a scanner creates an instance of this profile that is the *active* I/O profile. It enables communication between the two devices.

For each active I/O profile, a device DDT structure is created in the configuration project, which lets you use the device data, page 455 in the program. The structure of the device DDT is based on the item mapping, page 642.

The information also applies to the I/O profile of local slaves, page 709 added to the I/O scanner of a controller.

NOTE: For redundant M580 controllers, communication mapping and local slaves are configured only for modules of rack A.

Prerequisites

The following are prerequisites for establishing communication:

- I/O profile data of the device must be available in the **I/O PROFILE** tab of the **PROPERTIES** pane.
- For scanning of local slaves, at least one local slave, page 709 must be configured.

Scanner Capacity Limit Notification

In the Communication Mapping View for Modbus TCP, you can add I/O profiles to a scanner even if the maximum number of requests supported by the scanner is exceeded. However, in this case:

- You cannot configure the controller and a message is shown.
- An incorrect configuration is detected when you analyze the topology, page 1091.

Adding I/O Profiles to an I/O Scanner

Step	Action
1	Open the appropriate Communication Mapping View.
2	<p>From the TOOLBOX pane, drag one or more I/O profiles of devices to a scanner that appears in the view.</p> <p>Result: An instance of each profile is assigned to the scanner and becomes the active profile.</p> <p>NOTE: A device can only have one active I/O profile for a given scanner.</p>
3	To customize the properties of active I/O profile, page 709, depending on the attribute, select the active profile in the Communication Mapping View or the device to edit the value in the PROPERTIES pane.

Modifying the Scanner Assignment

To scan a device by using another scanner service of the controller, drag the I/O profile of the device to the other scanner.

Creating Items in Input/Output Registers

Overview

This topic describes how to create elementary, array, and bit type I/O items by using the **I/O Mapping Editor**. You can create items for read/write requests (Modbus TCP) and input/output connections (EtherNet/IP) of communication profiles that are configured for devices (except devices of the **EPE Managed Devices** product range).

You can modify various attributes of existing items including their type and offset.

The I/O mapping is reflected in the structure of the device DDT that is created, page 709 for the device in the configuration project of the scanner. For registers in which no I/O mapping is defined, an array of bytes is automatically created in the device DDT.

Prerequisites

The following are prerequisite to creating I/O items:

- You have added to the topology a device that uses the Modbus TCP or EtherNet/IP communication protocol (except devices of the **EPE Managed Devices** product range).
- In the I/O profile of the device, you have configured the read and/or write length, page 709 of at least one request or the input and/or output size, page 709 of at least one connection.
- The mapping mode, page 709 for the communication profile is set to **Manual**.

Changing Register Size and/or Address

The following table describes how to change the register size and/or address if the corresponding attribute is read-only in the I/O profile of the device. The procedure does not apply to the first request or connection of a device.

Step	Action
1	In the I/O Mapping Editor , select the Input or Output node of a request or connection. Result: The properties of the corresponding register appear in the Request or Connection section of the PROPERTIES pane.
2	Edit the value of Length and/or Address as needed. Result: The graphical representation of the register size is updated and the new value is shown in the properties of the I/O profile of the device.

Item Name Syntax

By default, item names use the syntax *BLOCKx_<Input/Output><Type abbreviation><Offset>_<Array elements>*, where:

- *BLOCKx* is the name of the register, page 547 that is addressed by the request/connection.
- *<Input/Output>* corresponds to I for a read and to Q for a write request/connection.
- *<Type abbreviation>* is defined in the table.
- *<Offset>* is expressed in bytes except for single bit items, for which it is expressed in bits.
- *<Array elements>* is the number of elements (for arrays only).

For example, *BLOCKA_IW24_2* is the name of an item of the first read request/ input connection of type Array of Word containing two words.

You can edit the name during or after item creation.

NOTE: When you create more than one item, <Offset> is replaced by an asterisk (*). If you edit the item name, insert one or more asterisks at any position after the first letter. The asterisk is replaced by an incremental number when the items are created and appear in the **I/O Mapping Editor**.

NOTE: Only default names are automatically updated when you modify, page 644 an item.

Data type	Type abbreviation	Designation
Bit	X	Bit
Byte ⁽¹⁾	B	Byte
String ⁽¹⁾	S	String
Word	W	Word
Int	I	Integer
UInt	UI	Unsigned integer
DWord ⁽¹⁾	DW	Double word
DInt	DI	Double integer
UDInt	UD	Unsigned double integer
Real	R	Real
Time ⁽¹⁾	T	Time
(1) The data type is not available for Modbus TCP communication.		

NOTE: An array of a type uses the same abbreviation as the type.

Creating Elementary or Array Type Items

To create elementary or array type items, proceed as follows.

Step	Action
1	In the I/O Mapping Editor , right-click Input , Output , or a selection of free consecutive bytes in the graphical representation and select Define Items . Result: The Item Definition dialog box opens. NOTE: For Modbus TCP communication, in the graphical representation, you must select an even number of bytes to be able to define items.
2	Edit Name if required.
3	Select the type of the item. To create an array, select ARRAY . NOTE: If you selected the command from the graphical representation, only types that are compatible with your selection are available.
4	Select the offset of the item and how many items to create. NOTE: Offset and Count are read-only when you create items by using the graphical representation. EcoStruxure Process Expert creates as many items as your selection allows starting at the lowest selected byte.
5	Click OK . Result: If sufficient space is available, the items are created and appear in the editor; otherwise, a notification is shown.

Creating Bit Items

To create one or more single bit items, first, create an item of a data type that supports bit mapping.

The data types are Byte, Word, Int, and UInt.

To create bit items, proceed as follows.

Step	Action
1	In the I/O Mapping Editor , right-click an existing item that supports bit mapping in the tree view or free bits in its bit field and select Define Items . Result: The Bit Item Definition dialog box opens. NOTE: To create several non-consecutive bits, select each bit in the bit field of the item.
2	Edit Name if required.
3	Select the position of the first bit item in the bit field (Offset) and how many items to create. NOTE: Offset and Count are read-only when you create bit items by using the bit field. EcoStruxure Process Expert creates as many bit items as you have selected bits.
4	Click OK . Result: If sufficient space is available, the items are created and appear in the editor; otherwise, a notification is shown.

Modifying Items

To modify an item, select it in the tree view or graphical representation and edit its attributes in the **PROPERTIES** pane.

The following table indicates which attributes you can modify.

Attribute	Elementary types	Array types	Bit types
Name	Yes ⁽¹⁾		
Comment	Yes		
Type	Yes ⁽²⁾	Yes ⁽²⁾	No
Offset	Yes	Yes	No
Rank	N/A	N/A	Yes
Number of elements	N/A	Yes	N/A
(1) You can only use unique names within a communication profile.			
(2) Selecting a type that does not support bit mapping removes existing bit mappings. You cannot change an elementary type to an array and vice versa.			

Deleting Items

To delete one or more items, select them and press **Del**.

Deleting an item also deletes its bit mappings, if applicable.

Data Exchange with Local Slaves - Example

Overview

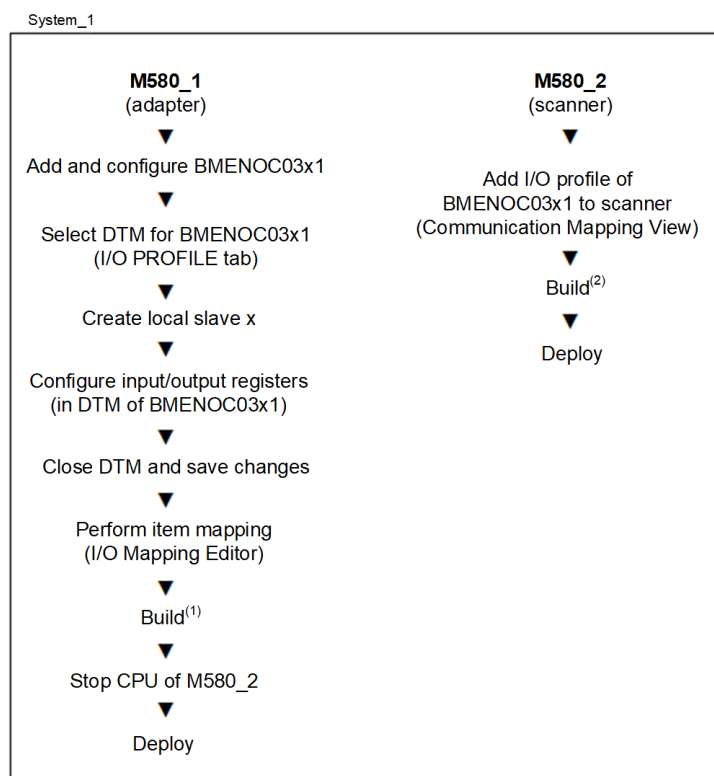
By using an example, this topic shows the workflow to implement data exchange between the following entities that exist in the topology of *System_1*:

- M580_1: The local rack of an M580 standalone controller in which a BMENOC03•1 communication module is configured as *local slave*, page 709.
- M580_2: The local rack of a second M580 standalone controller, which scans, page 641 the local slave.

A second example using the same entities shows the workflow to apply changes to the data that is exchanged.

Data Exchange Implementation Workflow

The first example shows the steps to perform on both controllers to implement the data exchange. It start with M580_1.



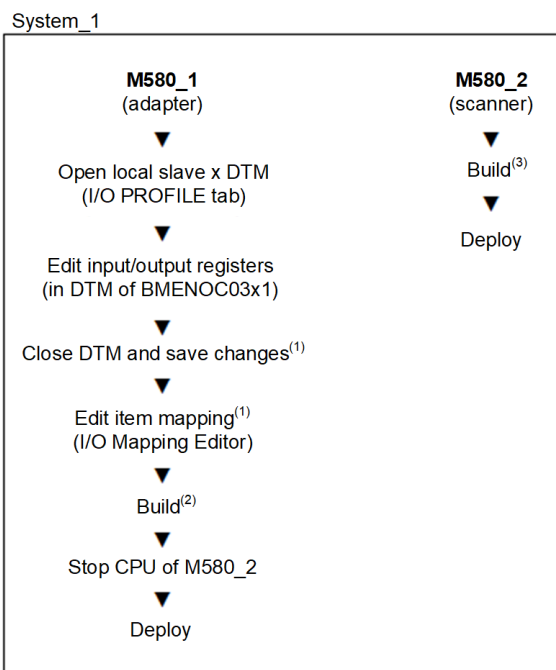
(1) Creates a device DDT named **Instance Name_LSx** in the controller configuration project. It is based on the DTM configuration of the local slave but without the mapped items.

(2) Adds the BMENOC03•1 module to the device list in the control project of the scanner and creates a device DDT named **Instance Name** with the register sizes and item mapping as defined in the I/O profile of the local slave but where:

- The input register corresponds to the output register of the local slave.
- The output register corresponds to the input register of the local slave.

Data Exchange Modification Workflow

In continuation of the first example, the second one shows the steps to perform on both controllers to apply changes to the item mapping that are performed in the I/O profile of the BMENOC03•1 local slave (for example, changing the input register size and item mapping). It starts with M580_1.



(1) Sets the build state of the executable of M580_2 to **Out Of Date**.

(2) Updates the **Instance Name_LSx** device DDT in the controller configuration project with the new input register size but not the item mapping.

(3) Updates in the **Instance Name** device DDT of the controller configuration project, the output register size, and item mapping.

Exporting and Importing Topological Entities (SBK)

Exporting Topological Entities

Overview

The **Export > Topology** command allows you to create a non-editable export file (.sbk), which contains information of selected topological entities, their folder hierarchy, and dependencies.

You can use this data with the import, page 651 functionality of the **Topology Explorer** to do the following:

- Transfer topological entities to a different system or EcoStruxure Process Expert infrastructure.
- Add and/or modify topological entities.
- Duplicate topological entities.
- Move, remove, replace, and/or add components of entities.
- Update properties of entities and folders.
- Prepare the topology of a system before importing Control and/or Supervision Participant projects.

NOTE: To export only I/O devices, page 660 to a comma-separated values (CSV) file, use the **Export > Devices** command.

Exporting M580 Safety Controllers

For information about exporting M580 safety controllers, refer to the topic describing how to engineer systems with M580 safety controllers, page 927.

Software Version Compatibility

The version of EcoStruxure Process Expert where you import an export file must be the same as the version that was used to create the export file (a service pack is considered the same version); otherwise, you cannot import the file.

Export File Contents

You can select the **Export > Topology** command at different levels to export the following:

- An entity: Exports the data pertaining to this topological entity and its components.
- A topological folder: Exports the entities contained in the folder and in subfolders.
- The **Topology** root folder: Exports the entities of the system and logical networks.

By default, at each level:

- The folder hierarchy of the exported entities is selected for export.
- If an entity appears on a [physical view, page 536](#), the view is exported. If the view contains other entities that are not exported, they are removed from the exported view. Physical links may be discarded.

You cannot export empty folders or folders that contain entities that are not selected for export. The rule does not apply for folders that are part of the parent folder hierarchy of an exported entity or folder.

NOTE: If connections to logical networks exist for a selected entity at the time of export, these links are automatically included in the export file. This allows recreating the connection at the time of import if [applicable rules are satisfied, page 654](#).

NOTE: The [password protection properties, page 592](#) and their value are exported but not the passwords, [page 804](#).

Topological Templates

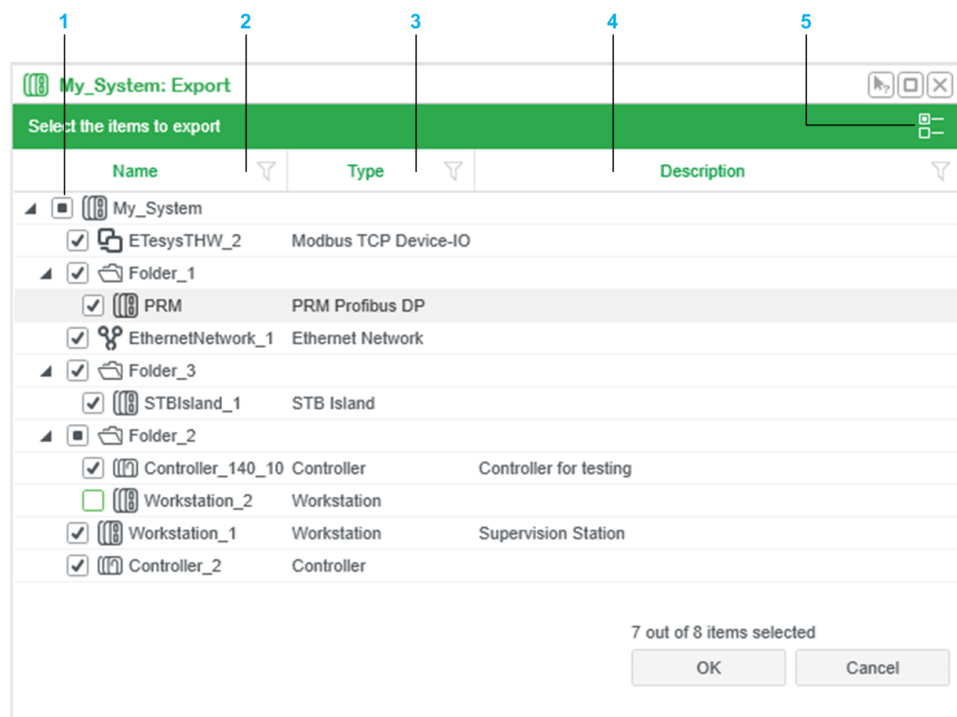
The export file does not contain the topological templates of the entities that you are exporting. To import entities in a different EcoStruxure Process Expert infrastructure, the Global Templates Library of the target infrastructure must contain the necessary templates with the same version. Entities for which the template is not present cannot be imported.

For information on how to export templates, refer to the topic that describes [managing Global Templates, page 1035](#).

Topological Export Window

The topological **Export** window allows you to select the entities and folders that you export.

The following figure shows an example of the **Export** window in tree view mode.



Item	Description
1	Items with their check box selected are exported. By default, items are selected. NOTE: When check boxes of folders are filled with a black square, at least one entity or a subfolder is not selected.
2	Name of topological entities, logical networks, and folders that you can export. NOTE: Physical views do not appear in this window.
3	Type of the topological entity (for example, controller, workstation, logical network).
4	Contents of the Description property of the entity.
5	Button to toggle between grid view and tree view.

NOTE: To expand/collapse a node in the **Export** window, select the node and press **Enter**.

Aborting Export Tasks

After confirming the export by clicking **Save** in the **Save** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The export file is not created.

Only the user who selected the command is allowed to cancel it.

Creating the Export File

To create a topology export file, proceed as follows.

Step	Action
1	Right-click the Topology root folder, a topological folder, or the entity that you want to export, and select Export > Topology . Result: The Export window opens.
2	Select the entities that you want to export.
3	Click OK . Result: The Save dialog box opens.
4	Select a location to save the export file, enter a file name, and click Save . Result: The export file (.sbk) that contains the information of the selected topological entities is created. NOTE: Click Cancel to close the Save dialog box and revert to the Export window.

Importing Topological Entities

Overview

The **Import > Topology** command allows you to import data of topological folders and entities from a topological export file (.sbk) that was created by using the export, page 276 functionality of the **Topology Explorer**.

You can select which entities to import out of those contained in the export file, except for Control Expert equipment, page 651, which is imported by default.

Before proceeding with the import, EcoStruxure Process Expert verifies if importing the selected entities creates a duplicate entity name or an IP address conflict. If necessary, the import operation includes a second step, which allows you to resolve conflicts either:

- Skip the entity; it is not imported.
- Update the entity of the target system with the information of the export file.
- Create the entity with a different, unique name.

The notification panel displays information on the tasks that completed successfully and those that could not be completed.

Applicable import rules are described in this topic. Examples are given to illustrate various scenarios, page 657.

NOTE: You cannot import an entity if in the target system the dependency of the entity is locked by another operation or user. For example, if in the target system the **Hardware Mapping Editor** is open, you cannot import into the topology entities that are linked to the logical network because it is locked by the hardware mapping process. However, you can import a workstation that is not assigned to the logical network.

Importing M580 Safety Controllers

For information about importing M580 safety controllers, refer to the topic describing how to engineer systems with M580 safety controllers, page 927.

Software Version Compatibility

The version of EcoStruxure Process Expert where you import an export file must be the same as the version that was used to create the export file (a service pack is considered the same version); otherwise, you cannot import the file.

Topological Templates

The export file does not contain the topological templates of the exported entities. If you are importing entities in a different EcoStruxure Process Expert infrastructure and the template that is required for an entity is not present, you cannot import the entity.

Information on the required templates and template version is indicated in the **Import** window.

Importing Control Expert Equipment

Control Expert equipment consists of entities of the following product types (**TOOLBOX CATALOG** pane):

- **Automation Devices** except **EPE Managed Devices**
- **Network Fieldbus Devices**
- **Network Infrastructure Devices**

If these entities were exported, they are not displayed in the Import window but are imported.

Network Connection Types

Starting with EcoStruxure Process Expert 2023, a [physical connection](#), page 615 refers to the connection of an M340 or Quantum controller, or a PROFIBUS Remote Master (PRM) to a logical network that is created and managed by using the **Physical Connections** command.

Other entities have logical network connections that are created and managed in the **PROPERTIES** pane.

Identification of Conflicts

The following table describes the meaning of the tooltips and color coding that are shown in the **Import** window, page 655 to indicate the status of entities and folders in relation to the target system.

Identification	Tooltip text	Description
Tooltip	OK	An entity of the same type with the same name does not exist in the topology of the target system. If you select if for import, the entity is created.
Tooltip	Entity already exists	An entity of the same type with the same name and with an identical configuration already exists in the topology of the target system. If you select if for import, the entity is created but with a different name to comply with the naming rule of the topology.
Tooltip	Conflict to be resolved	<p>An entity of the same type with the same name already exists in the topology of the target system but EcoStruxure Process Expert detected a change, which can, for example, be related to:</p> <ul style="list-style-type: none"> • Properties. • Devices of entities: Properties, position of modules on the rack, configuration, controller module, or module type. • Services of workstations. • (Physical) connections to logical networks. <p>To import the entity, you must select an action for it.</p> <p>NOTE: Refer to the topic describing import rules for conflicting entities, page 653.</p>
Yellow dot/IP	IP conflict	<p>At the time the file was exported, the entity was connected to a logical network that exists in the target system and the IP address of the entity is already used on this network.</p> <p>The tooltip indicates <conflicting IP address in the target system> - <path to the entity in the target system> - <logical network name in the target system>.</p> <p>After import, the entity is disconnected from this logical network. If the entity was connected to the logical network by using the PROPERTIES pane, its Logical Network attribute is set to NO CONF.</p> <p>The IP addresses of the entity are preserved.</p> <p>NOTE: If, in addition, the entity has the same name and is of the same type as an entity in the target system, the IP address conflict is not indicated.</p>
Red dot/T	Template conflict. Unable to find the target template. <template identifier – version>	<p>The entity that you want to import uses a template that is not present in the Global Templates Library. You must import the template before you can import the entity.</p> <p>For information on how to import templates, refer to the topic that describes managing Global Templates, page 1035.</p>

NOTE: Conflicts related to [Control Expert equipment](#), page 651 and physical views are not shown. If an equipment or a physical view with the same name exists in the target system, the imported object is renamed by using the **_n** suffix where *n* is an incremental number.

General Import Rule for Non-Conflicting Entities and Folders

Each entity that is contained in the export file and that you selected for import is created in the topology of the target system if no conflict is detected.

Depending on the import path setting that you select, folders may be created to recreate the original hierarchy.

If a folder with the same name already exists in the target system, its properties (for example, its comment) are automatically updated with the configuration of the export file.

Import Rules for Conflicting Entities

The following table describes the possible impact for various types of entities if EcoStruxure Process Expert detects a conflict, depending on the action that you select. The information applies if the export file contains an entity with the same name as an entity of the same type in the target system. In addition, the imported entity must have a different configuration than the target entity. The **Update** action applies to the entity in the target system.

Entity type	Create action	Update action	Comments
Workstation	Creates the entity with a new name.	<ul style="list-style-type: none">Services are updated.Properties of the entity and its services are updated.The entity is moved if the path in export file differs.	Removing a service from a workstation may impact existing service mappings of related executables.
Controller		<ul style="list-style-type: none">Properties of the entity and its modules are updated.Position of modules are updated.Modules are added, replaced, and/or removed.The entity is moved if the path in export file differs. <p>NOTE: The action is not allowed if the controller is not of the same hardware platform, page 592.</p>	<p>Moving, adding, replacing, or removing a module of a controller may impact existing service, hardware, and/or communication mappings of the related executables.</p> <p>This includes:</p> <ul style="list-style-type: none">Replacing the controller module of a controller.Changing a standalone controller into a redundant one or vice versa.
STB island ⁽¹⁾		<ul style="list-style-type: none">Properties of the entity and its devices are updated.Position of modules are updated.Modules are added and/or removed.The entity is moved if the path in export file differs.	You can overwrite parameters, such as: <ul style="list-style-type: none">IP address.Subnet mask.Explicit parameters that are defined in the corresponding templates.
PROFIBUS Remote Master (PRM) ⁽¹⁾			
Modbus TCP or EtherNet/IP device ⁽¹⁾			
Logical network		<ul style="list-style-type: none">Properties are updated.The entity is moved if path in export file differs.	<p>Updating an I/O device may impact existing hardware and/or communication mappings of the related executables.</p> <p>Existing connections to logical networks in the target system are not preserved.</p> <p>Also refer to the topic describing import rules for connections to logical networks, page 654</p>
(1) Device of the EPE Managed Devices product range			

NOTE: Also refer to the topic describing import rules for connections to logical networks, page 654.

Import Rules for Connections to Logical Networks

The following table describes the impact on the network connection of entities and the applicable conditions if you import into a system one or more entities that are connected to one or more logical networks.

Entities	Result after import	Conditions to be fulfilled
M580 controllers with the following modules: <ul style="list-style-type: none">• BME•58••••, BME•58••••S• BMENOC03•1• BMENUA0100• BMENOR2200	Entities remain connected to the networks ⁽¹⁾	If logical networks with the same name do not exist in the target system: <ul style="list-style-type: none">• The logical networks are included in the export file.• The entities and the logical networks are imported successfully with their original name.
M340 controllers		If logical networks with the same name exist in the target system, no IP address conflict is detected.
Quantum controllers		
Devices of the EPE Managed Devices product type		
Workstations		
Other entities	Entities are disconnected from the networks ⁽¹⁾	—
(1) If an entity was connected to the logical network by using the PROPERTIES pane, if it is disconnected, its Logical Network attribute is set to NO CONF .		

IP addresses of entities are preserved.

NOTE: After the import completes, perform an [analysis of the topology](#), page 1091 and make the necessary corrections to IP addresses and/or logical network connections. These corrections may impact existing hardware and communication mappings of the related executables.

Password Management After Importing Topological Entities

After you import into a system a controller and/or a workstation for which [password protection](#), page 592 is enabled, you must do the following to deploy a Control Participant project:

- If no password was set, you must set one.
- If a password was set, you must enter it when prompted.

For more information, refer to the topic describing how to manage [passwords of deployed Control Participant projects](#), page 804.

NOTE: The requirements apply even if you are importing the topological entities into the same system from which they were exported.

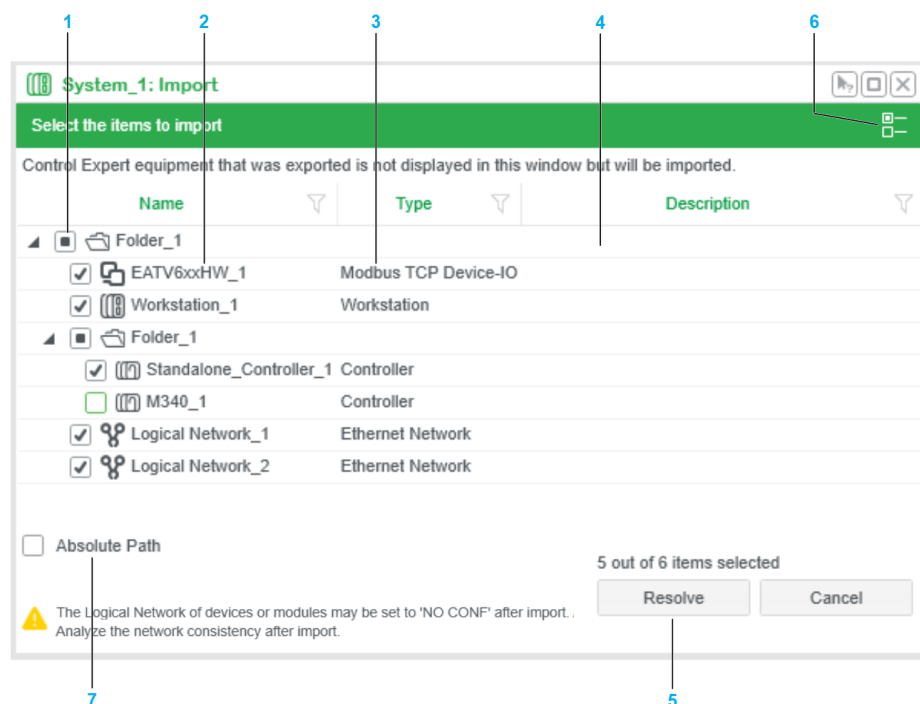
Opening the Topology Import Window

To open the topology **Import** window, proceed as follows.

Step	Action
1	Right-click the Topology root folder or a topological folder in the SYSTEM PROJECT pane and select Import > Topology . Result: The Import dialog box opens.
2	Browse to the topology export file (.sbk) that you want to import and click Open . Result: The Import window opens and displays the contents of the file. Control Expert equipment and physical views are not displayed.

Topology Import Window Description

The following figure shows an example of the **Import** window displaying the contents of a topology export file.



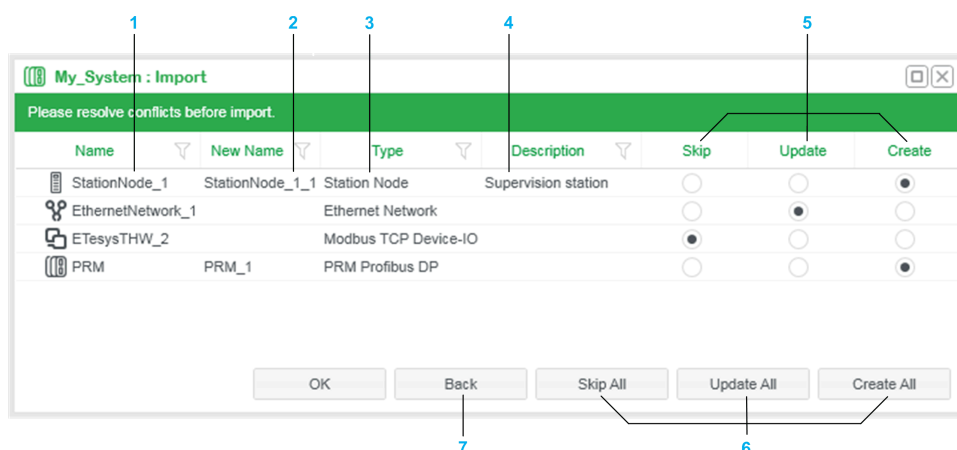
Item	Description
1	<p>Check boxes to select the folders and/or entities to import.</p> <p>By default, items are selected.</p> <p>NOTE: A check box filled with a black square indicates that at least one item in a sublevel is not selected.</p>
2	<p>Name of the folders and entities contained in the export file.</p> <p>Entities for which a conflict was detected are identified, page 652.</p> <p>In tree view mode, the column shows the hierarchy of the entities at the time of export. The hierarchy shown may be different depending on the import path setting.</p> <p>NOTE: Control Expert equipment, page 651 and physical views, page 648 are not displayed but are imported.</p>
3	Type of the entity.
-	<p>Path column that indicates the path of the folder or entity.</p> <p>The column is displayed only in grid view.</p>
4	Description that comes from the properties (comment) of the entity or folder contained in the export file.
5	<p>OK or Resolve button.</p> <p>Either button is displayed depending on the status of the instances selected for import in relation to the target system:</p> <ul style="list-style-type: none"> OK: No duplicate name conflicts are detected and selected instances are created with their original name. Resolve: At least one duplicate name conflict was detected for an instance selected for import. After clicking the button, a new import window opens, page 656 that displays only entities for which a conflict was detected. <p>The other entities for which no conflict was detected remain selected for import.</p>

Item	Description
6	Button to toggle between grid view and tree view.
7	<p>Import path setting check box:</p> <ul style="list-style-type: none"> Cleared (default): The folder where the import is performed becomes the parent folder of the imported folder structure. The hierarchy of the imported folder structure is retained. Selected: The hierarchy that existed at the time of export is preserved. Folders are created as needed to recreate the exported hierarchy. <p>NOTE: The check box is not displayed if you import to the system root folder.</p>

NOTE: To expand/collapse a node in the **Import** window, select the node and press **Enter**.

Resolving Detected Conflicts

The following figure shows an example of the **Import** window that is displayed if conflicts for the selected entities are detected and you click **Resolve** in the first import window.



Item	Description
1	<p>Name of entities that you selected for import and for which a name and/or IP address conflict was detected.</p> <p>The type of conflict is identified, page 652.</p> <p>NOTE: No conflicts are shown for Control Expert equipment, page 651 and physical views. If an equipment or a physical view with the same name exists in the target system, the imported object is renamed by using the <i>_n</i> suffix where <i>n</i> is an incremental number.</p> <p>NOTE: Only grid view mode is available in this window.</p>
2	<p>New name that will be used to create the entity if you select the Create action for the entity. The name of the entity is automatically changed by adding the <i>_n</i> suffix where <i>n</i> is an incremental number starting at 1.</p> <p>For example, if the original name of the instance is <i>Controller_1</i>, the new name is <i>Controller_1_1</i>.</p>
3	Type of the entity.
4	Description that comes from the properties (comment) of the entity or folder contained in the export file.
5	<p>Buttons that allow you to select an action for individual entities:</p> <ul style="list-style-type: none"> Skip: The entity is not imported. Update: Default action. The entity that has the same name in the target system is updated, page 653 with the configuration contained in the export file. Create: The entity is created in the target system with the name indicated in the New Name column.

Item	Description
6	Buttons that allow you to select an action for the conflicting entities as a whole. The actions are the same as for individual entities. NOTE: Clicking either of these buttons overrides the selection you made for individual entities.
7	Button to revert to the first import window. NOTE: Changes you make in this window are retained if you click Back and Resolve again.

Aborting Import Tasks

After confirming the import by clicking **OK** in the **Import** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. No entities are imported.

Only the user who selected the command is allowed to cancel it.

Importing Topological Entities

To import topological entities into an existing system, proceed as follows.

Step	Action
1	In the Import window, page 654, select the entities that you want to import.
2	Select an import path setting.
3	If a name conflict was detected, click Resolve , otherwise proceed to step 5. Result: A new import window opens, which displays entities with a conflict.
4	Select the action that you want to perform for each conflicting entity.
5	Click OK . Result: <ul style="list-style-type: none"> The topology of the target system is updated based on the import rules and your selection. An import summary appears in the notification panel.

Examples

This topic gives examples of various import scenarios of an entity into the topology of a system. The entity is not a **Control Expert** equipment, page 651.

The following table gives an example of an imported entity that creates a name conflict. The **Update** action is selected in the second step of the import.

Item	Entity in the target system	Imported entity	Result after import
Entity name	Workstation_1	Workstation_1	Workstation_1
Services	Control Expert	Supervision	Supervision
Properties	Description: Station1	Description (comment): Station2	Description (comment): Station2
Folder properties	None	Description (comment): Control Room	Description (comment): Control Room

The following table gives two examples of an imported entity that does not create a name conflict. In the second example, an IP address conflict is detected. In both cases, you can import Entity_2 in one step. Network_1 exists in the target system.

Item	Entity in the target system	Imported entity	Result after import	
Entity name	Entity_1	Entity_2	Entity_1	Entity_2
IP address	IP1	IP 2	IP 1	IP 2
Connection to Network_1	Not connected	Connected at time of export	Not connected	Connected
Entity name	Entity_1	Entity_2	Entity_1	Entity_2
IP address	IP 1	IP 1	IP 1	IP 1
Connection to Network_1	Connected	Connected at time of export	Connected	Not connected

The following table gives two examples of an imported entity that creates a name conflict but without an IP address conflict. The action that is selected in the second import window is indicated. Network_1 exists in the target system.

Item	Entity in the target system	Imported entity	Result after import	
The Update action is selected				
Entity name	Entity_1	Entity_1	Entity_1	
IP address	IP 1	IP2	IP2	
Connection to Network_1	Connected	Not connected at time of export	Not connected	
Entity name	Entity_1	Entity_1	Entity_1	
IP address	IP 1 (not used on Network_1)	IP2 not used on Network_1 of target system	IP2	
Connection to Network_1	Not connected	Connected at time of export	Connected	
The Create action is selected				
Entity name	Entity_1	Entity_1	Entity_1	Entity_1_1
IP address	IP 1	IP2 not used on Network_1 of target system	IP 1	IP2
Connection to Network_1	Connected	Connected at time of export	Connected	Not connected

The table gives two examples of an imported entity that creates an IP address conflict. In each example, a different action is selected in the second import window.

Item	Entity in the target system	Imported entity	Result after import	
The Update action is selected				
Entity name	Entity_1	Entity_1	Entity_1	
IP address	IP 1	IP2 address already used on Network_1 of target system	IP2	
Connection to Network_1	Connected	Connected at time of export	Not connected	
The Create action is selected				
Entity name	Entity_1	Entity_1	Entity_1	Entity_1_1
IP address	IP 1	IP 1	IP 1	IP 1
Connection to Network_1	Connected	Connected at time of export	Connected	Not connected

Exporting and Importing I/O Devices to/from CSV Format

Overview

The device export and import functionality of the **Topology Explorer** lets you create comma-separated value (CSV) files containing the data of I/O devices of the system topology (devices of the **Automation Devices** product type except PROFIBUS Remote Masters and STB islands).

You can use these files to import, [page 667](#) I/O devices into the topology of the same or another system and perform the following actions:

- Move or copy the I/O devices, [page 603](#), logical networks, and network connections.
- Perform bulk processes for creating and/or modifying I/O devices, logical networks, and network connections by editing the export file.

This section describes how to export I/O devices, edit the export file to perform the required changes, and import it.

NOTE: The functionality does not export other topological entities, such as controllers and workstations. To export these together with I/O devices, use the **Export > Topology** command, [page 647](#).

Exporting I/O Devices to CSV Format

Overview

You can export I/O devices (herein, devices) by using the **Export > Devices** command of the **Topology Explorer**.

The devices and their respective configuration parameters are saved to an export file in comma-separated, page 664 values (CSV).

NOTE: You must also export the topological templates, page 1039 that are used by the devices that you are exporting if the same version of template is not available in the Global Templates Library of the target system when you import. Devices whose template is not available cannot be imported.

Best Practices

To create devices by importing a CSV file, use the **Export > Devices** command from the topology of a system even if it does not contain devices. This creates an editable, page 663 CSV file that contains the necessary columns for the data that you must enter.

Using the Invariant Culture

EcoStruxure Process Expert uses the invariant culture, page 664 to format parameter values independently of the language setting that is used on the computer on which you create the export file.

For example, a decimal value that appears as 5,5 (comma separator) in the **Device Editor** because of your regional settings appears as 5.5 (dot separator) in the export file.

Software Version Compatibility

You can import devices that were exported in CSV from an earlier version.

Exported Devices

When you use the **Export > Devices** command, the following are exported:

- The devices that exist in the topology without the folder structure.
- Network connections of the devices.

NOTE: If a device is locked when you select the command (for example, because its properties are being edited), it is exported but with the last saved configuration.

Canceling Export Tasks

After confirming the export by clicking **Save** in the **Export** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The export file is not created.

Only the user who selected the command is allowed to cancel it.

Creating the Device Export File

To export devices of the topology, proceed as follows.

Step	Action
1	In the Topology Explorer , right-click the system root folder and select Export > Devices . Result: The Export dialog box opens.
2	Select the location and file name for the export file.
3	Click Save . Result: The export file (.CSV) is created and a summary of the exported devices is displayed in the notification panel.

Editing the CSV Device Export File

Overview

By editing the device export file (.csv), you can create and/or update devices, page 603 and their network connections in the topology of a system.

EcoStruxure Process Expert applies the changes when you import the edited CSV file, page 667 into the topology of the system. At that time, you can review and modify the list of imported devices.

The rules and requirements that apply when you open and edit CSV export files are described in this topic.

Best Practices

- Before editing a device export file, create a copy of it. If needed, it lets you restore the original device topology.
- Edit only an up-to-date export file to help avoid data conflicts when importing.

Opening the CSV File by Using a Spreadsheet Editor

When you open a CSV export file by using a spreadsheet editor, if either of the following applies to a device parameter value, it may be modified by the editor:

- It is of type *Duration* with millisecond values
- It contains special characters
- It contains strings starting with a hyphen (-)

To help avoid unwanted changes, import the data into a blank workbook in *Unicode (UTF-8)* format and then, proceed as follows.

NOTE: Microsoft Excel is used as an example only. Menu items and dialog box names may vary depending on your version.

Step	Action																																																															
1	Open a blank workbook.																																																															
2	In the Data menu, select to import from text or CSV.																																																															
3	In the Import Text File dialog box, select the exported device CSV file.																																																															
4	Select the following attribute values for the import of data: <ul style="list-style-type: none">• File origin: <i>65001 : Unicode (UTF-8)</i>• List separator (delimiter): <i>Comma</i>• Format for all columns (Data Type): <i>Text⁽¹⁾</i>																																																															
5	<p>Load the values to the blank workbook.</p> <p>NOTE: After loading is completed, the first row of the CSV device export file must start with <i>Action</i> (first cell). The following example shows an extract of a CSV device export file after loading values to the blank workbook.</p> <table><tr><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th><th>G</th><th>H</th></tr><tr><td>1</td><td>Action</td><td>DeviceName</td><td>Description</td><td>Template</td><td>IPAddress</td><td>SubnetMask</td><td>ID</td><td>EthernetNetwork</td></tr><tr><td>2</td><td>Create</td><td>ATV1</td><td>My ATV1</td><td>\$ETesysTHW</td><td>192.168.10.20</td><td>255.255.255.0</td><td></td><td>EthernetNetwork_1</td></tr><tr><td>3</td><td>Create</td><td>ATV2</td><td>My ATV2</td><td>\$ETesysTHW</td><td>192.168.10.21</td><td>255.255.255.0</td><td></td><td>EthernetNetwork_1</td></tr><tr><td>4</td><td>Create</td><td>ATV3</td><td>My ATV3</td><td>\$ETesysTHW</td><td>192.168.10.22</td><td>255.255.255.0</td><td></td><td>EthernetNetwork_1</td></tr><tr><td>5</td><td>Create</td><td>ATV4</td><td>My ATV4</td><td>\$EATV320HW</td><td>192.168.10.23</td><td>255.255.255.0</td><td></td><td>EthernetNetwork_1</td></tr><tr><td>6</td><td>Create</td><td>ATV5</td><td>My ATV5</td><td>\$EATV320HW</td><td>192.168.10.24</td><td>255.255.255.0</td><td></td><td>EthernetNetwork_1</td></tr></table>		A	B	C	D	E	F	G	H	1	Action	DeviceName	Description	Template	IPAddress	SubnetMask	ID	EthernetNetwork	2	Create	ATV1	My ATV1	\$ETesysTHW	192.168.10.20	255.255.255.0		EthernetNetwork_1	3	Create	ATV2	My ATV2	\$ETesysTHW	192.168.10.21	255.255.255.0		EthernetNetwork_1	4	Create	ATV3	My ATV3	\$ETesysTHW	192.168.10.22	255.255.255.0		EthernetNetwork_1	5	Create	ATV4	My ATV4	\$EATV320HW	192.168.10.23	255.255.255.0		EthernetNetwork_1	6	Create	ATV5	My ATV5	\$EATV320HW	192.168.10.24	255.255.255.0		EthernetNetwork_1
	A	B	C	D	E	F	G	H																																																								
1	Action	DeviceName	Description	Template	IPAddress	SubnetMask	ID	EthernetNetwork																																																								
2	Create	ATV1	My ATV1	\$ETesysTHW	192.168.10.20	255.255.255.0		EthernetNetwork_1																																																								
3	Create	ATV2	My ATV2	\$ETesysTHW	192.168.10.21	255.255.255.0		EthernetNetwork_1																																																								
4	Create	ATV3	My ATV3	\$ETesysTHW	192.168.10.22	255.255.255.0		EthernetNetwork_1																																																								
5	Create	ATV4	My ATV4	\$EATV320HW	192.168.10.23	255.255.255.0		EthernetNetwork_1																																																								
6	Create	ATV5	My ATV5	\$EATV320HW	192.168.10.24	255.255.255.0		EthernetNetwork_1																																																								
6	Save the workbook in <i>CSV UTF-8 (comma delimited)</i> format (keep saving in this format when you edit the file).																																																															

(1) You may need to edit the file during the import process to access the attribute (*Transform Data*).

Setting the List Separator Format

CSV export files use the comma as list separator (delimiter) independently of the setting that is in effect on the computer on which the file is created.

If the format of the list separator on the computer is different from comma, data will not be displayed correctly when you open the export file by using a spreadsheet editor.

To verify the setting and change it to comma if necessary, open the Windows **Control Panel** and click **Clock and Region > Region**. In the **Formats** tab, click **Additional settings....**

NOTE: For information on how to use commas in values of string data types in the CSV file, refer to the topic describing [editing rules](#), page 666.

Invariant Culture

Use the invariant culture to modify or add parameter values in CSV export files.

If required during import, EcoStruxure Process Expert converts the format according to the language setting that is used on the target computer.

The following table describes the formats of the invariant culture to be used in a CSV export file.

Type	Format	Example
Decimal separator	<i>x.y</i>	2.5
Time	<i>hh:mm:ss</i>	14:30:00
Date	<i>mm/dd/yyyy</i>	11/20/2014
Date Time	<i>mm/dd/yyyy hh:mm:ss</i>	11/20/2014 14:30:00
Duration	<i>DD.hh:mm:ss.ms</i>	10.12:30:10.15 (representing 10 days, 12 hours, 30 minutes, 10 seconds, 15 milliseconds)

NOTE: Some of the types described in the above table may not be used in CSV device export files.

CSV Device Export File Content Requirements

The following table describes the columns that appear in the CSV device export file, the requirements, and the impact on each one. Use this information when you edit existing data and/or add columns. The conflicts that can be detected, page 669 are described in the topic documenting the import of devices.

Header	Description	Requirement and impact
<i>Action</i>	Action that is performed on the device during import.	A column with this header and a valid value in each row are required. Possible values (not case-sensitive): <ul style="list-style-type: none"> <i>Create</i> (default) <i>Update</i>
<i>DeviceName</i>	Device name, page 31 (<i>\$Name</i>). The value is used to identify the device or automatically calculated in the case of a device creation.	A column with this header and a unique, valid value in each row are required. To update a device, the value must correspond to the name of a device that exists in the topology of the target system. NOTE: If the column contains several entries with the same value, only one device will be created at the most. It is the one in the topmost row.
<i>Description</i>	String that appears in the description of the device (comment).	A column with this header is required. The value is optional. If a row contains data, it must be valid.
<i>Template</i>	Topological template, page 584 that is used by the device.	A column with this header and a valid value in each row are required. The topological template must exist in the Global Templates Library of the target system. NOTE: EcoStruxure Process Expert automatically uses the highest template version that is available independently of the version that the device was using at the time of export.
<i>IPAddress</i>	IPv4 address of the device.	A column with this header is required. The value is optional. If a row contains data, it must be valid. The value must be in the proper IPv4 format. NOTE: To create a network connection for a device, you must enter an IP address that is available on the logical network configured for the device.
<i>SubnetMask</i>	Subnet mask that is used to define IP addresses on the logical network that the device belongs to.	A column with this header is required. The value is optional. If a row contains data, it must be valid.
<i>ID</i>	<i>Modbus</i> Unit ID of the device.	A column with this header is required. The value is optional. If a row contains data, it must be valid.
<i>EthernetNetwork</i>	Name of the logical network to which the device is connected.	A column with this header is required. The value is optional. Possible values: <ul style="list-style-type: none"> Name of an existing logical network in the topology of the target system: A connection is created between this network and the device. Name of a new logical network: The network is created in the topology of the target system and a connection is created between this network and the device. Blank: No network connection is created. In case of a device update, an existing network connection in the target topology is removed. NOTE: To create a network connection for a device, its IP address must be available on the logical network.
<i>Privilege</i>	Runtime privilege required to access the runtime service.	A column with this header is required. The value is optional. If a row contains data, it must be valid.
<i>IOSType</i>	Parameter to select the I/O scanning element to be used. For <i>Tesys T</i> devices only.	A column with this header is required. The value is optional. If a row contains data, it must be valid. Possible values: <ul style="list-style-type: none"> <i>TesysTEFast</i> <i>TesysTE</i>

Header	Description	Requirement and impact
<i>EnableDHCP</i>	Indicates whether the device subscribes to a DHCP server service.	<p>A column with this header is required. The value is optional.</p> <p>If a row contains data, it must be valid.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <i>TRUE</i> <i>FALSE</i>
<i>IdentifyByDevice-Name</i>	Indicates whether the DHCP server identifies the device by using <i>DTMDeviceName</i> or <i>MACAddress</i> .	<p>A column with this header is required. The value is optional.</p> <p>If a row contains data, it must be valid.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <i>TRUE</i>: Requires entering a value for <i>DTMDeviceName</i>. <i>FALSE</i>: Requires entering a value for <i>MACAddress</i>. <p>The value is ignored and not exported/imported if <i>Enable DHCP = FALSE</i>.</p>
<i>MACAddress</i>	MAC address of the device.	<p>A column with this header is required. The value is optional.</p> <p>If a row contains data, it must be valid and in the proper format (hyphen as separator).</p> <p>The value is ignored and not exported/imported if <i>Enable DHCP = FALSE</i> or <i>IdentifyByDeviceName = TRUE</i>.</p>
<i>DTMDeviceName</i>	Name of the device that is used by the DHCP server to identify it.	<p>A column with this header is required. The value is optional.</p> <p>If a row contains data, it must be valid.</p> <p>The value is ignored and not exported/imported if <i>Enable DHCP = FALSE</i> or <i>IdentifyByDeviceName = FALSE</i>.</p>

NOTE: Unless mentioned otherwise, for optional parameters for which no value is entered, the default value of the specified template or of the topology domain is used.

CSV Device Export File Editing Rules

The following table indicates the rules that apply when you edit a CSV device export file.

Item	Rule	
Rows of device data	Refer to the requirements, page 665. You can delete the row of a device if you do not want to import it or modify it through import.	
Empty rows	Cannot be added.	
Comment rows		
Empty columns		
Order of columns	Cannot be changed.	
List separator	Only comma, page 664 is accepted.	
Values	Use the invariant culture, page 664. To use commas or quotation marks in values of string data types when you edit the file by using a text editor, for example, Notepad, follow these rules:	
	Expected value after import	Value to be entered
	Motor, Pump	"Motor, Pump"
	Motor"Pump	"Motor""Pump"

Importing a CSV File Into the Topology

Overview

By using the **Import > Devices** command of the **Topology Explorer**, you can modify the devices of the topology with the configurations that you made in the CSV device export file.

The import process consists of the following steps:

1. Automatic validation of the CSV file structure.
2. If valid, displaying the devices contained in the imported file along with the action that you configured for each one, notifying of detected issues, and selecting which devices to import.
3. Importing selected devices that satisfy applicable rules (for example, uniqueness of device name and presence of template) or notifying the detected conflicts, page 669.
4. Displaying a summary of the import operation that includes information about incorrect configurations and devices that were not imported.

NOTE: Ensure that the following are available before you import the CSV file:

- The templates that are used by the devices that you are importing must exist in the Global Templates Library and not be open for editing; otherwise, you cannot import these devices. Refer to the topic describing how to export/import templates, page 1035.
- The DTMs, page 629 that are used by devices, if applicable, must be installed on the computers of the target EcoStruxure Process Expert infrastructure and the hardware catalog must be updated.

Software Version Compatibility

You can import devices that were exported in CSV format from an earlier version.

Device Import Behavior

The following table describes the default import behavior when you configure the export file, page 663 to *create* devices in the following scenario.

Scenario	Result
A device parameter value is invalid or no value is configured for an optional parameter.	The device is created and configured with parameters whose value is valid. For the invalid and not configured parameter, the default value of the template or topology domain is used.

The following table describes the import behavior when you configure the export file, page 663 to *update* devices in the following scenario.

Scenario	Result
A device parameter value is invalid or no value is configured.	Only parameters whose value is valid are updated. For the invalid and not configured parameter, the present value of the device is retained.

NOTE: Refer to the table describing the device export file content requirements, page 665 for additional information on specific parameters.

Canceling Import Tasks

After confirming the import by clicking **OK** in the **Import Devices** or **Resolve Uniqueness Conflicts** window, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task.

Only the user who selected the command is allowed to cancel it.

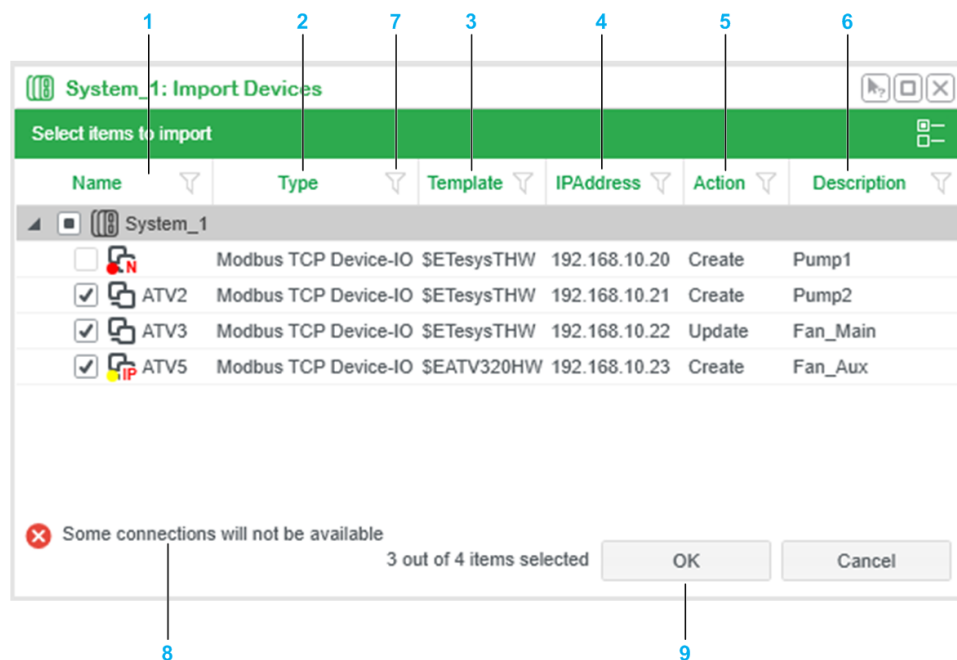
Opening the Import Devices Window

To open the **Import Devices** window, proceed as follows.

Step	Action
1	<p>In the Topology Explorer, right-click the system root folder and select Import Devices.</p> <p>Result: The Open dialog box opens.</p>
2	<p>Select the device export file (.csv) and click Open.</p> <p>Result: The Import Devices window opens and displays the content of the file.</p> <p>NOTE: The logical networks and some device parameters contained in the file are not shown.</p>

Import Devices Window Description

The following figure shows an example of the **Import Devices** window. This example illustrates a scenario where a conflict is detected for two devices and one of them cannot be imported.



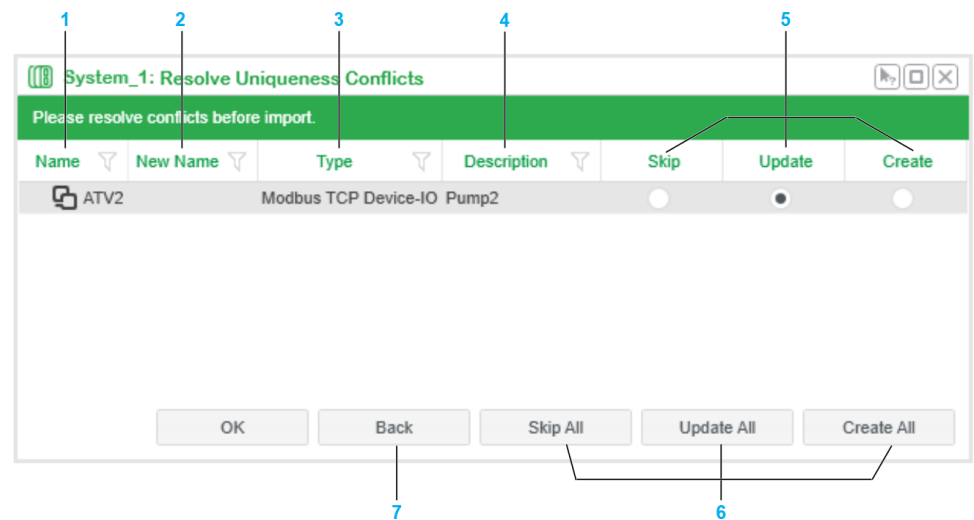
Item	Description										
1	<p>Name of the devices contained in the imported file.</p> <p>A selected check box indicates that the device is imported if possible.</p> <p>A color icon next to a device indicates a detected conflict. Details are shown in a tooltip.</p> <table border="1"> <thead> <tr> <th>Icon/ letters</th><th>Description</th></tr> </thead> <tbody> <tr> <td>Yellow dot/IP</td><td> <p>At the time the file was exported, the device was connected to a logical network that exists in the target system and its IP address is already used on this network.</p> <p>The tooltip text indicates <conflicting IP address in the target system> - <path to the device in the target system> - <logical network identifier in the target system>.</p> <p>You can import the device but network connections are impacted. For details, refer to the topic describing import rules for network connections, page 654.</p> <p>NOTE: If, in addition, the device has the same name and is of the same type as a device in the target system, the IP address conflict is not indicated.</p> </td></tr> <tr> <td>Red dot/T</td><td> <p>The device you want to import uses a template that is not present in the Global Templates Library. You must import the template before you can import the device.</p> <p>For information on how to import templates, refer to the topic that describes managing templates, page 1035.</p> </td></tr> <tr> <td>Red dot/A</td><td> <p>The <i>Action</i> parameter for the device in the device CSV export file is invalid. You cannot import the device.</p> </td></tr> <tr> <td>Red dot/N</td><td> <p>The <i>DeviceName</i> parameter of the device in the device CSV export file blank. You cannot import the device.</p> </td></tr> </tbody> </table>	Icon/ letters	Description	Yellow dot/IP	<p>At the time the file was exported, the device was connected to a logical network that exists in the target system and its IP address is already used on this network.</p> <p>The tooltip text indicates <conflicting IP address in the target system> - <path to the device in the target system> - <logical network identifier in the target system>.</p> <p>You can import the device but network connections are impacted. For details, refer to the topic describing import rules for network connections, page 654.</p> <p>NOTE: If, in addition, the device has the same name and is of the same type as a device in the target system, the IP address conflict is not indicated.</p>	Red dot/T	<p>The device you want to import uses a template that is not present in the Global Templates Library. You must import the template before you can import the device.</p> <p>For information on how to import templates, refer to the topic that describes managing templates, page 1035.</p>	Red dot/A	<p>The <i>Action</i> parameter for the device in the device CSV export file is invalid. You cannot import the device.</p>	Red dot/N	<p>The <i>DeviceName</i> parameter of the device in the device CSV export file blank. You cannot import the device.</p>
Icon/ letters	Description										
Yellow dot/IP	<p>At the time the file was exported, the device was connected to a logical network that exists in the target system and its IP address is already used on this network.</p> <p>The tooltip text indicates <conflicting IP address in the target system> - <path to the device in the target system> - <logical network identifier in the target system>.</p> <p>You can import the device but network connections are impacted. For details, refer to the topic describing import rules for network connections, page 654.</p> <p>NOTE: If, in addition, the device has the same name and is of the same type as a device in the target system, the IP address conflict is not indicated.</p>										
Red dot/T	<p>The device you want to import uses a template that is not present in the Global Templates Library. You must import the template before you can import the device.</p> <p>For information on how to import templates, refer to the topic that describes managing templates, page 1035.</p>										
Red dot/A	<p>The <i>Action</i> parameter for the device in the device CSV export file is invalid. You cannot import the device.</p>										
Red dot/N	<p>The <i>DeviceName</i> parameter of the device in the device CSV export file blank. You cannot import the device.</p>										
2	Indicates the type of the device.										
3	Identifier of the template that device is using.										
4	IP address that is configured for the device.										

Item	Description
5	Indicates the action that is performed for the selected device according to the value of the <i>Action</i> parameter in the imported file. NOTE: In the import summary that is shown in the notification panel when you create a device, the action appears as <i>Create</i> for the device but as <i>Update</i> for its parameters.
6	Content of the <i>Description</i> column.
7	You can sort and filter, page 120 data.
8	Indicates that some network connections, either configured in the imported file or that exist in the target topology, will not be available after import. This is because conflicts related to IP addresses and/or logical networks were detected.
9	OK or Resolve button. Either button is displayed depending on the status of the devices selected for import: <ul style="list-style-type: none">• OK: No conflicts are detected and selected devices are created with their original name.• Resolve: At least one conflict was detected for a device selected for import. After clicking the button, the Resolve Uniqueness Conflicts window opens.

Resolving Conflicts During Device Import

The **Resolve Uniqueness Conflicts** window helps you resolve the device-uniqueness conflicts that are detected during import.

The following figure shows an example of the **Resolve Uniqueness Conflicts** window in which one conflicting device is displayed.



Item	Description
1	Name of the device as it appears in the imported file.
2	New name of the device if you select to create it instead of updating the existing one.
3	Indicates the type of the device.
4	Content of the <i>Description</i> column.
5	Buttons that let you set an action for individual devices: <ul style="list-style-type: none"> Skip: The device is ignored for the import. Update: Default action. Modifies the existing device with the imported parameter values. The name of the device is not modified. Create: Creates a new device with New Name and the imported parameter values.
6	Buttons that let you set an action for the conflicting devices as a whole. The actions are the same as individual devices. NOTE: Clicking these buttons overrides the selection you made for individual devices.
7	Reverts to the Import Devices window. Your selection for individual devices is retained.

Importing Devices into the Topology

To import the content of a CSV device export file in the topology of a system, proceed as follows.

Step	Action
1	In the Import Devices window select the devices that you want to import.
2	If the Resolve button is displayed, proceed to step 3. Otherwise, click OK . Result: The devices are imported and displayed in the SYSTEM PROJECT pane. A summary of the import process appears in the notification panel.
3	In the Import Devices window, click Resolve . Result: The Resolve Uniqueness Conflicts window opens.
4	Select the actions that you want to perform for conflicting devices.
5	Click OK . Result: The devices that were shown in the Import Devices and Resolve Uniqueness Conflicts windows are imported and displayed in the SYSTEM PROJECT pane. A summary of the import process appears in the notification panel.

Properties of Topological Entities

What's in This Chapter

PROPERTIES and Properties Inspector Panes.....	673
PROPERTIES Pane - DESCRIPTION Tab	675
PROPERTIES Pane - CONFIGURATION Tab	676
PROPERTIES Pane - SECURITY Tab	690
PROPERTIES Pane - SERVICES Tab	694
PROPERTIES Pane - I/O PROFILE Tab	709
PROPERTIES Pane - DIAGNOSTIC Tab	717

This section describes the **PROPERTIES** pane of the **Topology Explorer** and the attributes that appear therein.

PROPERTIES and Properties Inspector Panes

Overview

The **PROPERTIES** pane lets you view and/or configure the attributes of the object that is selected.

Depending on the object that is selected, the tabs displayed and the information in each tab may vary.

NOTE: You can reopen the pane by using the **Views** menu in the toolbar of the **Topology Explorer**.

Properties Inspector Pane

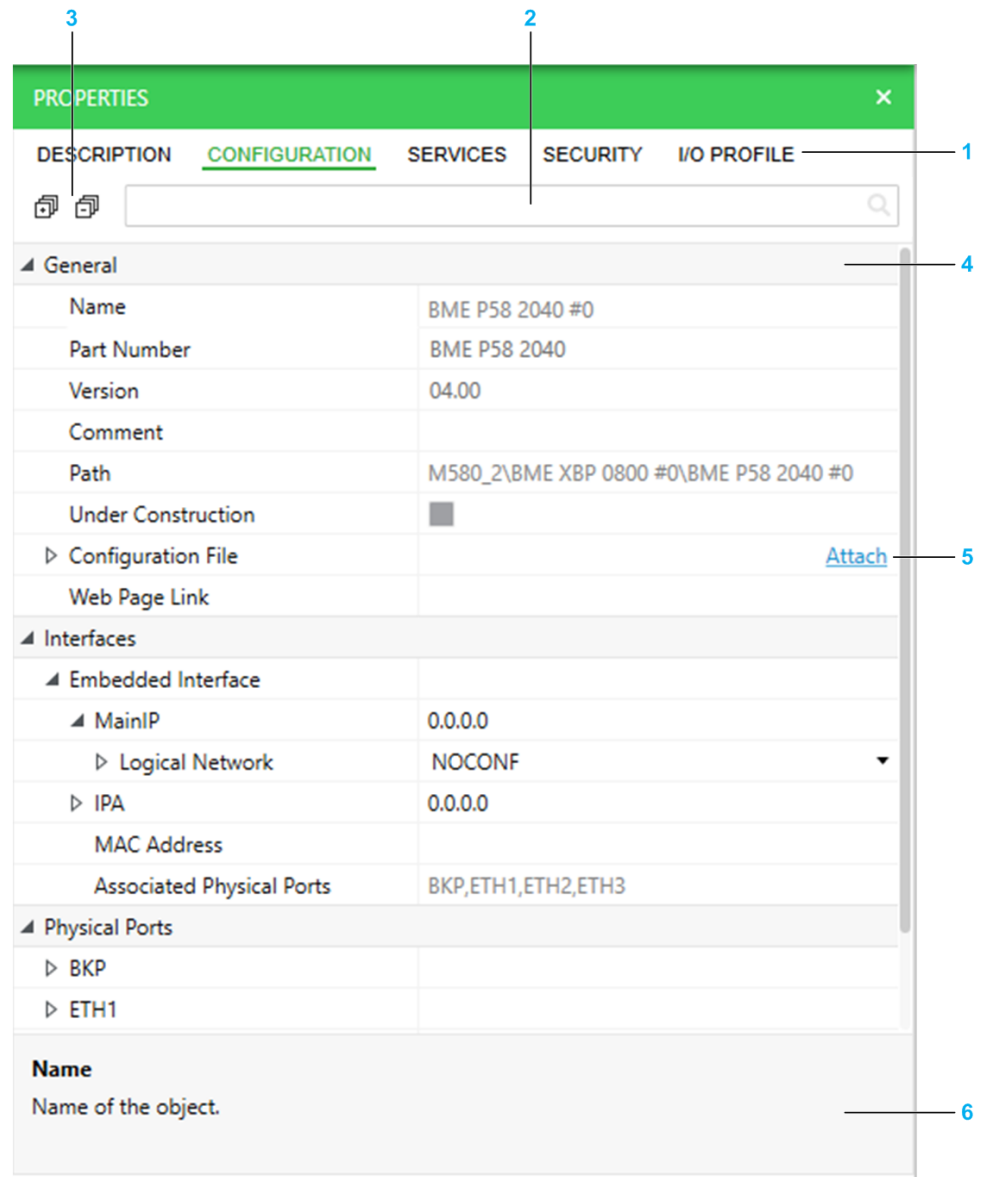
Properties Inspector panes are additional panes that you can open by using the **Open With** context-menu command of an object.

Each pane displays the same type of information as the **PROPERTIES** pane but only for the object from which you open it. The name of the object appears in the title bar of the pane.

It makes it possible to view the properties of several objects at the same time.

PROPERTIES Pane Description

The following figure shows an example of the content of the **PROPERTIES** pane when an entity is selected in the **Topology Explorer**.



Item	Description
1	Lets you access the various tabs showing the attributes of the selected object. When you select equipment from the catalog, only its description is available.
2	Search field. Enter a string to display only categories and attributes that contain the string.
3	Controls to expand/collapse attribute categories and nodes.
4	Attributes are grouped by category.
5	Controls located on the right-hand side let you attach, create, delete, and open objects that are managed in this view.
6	Name and/or description of the selected attribute.

PROPERTIES Pane - DESCRIPTION Tab

General Category - DESCRIPTION Tab

Description of Attributes

Attribute	Description
Icon	Picture of the product modeled by the entity.
Part Number	Part number of the product modeled by then entity.
Short Description	Description of the product range.
Version	Version.
Description	Description.
Online Documentation	Hyperlink to the product page on the Schneider Electric website. An Internet connection is required.

NOTE: The attributes that appear in this tab when you select the system root folder correspond to the properties of the [system](#), [page 137](#).

PROPERTIES Pane - CONFIGURATION Tab

Overview

The **CONFIGURATION** tab is available for entities that have attributes related to hardware and network services. It lets you view and/or configure their attributes, which are grouped in categories.

This topic describes the attributes that can be displayed. Depending on the entity that is selected, some may not appear in the tab.

General Category - CONFIGURATION Tab

Description of Attributes

Attribute	Default value	Description	
Name	—	Name of the object in the topology. The name must be unique and comply with applicable naming rules, page 31. NOTE: For a redundant controller, by default, both local racks have the same name but each one is identified by the <code>_A</code> or <code>_B</code> suffix that corresponds to the role of the controller. For the following objects, the suffix that is added after their name is not editable:	
		Object	Suffix
		Remote racks (drops)	The drop address. It is shown in a 3-digit format. For example, #001 for drop 1. The value is based on the configuration of <i>Device Name</i> of the drop adapter module in the Control Participant when you add the drop. Thereafter, it is updated when you edit the DHCP client Identifier , page 694 of the drop adapter module. It lets you configure the rotary switches of the module. The first two digits representing the <i>Tens</i> and the third the <i>Ones</i> switch.
		Backplanes of local and remote racks and their modules	The position (address) of the backplane in the rack and the position of the module on the backplane. The format is #<position>. For example, #0.
		Power supply modules	<Position> is replaced by <i>P</i> . For racks that support redundant power supplies, <i>PL</i> and <i>PR</i> are used to identify the two power supplies.
Part Number	—	Part number of the product modeled by the entity. For some devices that are added to the topology from the TOOLBOX , you can select the part number from a list. For others, a value appears after you select a reference, page 565 in the DTM window of the device.	
Version	—	Version of the product modeled by the entity. For some entities that are added to the topology from the TOOLBOX , you can select the version from a list.	
Label	—	Refer to the description of physical link properties, page 608.	
Source	—		
Destination	—		

Attribute	Default value	Description
Comment	Blank	Optional. You can enter a comment by using free-form text.
Path	—	Read-only. Path to the object in the SYSTEM PROJECT pane. A tooltip indicates the entire path. NOTE: For racks and modules, the controller and its parent folders are not part of the path. For fieldbus devices, the gateway object is not part of the path.
Under Construction	FALSE (cleared)	When you select the check box (TRUE): <ul style="list-style-type: none"> You cannot deploy to the entity. Attributes of the entity are not analyzed, page 1091 or only partially. No diagnostic information, page 1107 is shown. In a physical view, the entity is shown with light gray fill, page 578. NOTE: For certain entities and modules, the value is read-only and inherited from the parent object.
Configuration File	Blank	Optional. Lets you save a copy of a file to the content repository. The file can be, for example, a configuration project that is managed by an external program. Click Attach to select a file on the local computer or a network location. You can attach a file only if its file name extension is allowed (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>). Each file is saved with a unique, randomly created name. Once attached, the file can be managed from the context menu of the entity by one user at a time.
Last Attached By	Blank	User name that was entered to log into the engineering client and date/time when the file specified under Configuration File was attached last. The attributes are also updated if you open a file by using the Configuration File > Open command, page 570 and your changes are saved to the repository.
Last Attached On		
Controller Family	M340/M580	Indicates the hardware platform of the controller that communicates with the PROFIBUS Remote Master (PRM). The value must correspond to the family of the controller that you select in the service mapping tab, page 723 of the executable of the Control project. Values: <ul style="list-style-type: none"> M340/M580 Quantum
Web Page Link	Blank	Optional. Enter a URL that uses the HTTP or HTTPS protocol to open the corresponding Web page by using the Open Web Browser command, page 570. (The URL must start with <i>http://</i> or <i>https://</i> .) NOTE: Leave the attribute blank to open the embedded Web page of the module or entity (if available).
Index	—	Read-only. Address of the rack in the bus editor.

Attribute	Default value	Description
Computer Name	Blank	Optional. The value is applied to the <i>Computer</i> property in the Network Addresses table of the Supervision Participant project when you build its executable. The value is required, for example, if you use encryption with Supervision.
Runtime Access Privilege	Blank	Defines the privilege required to access the runtime service (string).

Interfaces Category - CONFIGURATION Tab

Overview

The **Interfaces** category regroups attributes related to the TCP/IP configuration of entities.

It contains at least one subcategory.

If an entity manages other IP addresses in addition to the host addresses, their attributes are shown in separate interface subcategories, which correspond to the respective ports, for example, *Port1_Interface*. Typically, enabling the routing capability of a port (**Interface** attribute, page 682) creates such a subcategory.

Embedded Interface Subcategory

The subcategory is shown for entities that manage at least one IP address. It contains the attributes of the host IP addresses. A device can manage more than one host IP address.

NOTE: If the interface of the entity is on a communication card, the name of the interface is **Ethernet Interface**.

Each interface subcategory features one or more of the following attributes.

Attribute	Description
Ethernet Protocol	Indicates the communication protocol that is used by the interface. Possible values: <ul style="list-style-type: none"> • Modbus TCP • EtherNet/IP For entities that support it, you can select either value.
Associated Physical Ports	Lists the ports, page 682 of the entity to which the interface settings (such as Ethernet protocol, IP address, logical network) apply.
MAC Address	Lets you enter the identification number of the physical device. The value is used as identifier for DHCP clients, page 694 that are identified by their MAC address. Format: <i>MM-MM-MM-SS-SS-SS</i> where <i>MM</i> (vendor identification) and <i>SS</i> (device-specific identification) are hexadecimal numbers.
Scanning Device Outside Logical Network	The attribute is available only for BMENOC03•1.4 modules. When the check box is selected (TRUE), the information message related to analysis rule ID 3008, page 1098 is displayed. NOTE: If this check box is selected, it clears the Automatic Device(s) subnet update check box in the IP address configuration tab of the module in the control project and vice versa.

Attributes of IP Addresses

IP addresses can be configured manually or automatically by assigning the entity to a logical network, page 681 or by using the **LOGICAL NETWORK VIEW**, page 541.

Each interface subcategory features one or more of the following attributes.

Attribute	Description
IP Address	IP address of the entity. NOTE: For NICs of workstations, to use the controller simulator of the local computer, enter <i>127.0.0.1</i> , which corresponds to the local host. If no system server is installed on the local computer, enter the IP address, page 801 that is displayed in the simulator panel. The logical network of the Ethernet Interface does not need to be configured. NOTE: For STB islands of the EPE Managed Devices product range, the attribute is enabled only after you configure the STB island.
MainIP	MainIP is an IP address that is available for: <ul style="list-style-type: none"> The controller module. It is used for services other than the remote I/O (RIO) scanner. Certain communication modules.
IPA	IPA is used by the RIO scanner. NOTE: For BMENUA0100 modules, IPA contains the IP address of the backplane port. It is automatically configured with an IP address that belongs to the same logical network as MainIP of the controller.
MainIP+1	Available only for interfaces in local rack B of a redundant controller. The attribute is read-only and calculated automatically based on the value of MainIP .
IPB	Available only for interfaces in local rack B of a redundant controller. Used by the RIO scanner. NOTE: For BMENUA0100 modules, IPB is the IPv4 address of the backplane port.

IPv4 Control Port Sections

The **IPv4 A** and **IPv4 B** sections and their attributes are available for BMENUA0100 modules and apply to the control port, page 682.

The **IPv4 B** section pertains to the module in local rack B of a redundant controller. Its attributes are automatically configured based on the values of attributes in the **IPv4 A** section and vice versa.

Attribute	Description
Enabled	When the check box is selected (TRUE, default value) enables IPv4 communication for the control port and shows the other attributes of the section.
IPv4 A	IPv4 address of the module. By default, an IP address is configured that belongs to a new logical network that is created.
IPv4 Mode	Identifies the source of the IPv4 address. Possible values: <ul style="list-style-type: none"> Default: Disables the Logical Network section and IPv4 A, which shows as <i>0.0.0.0</i> (default setting). The module is automatically configured with IPv4 address <i>10.10.MAC5.MAC6</i> where: <ul style="list-style-type: none"> <i>MAC5</i> is the fifth byte of the MAC address of the module. <i>MAC6</i> is the sixth byte of the MAC address of the module. Static : Enables the Logical Network section, which lets you configure a static IP address.

IPv6 Control Port Sections

The **IPv6 A** and **IPv6 B** sections and their attributes are available only for BMENUA0100 modules and apply to the **control port**, page 682.

The **IPv6 B** section pertains to the module in local rack B of a redundant controller. Its attributes are automatically configured based on the values of attributes in the **IPv6 A** section and vice versa. Except for **Gateway Address**, which you can configure independently.

NOTE: The use of IPv6 addresses is not supported and you cannot save the configuration or synchronize changes if you enable it.

Logical Network Section

For each IP address, the following attributes are available.

Attribute	Description
Logical Network	<p>Logical network that the interface of the object belongs to.</p> <p>You can assign an interface to an existing logical network or to a new one by using the list. This sets the IP address to the next available address on this logical network.</p> <p>The list contains the following entries:</p> <ul style="list-style-type: none"> • NOCONF: Not configured. Sets the IP address, Subnet Mask, and Gateway Address to 0.0.0.0. The interface of the object is not assigned to a logical network. • Existing logical networks: Assigns the interface to the logical network that you select. • New: Create a new logical network by entering its name and configuring its properties. The interface of the object is assigned to the newly created logical network. <p>NOTE: For certain interfaces, the value is preconfigured or inherited from another entity (for example, a module may have the same value as the controller module of the controller) and may be read-only (for example, for BMECRA**** adapter modules).</p> <p>NOTE: For redundant controllers, changing the value for the interface of a module in one local rack also changes it for the counterpart module in the other rack.</p>
Subnet Mask	The attribute is read-only and configured based on the configuration of the logical network that the interface of the object is assigned to (Logical Network attribute)
Gateway Address	

Physical Ports Category - CONFIGURATION Tab

<Port Name> Section

For a description of specific ports of modules of the M580 platform, refer to the next sections.

Attribute	Description
Enabled	Clearing the check box (FALSE) disables the port and its services. If possible, disable a port when it is not used. NOTE: If the attribute is not present, the port is enabled and cannot be disabled.
Interface	Selecting the check box (TRUE) enables the routing capability of the port and creates an <i>interface subcategory</i> , page 679 for the port.
Service L2	You can select None or any combination of services from the following: <ul style="list-style-type: none"> • QoS • RSTP The services that are enabled are shown in the text box when the list is collapsed.
Redundant Port	Creates a second redundant port with the same configuration.
Port Type	Indicates the type of network that the port supports. For example, a DIO (Device I/O) or RIO (remote I/O) ring. For certain entities, you can select the port type from a list. NOTE: To change the port type of several editable ports at once, select them on the entity in a physical view (without selecting the entity itself) and change the value of Port Type in the PROPERTIES pane.

BKP Backplane Port Section

Attribute	Description
Enabled	Clearing the check box (FALSE) disables the port and its services. If possible, disable a port when it is not used. NOTE: If the attribute is not present, the port is enabled and cannot be disabled.
Baud Rate	100 Mbits/s full duplex. Read-only.
Interface	Selecting the check box (TRUE) enables the routing capability (IP forwarding) of the port and creates an <i>interface subcategory</i> , page 679 for the port. NOTE: To use the routing capabilities of the communication module, configure the BKP_interface with an IP address that belongs to the same logical network as the embedded interface of the controller module.
Port Type	Indicates the type of network that the port supports. For example, a DIO ring.

ETH1 Service Port Section

For BMENOS0300 modules, the configuration of the Ethernet ports is performed exclusively by using the rotary switch located on the module. Configuring **Service L2** has no impact. For details, refer to the module installation and configuration guide.

Attribute	Description
Enabled	<p>Clearing the check box (FALSE) disables the port and its services.</p> <p>If possible, disable a port when it is not used.</p> <p>NOTE: If the attribute is not present, the port is enabled and cannot be disabled.</p>
Baud Rate	<p>Possible values:</p> <ul style="list-style-type: none"> • Auto 10/100 Mbits/s (default) • 10 Mbits/s half duplex • 10 Mbits/s full duplex • 100 Mbits/s half duplex • 100 Mbits/s full duplex • 1000 Mbits/s half duplex • 1000 Mbits/s full duplex <p>The attribute is set to the default value and read-only for controller modules.</p>
Service Port Mode	<p>Possible values:</p> <ul style="list-style-type: none"> • Access Mode • Mirroring Mode: Enables Mirrored Ports. • Extended Network: Creates the ETH1_Interface interface, page 679, which you can configure. Applies to BMENOP0300 modules only. <p>NOTE: For BMENOP0300 modules, setting ETH1 to Access Mode automatically associates the ports that are enabled (for example, BKP or ETH2) to the embedded interface of the module and configures its ETH-1 IP address with that of MainIP.</p> <p>Setting ETH1 to Mirroring Mode configures its ETH-1 IP address with <i>0.0.0.0</i>.</p>
Mirrored Ports	<p>The ports that are mirrored are shown in the text field when the list is collapsed.</p> <p>You can select any combination of ports from the following:</p> <ul style="list-style-type: none"> • Internal: Embedded interface • ETH2 • ETH3 • BKP: Backplane port <p>To be selected, a port must be enabled.</p>
Interface	<p>The attribute is available only when Interface of BKP is enabled (if applicable).</p> <p>Selecting the check box (TRUE) enables the routing capability of the port and creates an interface subcategory, page 679 for the port.</p> <p>NOTE: To use the routing capabilities of the port, configure ETH1_interface with an IP address that belongs to the extended network.</p>
Port Type	Indicates the type of network that the port supports. For example, a DIO ring.

ETH2 and ETH3 Device Network Port Sections

For BMENOS0300 modules, the configuration of the Ethernet ports is performed exclusively by using the rotary switch located on the module. Configuring **Service L2** has no impact. For details, refer to the module installation and configuration guide.

Attribute	Description
Enabled	<p>Clearing the check box (FALSE) disables the port and its services.</p> <p>The attribute is not available when Interface of BKP is enabled (if applicable).</p> <p>If possible, disable a port when it is not used.</p> <p>NOTE: If the attribute is not present, the port is enabled and cannot be disabled.</p>
Baud Rate	<p>Possible values:</p> <ul style="list-style-type: none"> • Auto 10/100 Mbits/s (default) • 10 Mbits/s half duplex • 10 Mbits/s full duplex • 100 Mbits/s half duplex • 100 Mbits/s full duplex • 1000 Mbits/s half duplex • 1000 Mbits/s full duplex <p>The attribute is set to the default value and read-only for controller modules with both DIO and RIO scanner service.</p>
Service L2	<p>You can select None or any combination of services from the following:</p> <ul style="list-style-type: none"> • QoS • RSTP <p>The services that are enabled are shown in the text box when the list is collapsed.</p> <p>NOTE: Some modules support only one service or require that RSTP is enabled.</p>
Port Type	Indicates the type of network that the port supports. For example, a DIO or RIO ring.

NOTE: The values of attributes of both dual ports are cloned.

Control Port Section

The section is available only for BMENUA0100 modules.

Attribute	Description
Enabled	<p>Selecting the check box (TRUE) enables the control port, which becomes the exclusive interface for IPv4 or IPv6 communication, page 679 to the embedded OPC UA server.</p> <p>Clearing the check box (FALSE) disables the control port and the backplane port (BKP) supports IPv4 communication to the OPC UA or Edge server.</p> <p>If possible, disable a port when it is not used.</p> <p>NOTE: In a redundant controller, the value of the attribute is cloned for modules in racks A and B.</p>
Port Type	Indicates the type of network that the port supports. For example, a DIO ring.

Server Memory and Client Memory Categories - CONFIGURATION Tab

Server Memory

The category is displayed for controller modules after you configure the controller by using the **Configure** command.

Attribute	Description
Server Memory Start	<p>Holding register address (%MW) for peer to peer data.</p> <p>(Integer)</p> <p>Default value: Blank</p> <p>You can only enter an odd address, page 778 for this parameter (for example, 1, 5, 11).</p> <p>For more information, refer to Configuring the Controller I/O Scanner Parameters, page 591.</p>
Server Memory Length	<p>Holding register size for peer to peer data.</p> <p>(Integer)</p> <p>Default value: Blank.</p> <p>NOTE: The available memory length is <i>ServerMemoryLength</i> - 1 because one word is reserved to move the channel address for managing the 32-bit alignment, page 778 for peer to peer communication on Modicon M340, M580, and Quantum platforms.</p> <p>For more information, refer to Configuring the Controller I/O Scanner Parameters, page 591.</p>

Client Memory

The category is displayed for communication modules, including controller modules, when the I/O scanner service is enabled after you configure the controller by using the **Configure** command. It is not displayed for controllers of the Modicon M580 platform and BMENOC.... communication modules.

Attribute	Description
Client Memory Start	<p>Holding register address (%MW) for I/O scanning.</p> <p>(Integer)</p> <p>Default value: Blank.</p> <p>For more information, refer to Configuring the Controller I/O Scanner Parameters, page 591.</p>
Client Memory Length	<p>Holding register size in words for I/O scanning</p> <p>(Integer)</p> <p>Default value: Blank.</p> <p>For more information, refer to Configuring the Controller I/O Scanner Parameters, page 591.</p>

Communication Card Category - CONFIGURATION Tab

Description of Attributes

Attribute	Description
Reference	<p>Defines the type of communication card in the entity.</p> <p>Possible values:</p> <ul style="list-style-type: none">• None: No communication card is present or it is present but not configured. The communication card, the corresponding ports, and their attributes are not visible.• ATV Communication Card: Shows the communication card, the corresponding ports, and enables their attributes.• PC Network Interface Card: Shows an additional Network Interface Card (NIC), the corresponding ports, and enables their attributes.

Fieldbus Category - CONFIGURATION Tab

Description of Attributes

The category appears only when you select a secondary device in the **Fieldbus Editor**.

Attribute	Description
Logical Number	<ul style="list-style-type: none">For PROFIBUS devices:<ul style="list-style-type: none">Address of the device on the bus.Range: 0 to 126. 1 is reserved and cannot be used.Read-only.For HART devices:<ul style="list-style-type: none">Channel number of the module.Range: 0 to x-1 (where x is the number of channels that the module is equipped with).

DTM Category - CONFIGURATION Tab

Description of Attributes

Attribute	Description
Device Name	The information is related to the DTM, page 629 that is installed on the local computer and used by the device.
Device Version	
Vendor	
DTM Name	
DTM Version	

Hot Standby Category - CONFIGURATION Tab

Description of Attributes

The category is displayed for redundant M580 controller modules after you configure the controller by using the **Configure** command.

Attribute	Description
ControllerExist	<p>Select a value depending on the physical controller configuration:</p> <ul style="list-style-type: none">• PrimaryAndStandby (default value): When both the primary and the standby physical controllers are present, reachable, and connected by a high-speed cable link.• Primary: If either of the following applies:<ul style="list-style-type: none">◦ Only the primary physical controller is present.◦ Both the primary and the standby physical controllers present but not connected by a high-speed cable link. <p>The value has an impact on the deployment and execution, page 793 stages.</p> <p>Changing the value in one controller module also changes it for the controller module in the counterpart rack</p>

PROPERTIES Pane - SECURITY Tab

Overview

The **SECURITY** tab is available for entities that have attributes related to communication security services. It lets you view and/or configure their attributes, which are grouped in categories.

This topic describes the attributes that can be displayed. Depending on the entity that is selected, some may not appear in the tab.

Global Policy Category - SECURITY Tab

Description of Attributes

Attribute	Description
Security Level	<p>Predefined security level that is applied to the Ethernet services of the module.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • No: Enables Ethernet services and disables access control. This is the lowest security level. • Enforced: Disables Ethernet services and enables access control. • Custom: The value is selected automatically if you change the default value of an attribute in the Services category for the security levels No or Enforced.
Secure Protocol	<p>Communication protocol that is used for deployment.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • No (HTTP): Uses the <i>HTTP</i> protocol. • HTTPS • FTPS • SNMPv3 • IPsec
Pre-Shared Key⁽¹⁾	<p>The attribute cannot be left blank.</p> <p>Enter a 16-character string containing at least one character from each of these categories:</p> <ul style="list-style-type: none"> • Uppercase characters from the classical Latin alphabet (A...Z) • Lowercase characters from the classical Latin alphabet (a...z) • Base-10 digits (0...9) • Special characters (~, !, @, \$, %, ^, &, *, _ , +, -, =, ` , , \, (,), [,] , : , " , ' , < , >)
Enable DH 2048⁽¹⁾	When the check box is selected (TRUE), generates 2048-bit Diffie-Hellman parameters.
Enable Confidentiality⁽¹⁾	When the check box is selected (TRUE), Ethernet services are encrypted.
(1) Available only if Secure Protocol is set to IPsec .	

Services Category - SECURITY Tab

Description of Attributes

Depending on the value of the **Security Level** attribute, page 690, the value of the following attributes may change.

Attribute	Description
FTP	Possible values: <ul style="list-style-type: none"> • Enabled: Enables the service for the module. This setting is required to configure the service for each authorized address of a module in the ACCESS CONTROL VIEW. • Disabled: Makes the service unavailable for the module. In the ACCESS CONTROL VIEW, disables and resets the corresponding service for the authorized entries of a module.
TFTP	
HTTP	
EIP	
SNMP	
DHCP	<ul style="list-style-type: none"> • The service is disabled if Security level is set to Enforced. • The service is enabled if Security level is set to No.
IPsec	<ul style="list-style-type: none"> • The service is enabled if Security level is set to Enforced. • The service is disabled if Security level is set to No.
Access Control	Possible values: <ul style="list-style-type: none"> • Enabled: Lets you configure the enabled Ethernet services for each authorized address of a module in the ACCESS CONTROL VIEW, page 556. • Disabled: Disables the possibility to configure Ethernet services individually for the authorized entries of a module. The existing configuration is retained. Click Edit to open the ACCESS CONTROL VIEW .

Password Protection Category - SECURITY Tab

Description of Attributes

Attribute	Description
Controller	<p>Lets you manage password protection, page 804 for deployment and execution operations performed on the controller.</p> <ul style="list-style-type: none"> • Enabled (default): Enables password protection. • Disabled: Disables password protection after you confirm the selection. You cannot disable the property if the System Access Password property, page 139 of the system is enabled. <p>By default, the value is set according to the value of the Controller Access Password at Creation property, page 139 that is configured at the system level.</p> <p>NOTE: To disable password protection when a password is set, you must enter it. If you already deployed a Control Participant project while the property was set to Disabled, you can enable it and set a password for the already deployed Control project.</p>
Safety	<p>The property is available only for M580 safety controllers and lets you manage the functional safety password protection, page 925.</p> <ul style="list-style-type: none"> • Enabled (default): Makes the use of a password mandatory when performing operations on the safety-related configuration and/or safety-related program of the controller. • Disabled: Disables password protection after you confirm the selection. <p>To disable password protection when a password is set, you must enter it.</p>

Control Expert Category - SECURITY Tab

Description of Attributes

Attribute	Description
Simulator Password Protection	<p>The property is available only after you create a Control Expert service, page 597 on a workstation.</p> <p>It lets you manage password protection, page 804 for deployment and execution operations performed on simulator instances. The setting applies to the Control Expert services of the workstation.</p> <ul style="list-style-type: none">• Enabled (default): Enables password protection.• Disabled: Disables password protection after you confirm the selection. You cannot disable the property when the System Access Password property, page 139 of the system is enabled. <p>By default, the value is set according to the value of the Controller Access Password at Creation property, page 139, which is configured at the system level.</p> <p>NOTE:</p> <p>To disable the property when a password, you must enter it. If several Control Expert services exist, you must enter the password of each one.</p> <p>If you already deployed a Control Participant project while the property was set to Disabled, you can enable it and set a password for the already deployed Control project.</p>

PROPERTIES Pane - SERVICES Tab

Overview

The **SERVICES** tab is available for entities that have attributes related to Ethernet communication services. It lets you view and/or configure their attributes, which are grouped in categories.

This topic describes the attributes that can be displayed. Depending on the entity that is selected, some may not appear in the tab.

DHCP Server and DHCP Client Categories - SERVICES Tab

DHCP Server (Address Server) Category

Attribute	Description
Enabled	When the check box is cleared (FALSE), the DHCP server service of the module is disabled. NOTE: If the attribute is not present, the module provides the DHCP server service to client entities that are configured to subscribe to the service of this module. The service can be disabled in the SECURITY tab.
DHCP Name	Path and name of the interface, page 679 of the entity that provides the DHCP server service. The attribute is available only if Enabled is selected. NOTE: You can see the list of modules in the DHCP VIEW .
DHCP Prefix	Editable prefix that is used to create the DHCP identifier, page 694 of certain modules whose DHCP server is the M580 controller module. The prefix must fulfill the following requirements: <ul style="list-style-type: none"> • Contains three characters or less. • Can contain only letters and numbers. • Cannot consist of the letter C (or c) followed by two digits. Default value: <ul style="list-style-type: none"> • For a standalone M580 controller: <i>Mx8</i> • For a redundant M580 controller: <i>M58</i> NOTE: For M580 safety controllers, if the functional safety password protection, page 925 is enabled, the DHCP prefix value is not applied to the control project. If a value was configured in the control project before the function was enabled, it is retained.

DHCP Client Category

Attribute	Description
Enabled	When the check box is selected (TRUE), the entity can subscribe to a DHCP server service, page 624. NOTE: For certain entities, it is selected by default and read-only (for example, BMECRA**** adapter modules). NOTE: The other attributes of this category are available only if Enabled is selected.
DHCP Server Name	Name of the entity and its interface, page 679 that provide the DHCP server services. Select a value from the list of entities and their interface that exist in the topology and feature a DHCP server service, whether it is enabled or not. NOTE: For certain entities, the attribute value is predefined and read-only (for example, BMECRA**** adapter modules).
Protocol	Protocol that the entity supports. Possible values:

Attribute	Description
	<ul style="list-style-type: none"> • DHCP • BOOTP <p>NOTE: The possible values depend on the type of entity and/or the value of Identified by.</p>
Identified by	<p>Defines how the DHCP server identifies the client:</p> <ul style="list-style-type: none"> • MAC address • Device Name (entity Identifier) <p>NOTE: The possible values depend on the type of entity.</p>
Identifier	<p>Enter the appropriate value depending on the setting that you selected for Identified by. You can only use unique identifiers for a given DHCP server.</p> <p>For certain modules, you can only use values that comply with module-specific format requirements. Refer to the documentation of the module for details.</p> <p>For the following modules, the identifier must be entered by using the format described below:</p> <ul style="list-style-type: none"> • BMEAH10812 and BMEAH00412 • BMECXM0100 • BMENOS0300 • BMENUA0100 • BMEPXM0100 • BMESWT0100 <p><i>[DHCP Prefix][Controller]_[Slot]_[Part Number]</i>, where:</p> <ul style="list-style-type: none"> • <i>[DHCP Prefix]</i> is the prefix, page 694 of the M580 controller. • <i>[Controller]</i> is 0 for an M580 standalone and A or B for a redundant controller. The same naming convention applies to M580 safety controllers. • <i>[Slot]</i> is the slot number of the backplane where the module is installed. • <i>[Part Number]</i> is the part number, page 675 of the module. <p>If an entity is identified by MAC address, the attribute is read-only and the value provided by the MAC Address attribute.</p> <p>The attribute cannot be left blank if Enabled is selected.</p> <p>NOTE: For PMEPXM0100 modules and TeSys devices, the value must not exceed 16 characters.</p> <p>NOTE: For certain entities, the attribute is predefined and read-only.</p> <p>NOTE: For BMECRA••••• adapter modules the device name includes a three-digit number (for example, by default, 001 in <i>CRA_001_0</i> for a BMECRA••••• adapter module located in drop 1).</p> <p>The initial value of the three digits is based on the configuration of their <i>Device Name</i> in the Control Participant when you add the drop. Thereafter, you can only edit the three digits by using the Identifier attribute.</p> <p>Ensure that the setting of the rotary switches on the physical BMECRA••••• adapter module matches these three digits. The first two digits represent the <i>Tens</i> and the third digit the <i>Ones</i> rotary switch.</p> <p>This three-digit number also appears after the name of the drop, page 676 containing the BMECRA••••• adapter module.</p>

FDR Server Category - SERVICES Tab

Description of Attributes

Attribute	Default value	Description
Enabled	TRUE (selected)	When selected, the FDR server service of the module is enabled. The attribute may be read-only for certain modules.

RSTP Category - SERVICES Tab

Operational State

Attribute	Description
Bridge Priority	2-byte value for the switch. Possible values: <ul style="list-style-type: none">• Root(0)• Backup Root(4096)• Participant(32768): Represents the midpoint of the range (from 0 to 65535).

Bridge Parameters

For controller modules, the following attributes are available only for modules without remote I/O (RIO) scanner service.

Attribute	Default value	Description
Forward Delay (ms)	21,000	Read-only.
Maximum Age Time (ms)	40,000	Range: From 6,000 to 40,000
Transmit Hold Count (ms)	40	Range: From 1 to 40
Hello Time (ms)	2,000	Range: From 1,000 to 2,000

NTP Category - SERVICES Tab

Overview

This topic describes the attributes related to the NTP service.

For more information on NTP attributes and their configuration, refer to the documentation of the module.

Configuration

Attribute	Description
Enabled	When the check box is cleared (FALSE, default value), the NTP service for the module is disabled and the other attributes of the NTP category are hidden.
Version	The value depends on the firmware version of the module: <ul style="list-style-type: none"> • SNTP: Uses the Simple Network Time Protocol (SNTP). For firmware version earlier than V4.00. • V4: Uses the NTP version 4 protocol as of firmware version 4.00.
NTP State	Defines the NTP role of the module. Possible values: <ul style="list-style-type: none"> • For Version = SNTP: <ul style="list-style-type: none"> ◦ NTP Client: The network time service (SNTP) synchronizes the clock of the module to that of the time server. ◦ NTP Server: The module can synchronize clocks of NTP clients. • For Version = V4: <ul style="list-style-type: none"> ◦ Client/Server: The module has the role of both client and server. ◦ Server Only: The module can synchronize clocks of NTP clients.
Stratum	Specifies the relative position of the server in the NTP network. The attribute is only available for modules as of firmware version 4.00. Range: From 1 to 15 Default value: 10 NOTE: If NTP State is set to Client/Server , the value is used in orphan mode.
Update CPU Time	When the check box is selected (TRUE), the module acts as an NTP client, which provides the NTP time to the controller module clock. The attribute is available only if NTP State for the module is set to NTP Client .

Server Configuration

This subcategory is available only if the **NTP State** attribute is set to **NTP Client** or **Client/Server**.

For modules with firmware version earlier than V4.00:

Attribute	Description
Primary NTP Server IP Address	Enter the IP address of the server that is used to synchronize the clock of NTP clients.
Secondary NTP Server IP Address	Enter the IP address of the backup server that is used to synchronize the clock of NTP clients.
Polling Period (s)	Time (in seconds) between updates from the NTP server. Range: From 1 to 120 s in 1 s increments. NOTE: For certain modules, the attribute is not available.
Format	Lets you select the IP address format that you can use for each NTP server. Possible values: <ul style="list-style-type: none"> • IPv4 (default) • IPv6 The attribute is available only for BMENUA0100 modules.

NOTE: For a redundant controller, the primary and secondary NTP server attribute values that are configured for a BMENUA0100 module in local rack A are not automatically applied to the module in rack B.

For modules as of firmware version 4.00:

Attribute	Description
NTP Server 1 IP Address to NTP Server 8 IP Address	You can configure the IP address of up to eight servers that can be used to synchronize the clock of NTP clients. Specify the preferred NTP server to be used by the controller module by selecting its check box.
Quality Threshold (ms)	Threshold (in milliseconds) for NTP accuracy. Range: From 1 to 200 ms Default value: 50 ms

Syslog Server Category - SERVICES Tab

Description of Attributes

For more information on the events that are logged, refer to *Logging DTM and Module Events to the SYSLOG Server* in the help of the module.

NOTE: For M580 safety controllers, if the functional safety password protection is enabled, the following attribute values are not applied to the control project. If a value was configured in the control project before the function was enabled, it is retained.

Attribute	Default value	Description
Enabled	FALSE (cleared)	<p>When selected (TRUE):</p> <ul style="list-style-type: none"> Enables event logging for each possible originator of logged events in the architecture. Syslog server attributes are enabled and the configured values are propagated to each event-logging originator in the architecture. <p>The attribute is not available for systems.</p> <p>NOTE: When the attribute is selected, clearing it and selecting it again replaces the values that were configured for IP Address and Port with the default values.</p>
Reset to default	N/a	Replaces the values of IP Address and Port with the default values.
IP Address⁽¹⁾	0.0.0.0	<p>IP address of the Syslog server to which event logs are sent.</p> <p>This address is embedded in the program.</p>
Protocol	TCP	<p>Protocol that is used for communications with the Syslog server.</p> <p>Read-only.</p>
Port⁽¹⁾	601	<p>Port that is used for communications with the Syslog server.</p> <p>Range: From 0 to 65535</p>
<p>(1) The attributes are not editable if Security Editor (see <i>EcoStruxure™ Control Expert, Security Editor, Operation Guide</i>) is started in Secured mode on the system server computer and security is on.</p>		

SNMP Category - SERVICES Tab

Description of Attributes

Attribute	Description
Enabled	<p>When selected, enables SNMP for the module and lets you configure attributes of the SNMP category.</p> <p>For certain entities, the check box is selected and read-only.</p> <p>Your configuration is retained if you select/clear the check box.</p> <p>NOTE: For M580 controller modules, the check box is read-only and its state is inherited from the SNMP security service, page 690.</p>
Version	<p>Version of SNMP that is used for communication.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • V1 • V3. For controller modules, the value is available as of firmware version 4.00. <p>The possible values depend on the type and version of the entity.</p> <p>Your configuration is retained if you switch SNMP versions.</p>

IP Address Managers

Attribute	Description
IP Address Manager 1	IP addresses of the first and second SNMP managers to which the agent that runs in the device sends trap notifications.
IP Address Manager 2	

Authentication

This subcategory is available only for SNMPv3.

Attribute	Description
User Name	<p>Required.</p> <p>To be valid, the user name must fulfill the following requirements:</p> <ul style="list-style-type: none"> • Contains 32 characters or less. • Contains only any combination of ASCII characters from 33 to 122. • Spaces are not allowed.

Agent

Attribute	Description
Enable SNMP Manager	When selected (TRUE), the Location (SysLocation) and Contact (SysContact) attributes are hidden and managed by the SNMP manager.
Location (SysLocation)	Physical location of the device. Enter a value by using characters of the US-ASCII table (32 characters maximum).
Contact (SysContact)	Contact information of the person in charge of device maintenance. Enter a value by using characters of the US-ASCII table (32 characters maximum).

Community Names

This subcategory is available only for SNMPv1.

Attribute	Default value	Description
Set	<i>private</i>	Password that the agent requires before it executes write commands received from the SNMP manager.
Get	<i>public</i>	Password that the agent requires before it executes read commands received from the SNMP manager.
Trap	<i>alert</i>	Password that the SNMP manager requires before it accepts trap notifications from an agent.

Security

This subcategory is available only for SNMPv1.

Attribute	Description
Enable “Authentication Failure” Trap	When selected (TRUE), the agent sends a trap notification to the SNMP manager in case it receives a <i>Get</i> or <i>Set</i> request from a device that it cannot authenticate.

QoS Category - SERVICES Tab

Overview

This topic describes the attributes related to the QoS service.

For controller modules, the following attributes are available only for modules without remote I/O (RIO) scanner service.

Description of Attributes

Attribute	Description
Enabled	When selected (TRUE), the module adds DSCP tags to the IP header of each Ethernet packet that it transmits to indicate its priority. It also enables the attributes of the other subcategories of QoS .

PTP

Attribute	Default value	Description
DSCP PTP Event Priority	59	Range: From 0 to 63
DSCP PTP General Priority	47	Range: From 0 to 63

EtherNet/IP Traffic

Attribute	Default value	Description
I/O Data Scheduled Priority Messages	47	DSCP value for I/O data scheduled priority messages that use the EtherNet/IP protocol. Range: From 0 to 63
Explicit Messages	27	DSCP value for explicit messages that use the EtherNet/IP protocol. Range: From 0 to 63
I/O Data Urgent Priority Messages	55	DSCP value for I/O data urgent priority messages that use the EtherNet/IP protocol. Range: From 0 to 63
I/O Data High Priority Messages	43	DSCP value for I/O data high priority messages that use the EtherNet/IP protocol. Range: From 0 to 63
I/O Data Low Priority Messages	31	DSCP value for I/O data low priority messages that use the EtherNet/IP protocol. Range: From 0 to 63

Modbus TCP Traffic

Attribute	Default value	Description
I/O Messages	43	DSCP value for I/O messages that use the Modbus TCP protocol. Range: From 0 to 63
Explicit Messages	27	DSCP value for explicit messages that use the Modbus TCP protocol. Range: From 0 to 63

Network Time Protocol Traffic

Attribute	Default value	Description
Network Time Protocol Messages	59	DSCP value for network time protocol (NTP) messages. Range: From 0 to 63

Advanced Category - SERVICES Tab

Overview

For controller modules, the following attributes are available only for modules without remote I/O (RIO) scanner service.

EtherNet/IP Timeout Settings

Attribute	Default value	Description
FW_Open I/O Connection Timeout (ms)	4960	Specifies the time (in milliseconds) the scanner waits for FW_Open response of an I/O connection.
FW_Open EM Connection Timeout (ms)	3000	Specifies the time (in milliseconds) the scanner waits for FW_Open response of an EM connection.
EM Connection RPI (ms)	10,000	Sets T->O and O->T RPI for all EM connections.
EM Request Timeout (ms)	10	Specifies the time (in milliseconds) the scanner waits between the request and the response of an explicit message.

EtherNet/IP Scanner Behavior

Attribute	Default value	Description
Allow RESET via explicit message	FALSE (cleared)	When selected (TRUE), the scanner resets if an identity object reset service request is received; otherwise, it ignores the request.
Behavior when CPU state is STOP	Idle	Possible values: <ul style="list-style-type: none">• Idle: The EtherNet/IP I/O connection stays open, but the <i>Run/Idle</i> flag is set to <i>Idle</i>.• Stop: The EtherNet/IP I/O connection is closed.

Email Category - SERVICES Tab

Configuration

Attribute	Description
Enabled	When selected (TRUE), the module sends an email message to the defined mail server, which is triggered by the DATA_EXCH function block. It also enables the attributes of the other subcategories of Email .

SMTP Server Configuration

Attribute	Description
SMTP Server Address	Enter the IP address of the SMTP mail server.
SMTP Server Port	TCP port number for SMTP transmissions. The value must match the port of the SMTP mail server you are using.

Authentication

Attribute	Description
Enabled	When selected (TRUE), the Login and Password values are used by the SMTP server to authenticate the client.
Login	Your ID must fulfill the following requirements: <ul style="list-style-type: none"> Only printable character is allowed. It contains 64 characters or less.
Password	Your password must fulfill the following requirements: <ul style="list-style-type: none"> Only printable character is allowed. It contains 64 characters or less.

Email Header 1–Email Header 3

Each email message contains a user-defined header. You can configure up to three headers. Each header contains data from the following attributes.

Attribute	Description
From	Email address of the sender.
To	Email addresses of the recipients must fulfill the following requirements: <ul style="list-style-type: none"> Separate each email address by a comma or semi-colon. Contains 128 characters or less.
Subject	Fixed part of the subject. Contains 32 characters or less.

OPC UA Category - SERVICES Tab

Description of Attributes

The attribute is available only for BMENUA0100 modules.

For a redundant controller, values that you configure for the module in local rack A are automatically applied to the module in local rack B.

Attribute	Default value	Description
OPCUA TCP Listening Port	4840	Number of the port that is used by the OPC UA server. Range (if different from default port): From 49152 to 65535
Sampling Rate	Default	Possible values: <ul style="list-style-type: none">• Default• 20 ms

Workstation Services Category - SERVICES Tab

Common Attributes

Attribute	Default value	Description
Identifier	–	Editable identifier of the service instance.
Template Identifier	–	Identifier of the template that is used by the service instance. Read-only.
Template Version	–	Version of the template that is used by the service instance. Read-only.

Control Services Attributes

Attribute	Default value	Description
Port	502	<p>Port number of the Controller simulator.</p> <p>Ensure that the Controller simulator, page 800 where you deploy the Control project uses the same port number.</p> <p>Changing the value removes service and communication mappings that exist for the Control Expert service.</p> <p>You cannot change the value if you deployed to this instance of the simulator.</p>

Supervision Attributes

Attribute	Default value	Description
TCP Listening Port	49152	<p>Configure the value to be the same as TCP Listening Port of the OPCUA server that is installed on the workstation.</p> <p>The value is used in the I/O device address, page 788 only if Protocol of the associated tag container is set to OPCUA.</p> <p>Range: From 49152 to 65535.</p> <p>The parameter is available only for workstations where an OFS service was created.</p>

PROPERTIES Pane - I/O PROFILE Tab

Overview

The I/O PROFILE tab is available for entities whose inputs/outputs and/or diagnostic data is managed by using Device Derived Data Types (DDT). It lets you view and configure some of the Device DDT attributes, which are grouped in categories.

This topic describes the attributes that can be displayed. Depending on the entity that is selected, some may not appear in the tab.

Ethernet Modbus TCP and Ethernet EtherNet/IP Categories - I/O PROFILE Tab

Overview

For devices of the **Automation Devices** product type (except devices of the **EPE Managed Devices** product range) that are configured by using a Device Type Manager (DTM), data becomes available only after you opened, [page 629](#) and closed the DTM once by using the device context menu. Generally, for Modbus TCP and EtherNet/IP devices, a request/connection is created by using the default data of the DTM. For certain devices (for example, TeSys T motor controllers), a request is created only after you selected a device reference and saved the change while closing the DTM.

For details about the relation between DTM and I/O profile data, refer to the topic describing how to [configure I/O profiles](#), [page 638](#).

NOTE: For redundant M580 controllers, the **I/O PROFILE** tab is available only for modules of rack A.

Description of Attributes

Attribute	Description
Service	Name and category of the I/O scanner service of the device Read-only.
Server	Name of the server that is used to communicate by using the respective protocol. Read-only.
DTM Reference⁽¹⁾	Part number and version of the module. Click Select to select from the list the DTM version that corresponds to the version of the module that you are using. Once a DTM is selected, you can create and configure local slaves. The box is empty by default. NOTE: Selecting a different value while local slaves exist deletes these local slaves.
(1) The attribute appears only for M580 controllers and BMENOC03•1 modules.	

Local Slaves Subcategory

The **Local Slaves** subcategory is created once you have selected a DTM. It shows the local slaves that exist for the module.

Clicking **Plus Sign (+)** creates an empty **Local Slave 1** and opens the selected DTM to let you configure the local slave.

Closing the DTM renames the local slave, adds its connection type according to the configuration that you made in the DTM, and creates the **Profile [1]** subcategory. Then, you can configure the profile, the connections in its memory registers, and the item mapping if needed. Refer to the tables further in this topic for details.

Each local slave has only one I/O profile that you can add to a scanner, page 641.

Clicking **Edit** reopens the DTM, which lets you perform changes (for example, removing the connection and replacing it by another local slave connection).

NOTE: Ensure that for a given controller, you do not create conflicting peer to peer communication configurations by using at the same time local slaves, peer to peer communication templates, page 463 (communication mapping, page 725), and/or specialized peer to peer communication function blocks, page 475.

NOTE: The number of local slaves that you can create depends on the module.

Communication Profiles Subcategory

A communication profile is the definition of the data exchanged by a device. The profile can contain one or more requests or connections. Creating several profiles for a device lets you, for example, preconfigure requests for various scanners by adjusting certain attributes to each one. The profile that is used for communication, page 638 between the device and the scanner is the *active* profile.

When a device has an active communication profile, it is added to the device list of the master DTM of the scanner, which creates a device DDT type and variable in the controller configuration project. The structure of the device DDT takes into account the I/O mapping, page 642.

For each device (except devices of the **EPE Managed Devices** product range), you can create up to six profiles. The minimum is one.

Each profile features the following attributes.

Attribute	Default value	Description
Profile Name	The name of the device when it is created followed by <i>ProfileX_Device_Status</i> .	<p>Name of the I/O profile, which is used as default value for the active I/O profile, page 641.</p> <p>Optional.</p> <p>NOTE: For certain modules, the default value is the device name (DHCP identifier, page 694) and is read-only (for example, BMECRA**** adapter modules).</p>
Instance Name	The name of the device when it is created followed by <i>ProfileX</i> .	<p>The name is used as default value for the active I/O profile, page 641.</p> <p>The value of this attribute in the active I/O profile becomes the name of the device DDT variable that is used in the controller configuration project to manage the device data.</p> <p>If the attribute is editable, the value cannot be null and must fulfill the following requirements:</p> <ul style="list-style-type: none"> • Contains 30 characters or less. • Can contain only alphanumeric characters of the US-ASCII character table and underscore (_). • Must start with a letter and cannot end with underscore. • Cannot contain two consecutive underscores. • Cannot contain spaces. <p>You can only use unique names for a given controller configuration project.</p> <p>NOTE: For certain modules, the default value is the name of the associated device DDT variable and is in read-only (for example, BMECRA**** adapter modules).</p>
Type Name	The value of Instance Name preceded by <i>T_</i> .	<p>The name is used as default value for the active I/O profile, page 641. It becomes the name of the device DDT type that is used in the controller configuration project to manage the device data.</p> <p>Read-only.</p>
Device Number	First available number in the range that corresponds to the scanner.	<p>Indicates the relative position of the device in the device list of the master DTM of the scanner.</p> <p>The list contains the numbers that can be assigned to the device when it is scanned by a given scanner. This includes numbers that may already be assigned to other devices, which are scanned by the same scanner independently of the communication protocol used by the device.</p> <p>The attribute is shown only if you select the I/O profile in a Communication Mapping View (active I/O profile, page 641).</p> <p>NOTE: For some devices, the attribute is read-only.</p> <p>NOTE: For local slaves, the attribute is not shown. A device number is assigned to each local slave by the master DTM of the scanner.</p>
Mapping	—	<p>Mapping mode that is used to perform and manage the I/O mapping for the profile.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Manual: <ul style="list-style-type: none"> ◦ Data in the I/O Mapping Editor is editable. ◦ Some I/O profile attributes are editable. • Automatic: <ul style="list-style-type: none"> ◦ I/O mapping and request/connection data of the DTM of the device is used and overwrites changes that you have made. ◦ Data in the I/O Mapping Editor is read-only. ◦ Some I/O profile attributes are read-only. ◦ Adding and removing requests or connections is disabled. ◦ Requests/connections added manually are deleted. <p>Click Edit to open the I/O Mapping Editor for the device.</p> <p>NOTE: For a BMECRA**** adapter module, automatic mode is selected by default and is read-only.</p>

For each active I/O profile, the following additional attributes are available when you select the profile in a Communication Mapping View.

Attribute	Description
Path	Name of the scanning device. Read-only.
Comment	You can enter a comment for the active I/O profile by using free-form text.

NOTE: If the value of **Instance Name**, **Type Name**, or **Profile** already exists in the system topology, the _x suffix is added, where x is an integer.

Memory Registers Section for Ethernet Modbus TCP

For each profile, you can create up to 64 requests. The minimum is one.

Subsec- tion	Attribute	Default value ⁽¹⁾	Description
Request [x]	Default UnitID ⁽²⁾	255	Number that is used to identify the target of the connection. Typically, a device behind a gateway. Range: From 0 to 255 NOTE: For devices of the EPE Managed Devices product range, the parameter is provided by the <i>DeviceID</i> element of the Modbus TCP server interface (<i>MBTCPServiceServer</i>), which is referenced by Schneider Electric Modbus TCP I/O device hardware templates. You can also configure the parameter in the Communication Mapping tab, page 771 of Supervision executables.
	Default Health Timeout (ms) ⁽²⁾	1500	Maximum allowed interval between device responses before a timeout is detected. Range: From 5 to 65535 in 5 ms intervals.
	Default Repetitive Rate (ms) ⁽²⁾	60	Data scan rate. Range: From 5 to 60000 in 5 ms intervals.
	Read Address ⁽³⁾	0	Range: From 0 to 65535
	Read Length ⁽³⁾	1	Number of words that are allocated to read from the server when you create a communication channel. The corresponding memory size is reserved in both the client and the server memory heap. Range: From 0 to 125
	Default Last Value ⁽²⁾	Hold Value	Input value that is retained if a communication interruption occurs. Possible values: <ul style="list-style-type: none"> • Hold Value • Set to Zero
Request [x]	Write Address ⁽³⁾	0	Range: From 0 to 65535
	Write Length ⁽³⁾	0	Number of words that are allocated to write to the server when you create a communication channel. The corresponding memory size is reserved in both the client and server memory heap. Range: From 0 to 120
	Default Gateway/Bridge Device	FALSE	When selected (TRUE), lets slower TCP/IP devices communicate with the I/O scanner of BMENOC0301/11.4 communication modules. The modules double the timeout setting by increasing the number of retransmissions to 6 instead of the typical 3.
<p>(1) Values for devices without DTM and for additional requests created in manual mapping mode. Otherwise, default values of the first request are those of the device DTM.</p> <p>(2) The value of the attribute can be modified independently in the active I/O profile, page 641 where the name appears without the term <i>Default</i>. Switching to automatic mapping mode resets the value of the attribute in the active and device I/O profiles.</p> <p>(3) For devices that are configured by using a DTM, in the first request, the value is read-only and provided by the DTM. In requests added manually, you can change the value, page 642 if the mapping mode is set to manual.</p>			

Memory Registers Section for EtherNet/IP

For each profile, you can create up to 2 connections.

For certain devices, the following may apply:

- Additional connections can only be created in the DTM.
- Connection data is managed in the Control Participant and does not appear in the **I/O PROFILE** tab (for example, for BMECRA•••• adapter and BMECXM0100 modules).

Subsec- tion	Attribute	Default value ⁽¹⁾	Description
Conne- ction [x]	Default RPI T->O ⁽²⁾	30	Request Packet Interval. The data refresh period (ms) of the input connection. Range: From 2 to 65535
	Default Timeout Multiplier	—	This value multiplied by the RPI produces a value that triggers an inactivity timeout. Use the indicated value (see <i>Modicon M580, Hardware, Reference Manual</i>) depending on the RPI value. Range: From 4 (default) to 512 in multiples of 4. NOTE: You can change the value of the attribute independently in the active I/O profile, page 641 where the name appears without the <i>Default</i> prefix. NOTE: Switching to automatic mapping mode resets the value of the attribute in the active and device I/O profiles.
	Default Input Fallback Mode	Set to Zero	Value of inputs if the communication is interrupted. Read-only.
	Input Size ⁽²⁾⁽³⁾	1	Number of words configured in the device. Range: From 1 to 509
	Input Mode ⁽²⁾	Multicast	Transmission type.
	Input Type ⁽²⁾	Fixed	Ethernet packet type (fixed or variable length) to be transmitted. Possible value: Fixed
	Input Priority ⁽²⁾	Sched- uled	The transmission priority depends on the DTM. Possible value: Scheduled
Conne- ction [x]	Input Trigger ⁽²⁾	Cyclic	Transmission trigger. Possible values: <ul style="list-style-type: none"> • Cyclic • Change of State • Application
	Default RPI O->T ⁽²⁾	30	Request Packet Interval. The data refresh period (ms) of the output connection. Range: From 2 to 65535
	Output Size ⁽²⁾⁽³⁾	1	Number of words configured in the device. Range: From 1 to 505
	Output Mode ⁽²⁾	Point to Point	Transmission type. Possible values: <ul style="list-style-type: none"> • Multicast • Point to Point
	Output Type ⁽²⁾	Fixed	Ethernet packet type (fixed or variable length) to be transmitted. Possible value: Fixed
	Output Priority ⁽²⁾	Sched- uled	The transmission priority depends on the DTM. Possible value: Scheduled
<p>(1) Values for devices without DTM and for additional connections created in manual mapping mode. Otherwise, default values of the first connection are those of the device DTM.</p> <p>(2) For devices that are configured by using a DTM, the value is read-only and provided by the DTM.</p> <p>(3) In connections added manually, you can change the value, page 642 of the attribute from the I/O Mapping Editor.</p>			

IOS Type Category - I/O PROFILE Tab

Description of Attributes

The parameter is available only for certain Modbus TCP devices of the **EPE Managed Devices** product range (for example, those using the *\$ETesysTHW* template).

Attribute	Description
IOS Type	<p>Defines the I/O scanning element that is used.</p> <p>Possible values:</p> <ul style="list-style-type: none">• <i>TesysTE</i>: To read/write the entire data from/to the device.• <i>TesysTEFast</i> (default value): To read/write a minimum set of data from/to the device. <p>The parameter value and the device template that is used by application instances must match; otherwise, you cannot perform the <i>hardware mapping</i>, page 747.</p> <p>For example, if you select <i>TesysTEFast</i>, instantiate the <i>\$TesysTEFast</i> template in the application.</p> <p>NOTE: Changing the parameter value after the hardware mapping is done removes the mapping.</p>

PROPERTIES Pane - DIAGNOSTIC Tab

Overview

The various attributes are grouped in categories.

This topic describes the attributes that can be displayed. Depending on the entity that is selected, some may not appear in the tab.

System Diagnostics Category - DIAGNOSTIC Tab

Overview

The topic describes the attributes that you can configure to obtain diagnostic data when system monitoring, page 1107 is enabled.

The DIAGNOSTIC tab is available for devices of the **Automation Devices** product type, **Generic Devices** product range that communicate by using the Modbus TCP protocol and that are capable of providing diagnostic data based on requests that are sent to the device. It lets you view and configure the services that you want to monitor.

Description of Attributes

Attribute	Default value	Description
Presence	TRUE (selected)	When selected, enables sending requests to the device in order to display a diagnostic icon, page 1107, which indicates the communication status of the device. NOTE: The <i>Service not available</i> (gray dot) icon is displayed by default if Presence is cleared.
Protocol	<i>Ping</i>	Protocol that is used to send requests to the device to obtain its communication status.
Request Code	Blank	The request that is sent to the device to obtain diagnostic data including the data address to be read. The request is protocol and device specific. Format: Byte array using decimals, comma separated, without spaces, in brackets. Example: For a Modbus request to read the content of analog output holding registers 40108 to 40109 from the device with address 17 which is <i>11 03 006B 0002 7687</i> Where: <ul style="list-style-type: none"> • <i>11</i> is the hexadecimal slave address. • <i>03</i> is Function Code 3. • <i>006B</i> is the address of the first register to read (decimal 107 with 4001 offset). • <i>0002</i> is the number of registers to read. • <i>7687</i> is the Cyclic Redundancy Check (CRC). Enter the following: [17,03,00,107,00,03,118,135]

Operating State Subcategories

Each subcategory corresponds to an operating state that can display the corresponding diagnostic icon, page 1116. More than one icon can be displayed at the same time for a device.

To configure these attributes, **Presence** must be enabled.

The following table describes the set of attributes that are available for each operating state.

Attribute	Default value	Description												
Enabled	FALSE (cleared)	When selected, enables sending requests (Request Code) to the device in order to display the corresponding operating state icon based on the result of Request Expression .												
Protocol	Modbus TCP	Protocol that is used to send requests to the device to obtain its operating status. Read-only.												
Request Expression	Blank	<p>Logical expression that is applied to the reply received from the device for Request Code. If TRUE, the corresponding operating state icon is displayed.</p> <p>The expression is device-specific.</p> <p>Example:</p> <p>If the reply to the above request code example is</p> <p><i>11 03 06 AE41 5652 4340 49AD.</i></p> <p>you can enter:</p> <ul style="list-style-type: none">[4] =174 to verify the high value of register 40108 (hexadecimal AE is 174).[4] . 0 to verify bit 0 of byte 4. <p>Syntax example:</p> <p>LEN>3 AND ([1]=0 OR [1]=2)</p> <p>You can use operators AND, OR, and NOT and the following functions.</p> <table><tr><th>Function</th><th>Description</th></tr><tr><td>[ByteIndex].BitIndex</td><td>Specifies the bit in the byte at <i>ByteIndex</i>, which is the offset of the byte in the reply.</td></tr><tr><td>[ByteIndex]=<i>value</i></td><td>Verifies if the value of the byte is equal to <i>value</i>.</td></tr><tr><td>[ByteIndex] !=<i>value</i></td><td>Verifies if the value of the byte is different from <i>value</i>.</td></tr><tr><td>LEN><i>value</i></td><td>Verifies if the length of the reply is greater than <i>value</i>.</td></tr><tr><td>LEN<=<i>value</i></td><td>Verifies if the length of the reply is smaller than <i>value</i>.</td></tr></table>	Function	Description	[ByteIndex].BitIndex	Specifies the bit in the byte at <i>ByteIndex</i> , which is the offset of the byte in the reply.	[ByteIndex]= <i>value</i>	Verifies if the value of the byte is equal to <i>value</i> .	[ByteIndex] != <i>value</i>	Verifies if the value of the byte is different from <i>value</i> .	LEN> <i>value</i>	Verifies if the length of the reply is greater than <i>value</i> .	LEN<= <i>value</i>	Verifies if the length of the reply is smaller than <i>value</i> .
Function	Description													
[ByteIndex].BitIndex	Specifies the bit in the byte at <i>ByteIndex</i> , which is the offset of the byte in the reply.													
[ByteIndex]= <i>value</i>	Verifies if the value of the byte is equal to <i>value</i> .													
[ByteIndex] != <i>value</i>	Verifies if the value of the byte is different from <i>value</i> .													
LEN> <i>value</i>	Verifies if the length of the reply is greater than <i>value</i> .													
LEN<= <i>value</i>	Verifies if the length of the reply is smaller than <i>value</i> .													

Mapping Stage

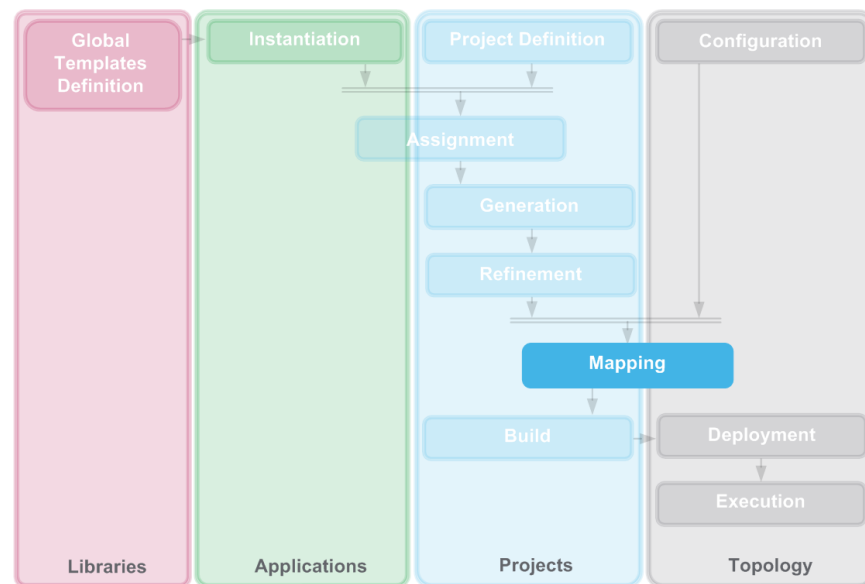
What's in This Chapter

Control Project Mapping Stage	720
Exporting and Importing Hardware Mappings	754
Supervision Project Mapping Stage	763

Overview

This chapter describes how to map the Control and Supervision projects to the topological entities of the system, and to create communication channels, which allow you to use the I/O scanner function of controllers.

The following figure shows the position of the **Mapping** stage within the system engineering life cycle.



Refer to the Mapping stage, page 63 for a description of the purpose of this stage.

Control Project Mapping Stage

Overview

This section describes how to use the **Project Explorer** to map a logical Control Participant project to one or more topological entities of the system to map the application to hardware channels. It also describes how to create and configure communication channels that allow using the controller I/O scanner function for communication with devices and peer to peer communication.

You must have completed the [Generation stage, page 396](#) and the [Configuration stage, page 512](#) of the Control Participant project to proceed with the mapping stage.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Creating Control Project Executables

Overview

A Control executable allows you to associate to the logical Control Participant project a topological entity acting as **engine**, [page 562](#) by mapping the **Service**. Based on that, you can further map **Hardware** (I/O and communication modules) and **Communication** (communication channels for various communication modules) of this engine.

You can map a logical Control Participant project to several engines of the system. To do so, create one executable for each engine.

For example, you can map the same logical Control Participant project to both:

- The controller that is in the plant by using executable_A.
- Another controller that is in the laboratory for test purposes by using executable_B.

Executables are associated to an **Execution Domain**, which serves as a filter for selecting applicable services, for example, to define the boundaries for peer to peer communication or runtime navigation services.

Creating Control Executables

To create a Control executable, in the **Control Project Browser**, right-click the **Executables** node of the Control project that you want to map and select **Create Executable**.

Result: A Control executable is created under the **Executables** node.

Control Executable Actions

The table describes the executable context menu commands.

Command	Description
Manage	<p>Opens the Manage window in the work area of the Project Explorer, which lets you perform the following actions:</p> <ul style="list-style-type: none"> • Service mapping, page 723: Associates the logical Control Participant project to an engine of the system. • View and edit the properties, page 722 of the executable. • Communication mapping, page 725: Defines the Modbus TCP Ethernet (I/O scanning) communication channels between a controller and devices and/or other controllers for peer to peer communication. • Hardware mapping, page 747: Lets you map the interfaces of the instances of the application that are assigned to the project to hardware channels of the controller.
Build , page 774	<p>Starts an automated process to integrate certain changes in the built Control Participant project by performing an incremental build.</p> <p>The command is available once you have done the service mapping.</p>
Build All	<p>Starts an automated process to create, page 775 the built Control Participant project the first time or to recreate it to integrate certain changes, page 781 that you have made to the Control Participant project or the system topology.</p> <p>The command is available once you have done the service mapping.</p>
Generate and Build , page 399	<p>Starts a generation of the Control Participant project followed by a build.</p> <p>If you have not built the logical Control Participant project yet, the build process that is executed is the same as when you select the Build All command.</p> <p>If the generation process does not complete successfully, no build is performed and information is displayed in the notification panel.</p> <p>NOTE: After selecting the command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task.</p>

Command	Description
Open Built Project , page 780	Opens the built Control project in the Control Participant, which allows you to view the program and configuration. The command is available once you have built the Control Participant project.
Delete	Deletes the executable including any associated: <ul style="list-style-type: none"> • Service and hardware mappings. • Communication mappings. <p>NOTE:</p> <p>Deleting an executable does not delete data backup files, page 296.</p>
Rename	Lets you enter a new identifier, page 722 for the executable.

Control Executable Properties

To open the properties of the executable, double-click it.

You can view and edit the following properties.

Property	Description
Identifier	The identifier must be unique within a Control Participant project.
Description	Optional. You can enter a description for the executable with free form text.
Execution Domain	Optional. String. Default value: Blank. For the following types of communication, you can exchange data only if these executables have the same execution domain: <ul style="list-style-type: none"> • For peer to peer communication: The executable of the owner project and the executable of the consumer project. • For runtime navigation services (see EcoStruxure™ Process Expert, Runtime Navigation Services , User Guide): The executable of the Control Participant project and the executable of the Supervision Participant project. <p>Values that you enter remain in the menu and you can select those of executables of other Control and Supervision Participant projects.</p> <p>You can assign a logical execution domain to the executable with free form text.</p> <p>You can modify the Execution Domain value at any time during the engineering life cycle.</p> <p>NOTE: Leaving Execution Domain blank is considered as a value.</p>
Build State	Indicates the status, page 779 of the executable. Default value: Not Built
Built On	Date and time when you last built the executable successfully by using a build command.

Assigning an Execution Domain

To assign an execution domain to a Control executable, proceed as follows.

Step	Action
1	Double-click the executable to open its properties.
2	Select a value from the Execution Domain menu or enter a new value.

NOTE: To remove or change the associated execution domain, select the empty row at the top of the menu.

Mapping Services

Overview

The **Service Mapping Editor** opens in the **Manage** window when you double-click the Control executable, page 721. It allows you to associate the logical Control Participant project to topological entities, page 562 to which you can deploy the Control executable.

These entities are controllers and Control services of workstations that exist in the topology of the system.

NOTE: If the **Controller Family** is not the same for the entity and the Control project, page 270 that you want to map, you may not be able to map certain hardware I/O interfaces.

Mapping a Controller

A controller is displayed in the **Service Mapping Editor** only after you opened it once in the Control Participant by using the **Configure** command.

Mapping a Redundant Controller

For redundant controllers, only the name of the controller node is displayed in the **Service Mapping Editor**. The mapping of the Control executable is done with the primary and the secondary controllers.

Mapping a Workstation

A workstation is displayed in the **Service Mapping Editor** only if you create at least one Control Expert service, page 598 in the workstation.

If you created several Control Expert services, their identifiers are shown in brackets after the workstation name.

Mapping Other Topological Entities

EcoStruxure Process Expert does not display topological entities other than controllers and workstations in the **Service Mapping Editor**. For entities of the **EPE Managed Devices** product range, you must map to the Control executable the controller that communicates with the entity.

NOTE: For PRMs, ensure that the controller family that is selected in the properties of the PRM, page 676 is the same as the family of the controller.

Changing the Service Mapping of Deployed Executables

If you already deployed the Control executable to an entity, you cannot change nor remove the service mapping of this executable. You need to create a new executable.

Mapping Services

To map the Control executable to an entity, proceed as follows.

Step	Action
1	Double-click the executable of the Control project that you want to map and select Manage . Result: The Service Mapping Editor opens.
2	Click the Engine menu. Result: The menu displays the names of compatible entities of the topology of the system.
3	If you have created peer to peer communication mappings, page 475, the Data Server for Peer to Peer Communication menu is also shown.
4	Select an entity and/or the communication module that you want to use for peer to peer communication. Result: <ul style="list-style-type: none">• The Control executable is mapped to the selected entity.• The Hardware Mapping tab appears at the bottom of the window. If the M580 controller is connected to a logical network (or has a physical connection, page 615 for other platforms), the Communication Mapping tab is available.

NOTE: To remove a mapping, in the **Service Mapping Editor**, right-click an entity and select **Unmap**.

Mapping Communication

Overview

To open the **Communication Mapping Editor**, click the **Communication Mapping** tab within the **Manage** window, page 721. The tab appears once you have performed the service mapping, page 723 with a controller that is connected, page 576 (M580) or linked, page 615 (M340 and Quantum) to a logical network.

It allows you to define the communication channels that are used for communication between the following entities and the controller that you configured as client and mapped to the Control executable:

- Modbus TCP devices.
- STB islands.
- PRMs.
- Other controllers that are mapped to other logical Control Participant projects for peer to peer communication.

This method uses the I/O scanner service of the controller that is configured as client.

A communication channel is the logical representation, at the platform level, of the Control Participant I/O scanner row.

When you create a communication channel, page 729, you can configure its properties.

For peer to peer communication, the **Communication Mapping Editor** also allows you to map network variables, page 738 to communication channels.

EcoStruxure Process Expert creates the I/O scanner lines that correspond to the communication channels in the controller acting as client during the build process, page 775.

NOTE: For peer to peer communication, proceed with the communication mapping from the executable of the consumer project, page 463.

NOTE: The communication mapping feature is not supported for executables that you have mapped to a workstation that is emulating a simulator.

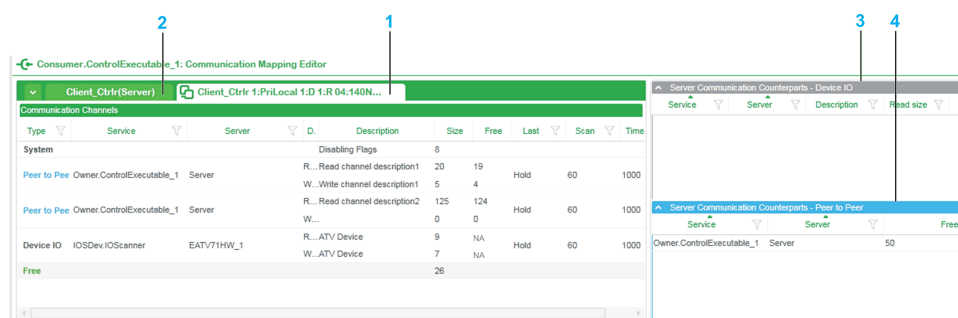
Prerequisites

For a description of the prerequisites for communication mapping, refer to the topics describing:

- Communication with devices, page 936.
- Peer to peer communication, page 943.

Communication Mapping Editor

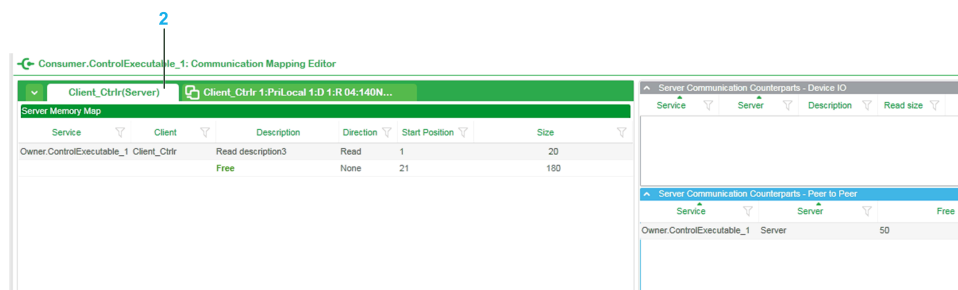
The following figure shows an example of the **Communication Mapping Editor** window, which displays two peer to peer communication and one Modbus I/O device communication channels. It also indicates that free memory remains in the server counterpart to create additional peer to peer communication channels and/or increase the size of existing channels.



Item	Description
1	Communication Channels pane, which displays the client memory area of the communication module of the controller that you have mapped to the Control executable. There is one tab for each communication module (including the CPU communication module) that is configured in the controller and whose I/O scanning service is enabled. Select a tab to display the communication channels that are configured in the memory area of the corresponding communication module of the controller acting as client.
2	<controller_name>(Server) tab, which displays the communication channels that are configured in the memory area of the controller when acting as server (if applicable). The information is displayed in the Server Memory Map pane.
3	Server Communication Counterparts - Device IO pane, which displays, topological entities such as Modbus TCP devices, PRMs, and STB islands, acting as servers, and with which you can define a communication channel. Click the arrow in the title bar to expand/collapse the pane
4	Server Communication Counterparts - Peer to Peer pane, which displays other Control projects, their executable, and the mapped controller entity with which you can define a communication channel. Click the arrow button in the title bar to expand/collapse the pane.

NOTE: If the executable of the Control project is mapped to a redundant controller, the **Communication Channels** pane displays only the client memory area of the communication module of the primary controller.

The figure shows an example of the **Server Memory Map** pane, which is displayed when you select the <controller_name>(**Server**) tab. It indicates that one peer to peer communication channel exists in the memory area of the client controller when acting as server, page 736.



The table describes the fields of the **Communication Channels** pane when you select the tab of the client communication module.

Column header	Description
Type	System Indicates in the Size column, the size of memory that is reserved, and in the Description column, the corresponding usage.
	Free Refer to the description of the Size column.
	Device IO Identifies channels that exist to communicate with Modbus TCP Ethernet devices, PRMs, and STB islands.
	Peer to Peer Identifies channels that exist to exchange data between Control Participant projects through the controllers that are mapped to these projects.
Service	<ul style="list-style-type: none"> For I/O device communication channels: <i>element category</i>. <i>communication element</i> Where <i>element category</i> is the category of the I/O scanner element, and <i>communication element</i> the name of the I/O scanner element for the I/O device. For peer to peer communication: <i>project.executable</i> Where <i>project</i> is the identifier of the Control Participant project (owner, page 463) that exposes the variables and <i>executable</i> the identifier of the executable of this project.
Server	Name of the server counterpart using the communication channel: <ul style="list-style-type: none"> The I/O device name for I/O device channels. The controller name for peer to peer communication channels.
Direction	Indicates the direction of the communication as Read , Write , or Both .
Description	Displays the value of the parameters Read description and Write description that you can view in the Channel Properties dialog box, page 733. NOTE: For channels of type System indicates the usage of the reserved memory.
Size	<p>For each channel, indicates the memory size that is allocated for reading from the server and writing to the server. It corresponds to the read and/or write size that you enter when you create a communication channel with a server counterpart.</p> <p>You can create communication channels until there is not enough sufficient memory left.</p> <p>NOTE:</p> <ul style="list-style-type: none"> For channel type Free, indicates the remaining free memory size that you can use to create communication channels. It is calculated based on the memory size that you have configured in the memory heap of the client by using the ClientMemoryLength parameter, page 592 and on the memory allocation of existing communication channels. Additional words used by the System (for example, for disabling flags) are also deducted from the memory size. For controllers of the M580 platform and NOC communication modules of any controller, the size value for channel type Free is not indicated because it is managed during build only, page 775.
Free	<p>Displays the remaining free memory in words for each existing peer to peer communication channel. The size is indicated separately for read and write channels if they exist. It is calculated based on Size and the size of variables that you have mapped, page 738 to read and/or write communication channels.</p> <p>For example, if the read channel size is 100 and you have mapped variables to this channel, which use 10 words, the displayed value is 89. If you have mapped variables that use less than a word (a variable of BOOL data type), the free size can be indicated in decimals. These correspond to 1/16th of a word. For example, a free memory size indicated as 8.14 (out of 10) corresponds to 8 words and 14/16th of a word (2/16th of a word are used by variables of BOOL data type, which use 1/16th of a word each).</p> <p>NOTE: 1 word of each channel is reserved to move the channel address to manage the 32-bit alignment for peer to peer communication across Modicon Quantum and M340 platforms. This word is deducted from the remaining free memory size.</p>

Column header	Description
Last	Displays the value of the parameter Last value that you can select and adjust through the Channel Properties dialog box, page 733.
Scan	Displays the value of the parameter Scan rate that you can adjust through the Channel Properties dialog box, page 733.
Timeout	Displays the value of the parameter Timeout that you can adjust through the Channel Properties dialog box, page 733.

The table describes the fields of the **Server Communication Counterparts - Device IO** pane.

Field	Description
Service	<i>element category.communication element</i> , where <i>element category</i> is the category of the I/O scanner element, and <i>communication element</i> the name of the I/O scanner element for the I/O device.
Server	Name of the STB island, PRM, or Modbus TCP device entity, acting as server and with which you can create a communication channel.
Description	Description of the data to be exchanged through the channel.
Read Size	Number of words that are read from the server.
Write Size	Number of words that are written to the server.

The table describes the fields of the **Server Communication Counterparts - Peer to Peer** pane.

Field	Description
Service	<i>project.executable</i> Where <i>project</i> is the identifier of the Control Participant project (<i>owner</i> , page 463) that exposes the variables and <i>executable</i> the identifier of the executable of this project.
Server	Name of the controller mapped to the owner project described in the Service field, which acts as counterpart, and that is available to create a communication channel.
Free	If no communication channel exists, indicates the number of words that you have configured in the memory heap of the controller acting as server, which is defined in the Server field. When you create a communication channel, the read and write sizes that you define in the Peer to Peer: Properties dialog box are deducted from this value.

Customizing Column Display

In the various panes of the editor, you can choose which columns you want to see by right-clicking a column header and selecting **Customize**. Certain columns are always shown.

If you hide a column for which a filter is applied, the filter is cleared.

Creating, Modifying, and Deleting Communication Channels

Overview

You can create, modify, move, and delete communication channels by using the **Communication Mapping Editor**, page 725.

Communication channels are used for communication between the following entities and the controller that you configured as client and mapped to the Control executable:

- Modbus TCP devices.
- STB islands.
- PRMs.
- Other controllers that are mapped to other logical Control Participant projects for peer to peer communication.

EcoStruxure Process Expert creates the I/O scanner lines that correspond to the communication channels in the controller acting as client during the build process, page 775.

The creation of a communication channel is prerequisite to mapping network variables, page 738.

Communication Channel Position

Communication channels are contiguous in the memory area of the controller acting as client.

Communication Channel Order

Device I/O communication channels appear in the memory area reserved for this type of communication channel in the order you create them.

When both peer to peer communication channels and device I/O communication channels exist, device I/O communication channels are positioned after peer to peer communication channels.

Restrictions when Creating Communication Channels

You cannot create communication channels with several executables that are mapped to the same controller. This applies to each communication module of a controller acting as client (a tab of the **Communication Channels** pane).

For example, if the following executables appear in the **Service** column of the **Server Communication Counterparts - Peer to Peer** pane and both are mapped (service mapping) to controller *Controller_1* (**Server** column):

- *Project_1.Executable_1*
- *Project_2.Executable_2*

Then, for a given client controller communication module, you can create a communication channel only with either one of these executables.

Creating Communication Channels

If you create a communication channel and you already deployed the associated built Control Participant project, you can only apply the change to the engine by using the **Deploy Built Project**, page 834 command, which requires stopping the controller.

NOTE: Following the creation of a peer to peer communication channel, you can view it in the memory heap of the server, in the **Server Memory Map** pane (<controller_name>(server) tab), by opening the **Communication Mapping Editor** from the counterpart owner project.

Creating a peer to peer communication channel in the executable of the consumer project, page 463 affects the executable of the counterpart owner project. To apply changes, you must start/stop the controllers where these executables are deployed in the appropriate order; otherwise, incorrect data may be read and/or written.

⚠ WARNING

LOSS OF CONTROL

- Deploy the executables to the controllers that communicate by using the same communication channel.
- Start/stop controllers in the appropriate order.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

To create a communication channel between the client and the server counterpart, proceed as follows.

Step	Action
1	From one of the Server Communication Counterparts panes, drag the server counterpart that you want to communication with to the Communication Channels pane. Result: The corresponding communication channel Properties dialog box opens.
2	Adjust the values of the communication channel properties as required.
3	Click OK . Result: <ul style="list-style-type: none"> • A communication channel between the selected server counterpart and the client is created with the parameter values that you configured. • The information of the server counterpart is shifted to the respective columns in the Communication Channels pane. • Values of the Size and Free columns are updated.

NOTE: If there is no sufficient space in the memory of the client or server to create the communication channel, a notification is displayed and you cannot create the channel. Adjust the client or server memory properties of the corresponding controller entity. This verification is not performed for controllers of the M580 platform and BMENOC•••• communication modules because the free memory size is only managed during build, page 775.

Modifying Communication Channel Properties

You can view modify the values of certain parameters of existing communication channels in the **Properties** dialog box of the channel.

If you modify the properties of a communication channel and you already deployed the associated built Control Participant project, you can only apply the change to the engine by using the **Deploy Built Project**, page 834 command, which requires stopping the controller.

Modifying peer to peer communication channel properties in the executable of the consumer project, page 463 affects the executable of the counterpart owner project. To apply changes, you must start/stop the controllers where these executables are deployed in the appropriate order; otherwise, incorrect data may be read and/or written.

⚠ WARNING

LOSS OF CONTROL

- Deploy the executables to the controllers that communicate by using the same communication channel.
- Start/stop controllers in the appropriate order.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

To modify properties of an existing communication channel, proceed as follows.

Step	Action
1	<p>In the Communication Channels pane, right-click the communication channel and select Properties.</p> <p>Result: The Properties dialog box opens.</p> <p>NOTE: The Properties dialog box does not open if the Variable Mapping Editor window, page 738 of this communication channel is open.</p>
2	Adjust the values of the communication channel properties as required.
3	<p>Click OK.</p> <p>Result:</p> <ul style="list-style-type: none"> • The communication channel properties are adjusted. • The values that are displayed in the Communication Mapping Editor are updated. • Information on the modification appears in the notification panel.

NOTE: If EcoStruxure Process Expert is unable to apply your modification because of insufficient free space, it displays a notification. This verification is not performed for controllers of the M580 platform and BMENOC.... communication modules because the free memory size is only managed during build, page 775.

Deleting Communication Channels

Deleting a communication channel has the following impacts:

- The memory size allocated to the channel is released in the memory heap of the client and of the server.
- Other channels that exist in the memory heap of the client are realigned to be contiguous.
- Network variables that are mapped, page 738 to the deleted channel are unmapped from the client and server communication channels.
- The build status, page 781 of the Control executable to which the controller is mapped changes to **Out Of Date** if it is built.

If you delete a communication channel and you already deployed the associated built Control Participant project, you can only apply the change to the engine by using the **Deploy Built Project**, page 834 command, which requires stopping the controller.

Deleting a communication channel affects both the consumer and the counterpart owner projects, page 463. To apply changes, you must start/stop the controllers where these executables are deployed in the appropriate order; otherwise, incorrect data may be read and/or written.

⚠ WARNING

LOSS OF CONTROL

- Deploy the executables to the controllers that communicate by using the same communication channel.
- Start/stop controllers in the appropriate order.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

To delete a communication channel, proceed as follows.

Step	Action
1	In the Communication Channels pane, right-click the communication channel and select Unmap . Result: The Unmap dialog box opens.
2	Click OK to confirm. Result: <ul style="list-style-type: none"> • The communication channel between the selected server counterpart and the client is deleted. • The server counterpart is displayed in the corresponding Server Communication Counterparts pane. • The values of the Size and Free columns are updated.

Moving Communication Channels Server-Side

When you move an existing peer to peer communication channel within the memory heap of the server counterpart, the following applies:

- You can only enter an odd starting address.
- The starting address has to leave enough free memory area to for the communication channel; otherwise you cannot move the channel.

If you move a communication channel and you already deployed the associated built Control Participant project, you can only apply the change to the engine, using the **Deploy Built Project**, page 834 command, which requires stopping the controller.

Moving a peer to peer communication channel in the executable of the owner project affects the executable of the consumer counterpart project, page 463. To apply changes, you must start/stop the controllers where these executables are deployed in the appropriate order; otherwise, incorrect data may be read and/or written.

⚠ WARNING

LOSS OF CONTROL

- Deploy the executables to the controllers that communicate by using the same communication channel.
- Start/stop controllers in the appropriate order.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

To move a peer to peer communication channel in the memory heap of the server counterpart, proceed as follows.

Step	Action
1	Open, page 725 the Communication Mapping Editor from the owner, page 463 project.
2	Select the <controller_identifier>(Server) tab to display the memory heap of the server. Result: The Server Memory Map pane opens.
3	Right-click the communication channel that you want to move and select Properties . Result: The Server Peer to Peer Channel: Properties dialog box opens.
4	Enter a positive, odd integer value in the Start position field.
5	Click OK . Result: If sufficient free memory is available at the specified position: <ul style="list-style-type: none"> • The selected communication channel is moved to the new start position. • The values of the Start Position and Size columns are updated in the Server Memory Map pane of the Communication Mapping Editor. Otherwise, a dialog box opens informing you that the communication channel cannot be moved.

Communication Channel Actions

Right-click a communication channel in the **Communication Channels** pane to open a context menu with the following commands.

Command	Description
Map Variables	Opens the Variable Mapping Editor , page 738, which allows you to map network variables and define the relative position of these variables in the selected communication channel. The command is available only for peer to peer communication channels.
Unmap	Deletes the selected communication channel, page 731.
Properties	Opens the Device IO: Properties or Peer to Peer: Properties dialog box, page 733, which lets you view and/or adjust parameter values of the selected communication channel.

Communication Channel Properties

When you create a communication channel, page 730, the appropriate dialog box opens, which allows you to adjust the values of the parameters of the communication channel in the memory heap of the client.

NOTE: You can view and/or adjust the values of certain parameters of existing communication channels.

The following figure shows an example of the **Device IO: Properties** dialog box that opens when you create a channel to communicate with an STB island. The dialog box is also displayed if you right-click the existing communication channel and select **Properties**.

Service	IOSMedium.IOScanner
Server	Advantys_1
Read description	Diagnostic and Data from Advantys Island
Read size (words)	57
Write description	Diagnostic and Data from Advantys Island
Write size (words)	28
Last value	<input type="radio"/> Reset <input checked="" type="radio"/> Hold
Scan rate (ms)	60
Timeout	1000

OK Cancel

The following figure shows an example of the **Peer to Peer: Properties** dialog box that opens when you create a peer to peer communication channel. The dialog box is also displayed if you right-click the existing communication channel and select **Properties**.

Service	Owner.ControlExecutable_1
Server	Server
Communication Service	Server 1: PriLocal 1: D 1: R 04:140NOE7710... ▼
Read description	Read channel description1
Read size (0-125 words)	30
Write description	Write channel description1
Write size (0-100 words)	5
Last value	<input type="radio"/> Reset <input checked="" type="radio"/> Hold
Scan rate (ms)	60
Timeout (ms)	1000

Controller stop is required to reflect the changes.

OK Cancel

Parameter	Description
Service	Refer to the description of the Service column of the Server Communication Counterparts - Device IO or Server Communication Counterparts - Peer to Peer pane.
Server	Refer to the description of the Server column of the Server Communication Counterparts - Device IO or Server Communication Counterparts - Peer to Peer pane.
Communication Service	<p>The parameter only appears in the Peer to Peer: Properties dialog box.</p> <p>Name of the communication module through which data is exchanged.</p> <p>You can select from the communication modules that exist in the server controller and that are connected to the same logical network as the client controller communication module.</p>

Parameter	Description
Read description	<p>Description for the read channel.</p> <p>NOTE: For peer to peer communication channels, the default value is blank. You must enter a description.</p>
Read size	<p>Number of words that are allocated to the channel and read from the server counterpart.</p> <p>The corresponding memory size is reserved in both the client and server memory heap.</p> <p>NOTE: For peer to peer communication channels the value must fulfill the following requirements:</p> <ul style="list-style-type: none"> • Default value: 0 • Maximum value: 125 • You can adjust the value as follows: <ul style="list-style-type: none"> ◦ The total value of columns Read size and Write size cannot exceed the lowest value indicated in either the Free column of the Server Communication Counterparts pane or the Size column (row Free) of the Communication Channels pane. ◦ You cannot configure a size that is lower than the total size of variables mapped, page 738 to the channel. ◦ You can increase the size if there is no overlapping channel in the server memory heap within the new size limit.
Write description	<p>Description for the write channel.</p> <p>NOTE: For peer to peer communication channels, the default value is blank. If you enter a value in the Write size field, you must enter a description.</p>
Write size	<p>Number of words that are allocated to the channel and written to the server counterpart.</p> <p>The corresponding memory size is reserved in both the client and server memory heap.</p> <p>NOTE: For peer to peer communication channels the value must fulfill the following requirements:</p> <ul style="list-style-type: none"> • Default value: 0 • Maximum value: 100 • You can adjust the value as follows: <ul style="list-style-type: none"> ◦ The total value of columns Read size and Write size cannot exceed the lowest value indicated in either the Free column of the Server Communication Counterparts pane or the Size column (row Free) of the Communication Channels pane. ◦ You cannot configure a size that is lower than the total size of variables mapped, page 738 to the channel. ◦ You can increase the size if there is no overlapping channel in the server memory heap within the new size limit.
Last value	<p>Value that is retained when a communication interruption occurs.</p> <p>Select between the following:</p> <ul style="list-style-type: none"> • Reset: Set to 0 • Hold: Last value <p>Default value: Hold</p> <p>NOTE: This setting applies to the I/O scanner line created in the Control Participant project. When communicating with an I/O device modeled by a device template, ensure that the DFB contained in the device template supports the selected setting. Otherwise, the value of variables read from the device may not be synchronized with the value sent to the Supervision layer by the DFB.</p> <p>For information on parameters of DFBs of device templates, refer to the description of the Control services of the template in the EcoStruxure Process Expert help.</p>

Parameter	Description
Scan rate	<p>The rate at which data is scanned in the channel.</p> <p>Range: 0...65535 ms</p> <p>Default value: 60 ms</p> <p>If you change the default value, ensure that the new value is a multiple of the repetitive rate step.</p> <p>For more information, refer to the Control Participant help.</p>
Timeout	<p>The maximum interval between responses from the entity for the channel. After this time expires, the communication is considered interrupted and the last value is retained according to the configuration of the Last value parameter.</p> <p>Range: 1...65535 ms</p> <p>Interval: 1 ms</p> <p>Default value: 1000 ms</p> <p>NOTE: Ensure that the TimeOut value is greater than the ScanRate value.</p> <p>For more information, refer to the Control Participant help.</p>

Server-Side Communication Channel Properties

Once you have created a peer to peer communication channel in the memory heap of the client, you can view properties of the server-side channel by right-clicking the communication channel and selecting **Properties**.

To view the communication channel, open the **Communication Mapping Editor** from the owner project.

NOTE: You can open an identical properties dialog box for the communication channel that exists in the client controller when acting as server by selecting the <controller_name>(**Server**) tab from the client-side communication mapping window, right-clicking the communication channel in the **Server Memory Map** pane, and selecting **Properties**.

NOTE: If you modify the properties of a communication channel server-side and you already deployed the associated built Control Participant project, you can only apply the change to the engine by using the **Deploy Built Project**, page 834 command, which requires stopping the controller.

The following figure shows an example of the **Server Peer to Peer Channel: Properties** dialog box showing the properties of the server-side peer to peer communication channel.

Server Peer to Peer Channel: Properties	
Partner	Owner.ControlExecutable_1
Client	Client_Ctrlr
Description	Read description3
Direction	Read ▼
Start Position (ODD Value)	1
Size (words)	20
<div>OK Cancel</div>	

Parameter	Description
Partner	<i>project.executable</i> Where <i>project</i> is the identifier of the Control Participant project (consumer, page 463) that receives the variables and <i>executable</i> the identifier of the executable of this project.
Client	Name of the controller entity (server) to which the owner project is mapped.
Description	Description for the read or write channel.
Direction	Read or Write .
Start position	Starting address of the communication channel in the memory heap of the server. The value is editable and allows you to move communication channels server-side, page 732.
Size	Number of words that are allocated to the communication channel server-side.

Variable Mapping Editor

Overview

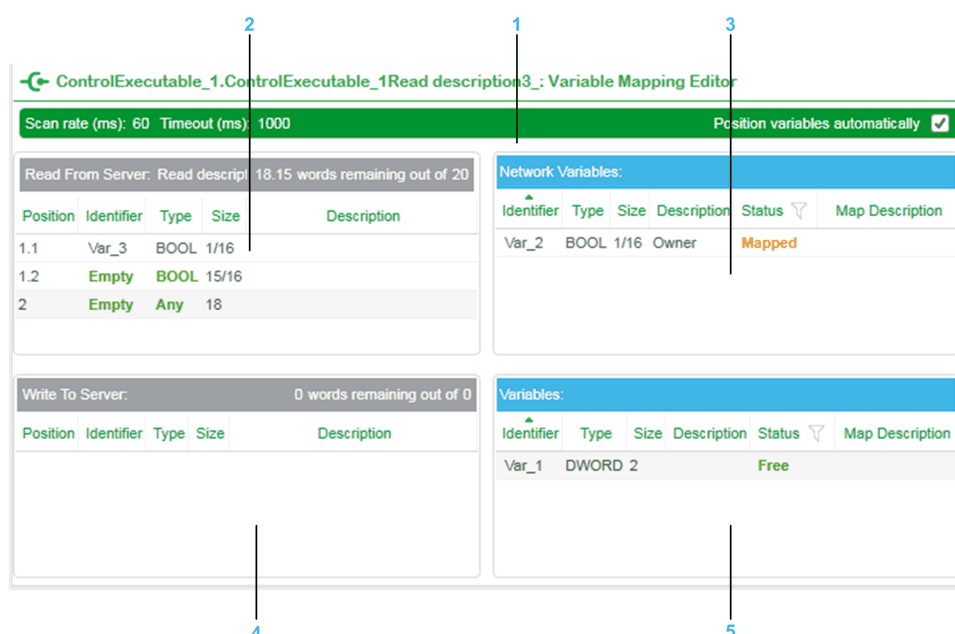
The **Variable Mapping Editor** opens when you right-click a peer to peer communication channel in the **Communication Mapping** pane, page 726 and select **Map Variables**.

It allows you to map variables, page 741 of the consumer project to peer to peer communication channels that exist in the memory heap of the client.

NOTE: The first time you open the **Variable Mapping Editor** and the first time you open it after refining the project, it may take time before the window appears.

Variable Mapping Editor

The following figure shows an example of the **Variable Mapping Editor**.



Item	Description
1	Information bar, which displays scan rate and timeout properties of the selected communication channel, as well as the check box to position variables automatically, page 742.
2	Pane that displays the network variables that the client reads from its server counterpart, and that are mapped to the selected communication channel. In this example, network variable <i>Var_3</i> , which is created in the consumer project is mapped to the communication channel that exists in the client controller.
3	Pane that displays the network variables that exist in the consumer project and that you can map to the communication channel that is created in the client controller. In this example, network variable <i>Var_2</i> has already been mapped by opening the Variable Mapping Editor from the owner project and using its counterpart variable to write to the server, page 463. You cannot map it again. You can view its mapping information by right-clicking Mapped .
4	Pane that displays the variables that the client writes to its server counterpart, and that are mapped to the selected communication channel. In this example, no variables are mapped.
5	Pane that displays the variables that exist in the consumer project for which a corresponding network variable exists in the owner project and that you can map to the communication channel. In this example, <i>Var_1</i> exists as network variable in the owner project and here you can map its counterpart variable to the channel that exists in the client controller to write to the server, page 463.

The table describes the fields of the **Network Variables** and **Variables** panes.

Column header	Description
Identifier	Identifier of the variable.
Type	<p>Data type of the variable.</p> <p>If EcoStruxure Process Expert detects inconsistencies in the type, the variable is shown in red color and a tooltip provides additional information.</p> <p>NOTE: When peer to peer communication with the topological entity acting as client is implemented by using an M580 CPU or a NOC communication module, only variables of the reference type (REF_TO) are shown, page 467.</p>
Size	<p>Size of the variable in words.</p> <p>NOTE: For variables of a size of less than 1 word, the size is displayed in corresponding fractions of words (for example, for data type BOOL, the size is displayed in 1/16th of words).</p>
Description	Displays the contents of the Comment property of the Control Participant of the variable in the consumer project.
Status	<p>Indicates the mapping status of the variable.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Free: The variable is not yet mapped to a communication channel. • Mapped: The variable is already mapped to another communication channel of the Control executable. <p>Click the filter icon in the column header to select the statuses that you want to see.</p> <p>NOTE: You can open the Variable Mapping Editor of the communication channel to which a variable with status Mapped is mapped by right-clicking the variable and selecting Go To Variable Mapping Editor.</p>
Map Description	For variables that are mapped to another communication channel, displays the content of the Read description or Write description field of this channel; otherwise the field is empty.

The table describes the fields of the **Read from Server** and **Write to Server** panes.

Column header	Description
Pane title bar	<p>For the Read from Server pane, displays:</p> <ul style="list-style-type: none"> • The contents of the field Read description. • The read size (in words) of the selected channel in the client memory heap and the remaining words available to map variables. <p>For the Write to Server pane, displays:</p> <ul style="list-style-type: none"> • The contents of the field Write description. • The write size (in words) of the selected channel in the client memory heap and the remaining words available to map variables. <p>NOTE: One word of each channel is reserved to move the channel address to manage the 32-bit alignment, page 778 for peer to peer communication across Modicon Quantum, M580, and M340 platforms.</p>
Position	<p>Position of the word to which you have mapped the variable within the memory size that you have allocated to the channel or the next available empty position within a word or of the next word.</p> <p>The position in the communication channel is the same in the client and in the server.</p> <p>For variables of a size of less than 1 word, positions are displayed using the x.y format where:</p> <ul style="list-style-type: none"> • x represents the word • y represents the position inside the word
Identifier	<p>Identifier of the mapped variable or Empty.</p> <p>Empty: Appears for the next available position to which you can map variables.</p>
Type	<p>Type of the mapped variable, of the remaining space in a word, or Any.</p> <p>To map variables to the remaining space of a word, the data type of the variables must match.</p> <p>Any: Appears when Identifier is Empty. You can map a variable of any data type to this space.</p>

Column header	Description
Size	Indicates either, the size of the mapped variable in words, the size of the remaining space inside a word, or the size of the empty space available for mapping. NOTE: For mapped variables of a size of less than 1 word, the size is displayed in corresponding fractions of words (for example, for data type BOOL, the size is displayed in 1/16th of words). The empty space of a word is indicated similarly.
Description	Displays the contents of the Comment property of the Control Participant of the variable in the consumer project.

Customizing Column Display

In the various panes of the editor, you can choose which columns you want to see by right-clicking a column header and selecting **Customize**. Certain columns are always shown.

When you hide a column for which a filter is applied, the filter is cleared.

Mapping Network Variables

Overview

By using the **Variable Mapping Editor**, page 738, you can map the following variables of the consumer project to peer to peer communication channels that exist in the memory heap of the client:

- Network variables, page 463: To read data from the server.
- Variables that are the counterpart to network variables that exist in the owner project: To write data to the server.

You can assign a position manually or let EcoStruxure Process Expert manage positions.

EcoStruxure Process Expert indicates if variables are already mapped to another channel in the **Variable Mapping Editor** and lets you open the channel.

NOTE: You can work simultaneously on different communication channels. The variable mappings that are performed on a channel and that affect other channels in the **Variable Mapping Editor** are reflected right away on these channels.

Prerequisites

To map variables, first create a peer to peer communication channel, page 725.

Position of Mapped Variables

The **Position variables automatically** check box lets you define how variables are positioned:

- The check box is selected (default setting): The variable is automatically mapped or moved to the compatible empty position that you selected.

If you selected several variables, the following actions are performed:

- If the space at the empty position that you have chosen allows it, variables are mapped or moved contiguously starting from this position.
- If there is not enough contiguous space at the empty position that you have chosen to map or move all variables but is sufficient for at least one variable, the **Mapping Confirmation** dialog box opens to inform you. You have the following options:

Yes: Maps or moves the variables for which sufficient space is available at the chosen empty position and maps or moves the other variables to the next available positions.

No: Maps or moves only the variables for which sufficient space is available at the chosen empty position but does not map or move the other variables. Then, a dialog box opens to indicate which variables could not be mapped or moved. These variables remain in the **Network Variables** or **Variables** section.

Cancel: Cancels the mapping or move operation.

- The check box is cleared: The **Select Memory Position** dialog box opens and you can select an available, compatible position to map or move the variable within the read or write size that you defined for the channel.

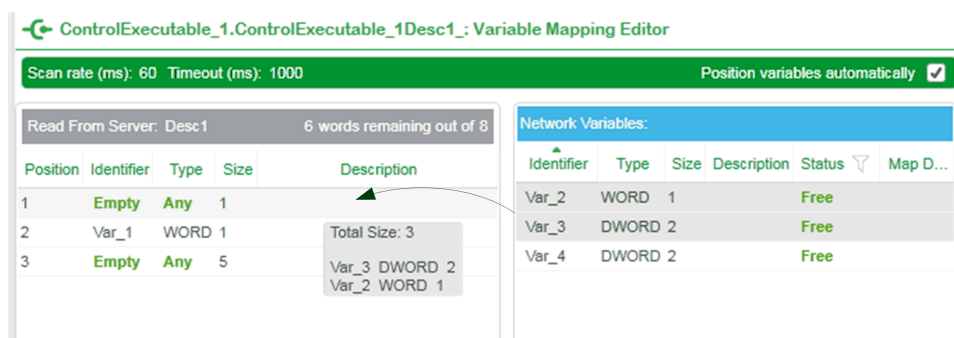
If you selected several variables, the position that you select in the **Select Memory Position** dialog box applies only to the first variable. The other variables are mapped or moved to the contiguous space that is available after the position that you selected. If not sufficient contiguous space is available, a message indicates which variables could not be mapped or moved.

The **Mapping Confirmation** dialog box opens only if there is empty space in a channel before and after an already mapped variable.

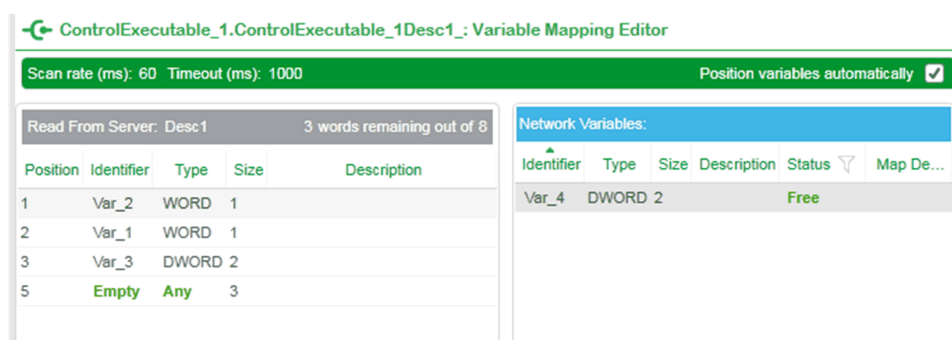
NOTE: You can map 32-bit variables (for example, variables of type REAL or DWORD) only to odd addresses in the **Variable Mapping Editor**. The variables are aligned during the build process, page 778 depending on the controller platform.

Example of Variable Mapping Editor Usage

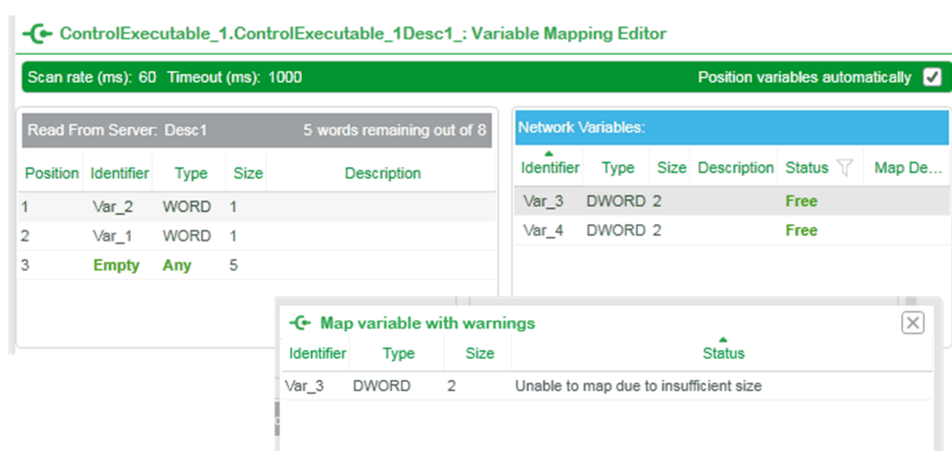
In this example, network variable *Var_1* is already mapped to a peer to peer communication channel and there is one word of free space available before (position 1) and five words available after (position 3). The **Position variables automatically** check box is selected. *Var_2* (WORD) and *Var_3* (DWORD) are selected and dragged to position 1. Because there is only sufficient space to map one of the variables at position 1, the **Mapping Confirmation** dialog box opens when you release the mouse button.



If you click **Yes**, EcoStruxure Process Expert maps *Var_2* at position 1, which you selected and *Var_3* automatically at position 3, which is the next available space where the variable fits.



If you click **No**, EcoStruxure Process Expert maps only *Var_2* at position 1. Then, it informs you that *Var_3* is not mapped because of insufficient space. *Var_3* remains in the **Network Variables** section and is available for mapping.



Variable Mapping Rules

The following rules apply when mapping a variable:

- A variable cannot be mapped to a position overlapping another mapped variable.
- A variable cannot be mapped if:
 - The variable is split up between two or more positions.
 - The variable does not fit entirely inside the communication channel.

Mapping Network Variables

Mapping variables to a peer to peer communication channel affects the executables of both the consumer and the counterpart owner projects, page 463. To apply changes, you must start/stop the controllers where these executables are deployed in the appropriate order; otherwise, incorrect data may be read and/or written.

WARNING

LOSS OF CONTROL

- Deploy the executables to the controllers that communicate by using the same communication channel.
- Start/stop controllers in the appropriate order.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

To map network variables to a communication channel (to read from the server), proceed as follows.

Step	Action	Result
1	In the Communication Mapping Editor , right-click the communication channel to which you want to map variables and select Map Variables .	Result: The Variable Mapping Editor opens.
2	Clear the Position variables automatically check box if you want to manually select the position of the variable.	—
3	In the Network Variables pane, drag the variables that you want to map onto a compatible empty position in the Read from Server pane.	Result: If you cleared the Map on Top check box, the Select Memory Position dialog box opens. Proceed to step 4; otherwise: <ul style="list-style-type: none"> • The variables are mapped and displayed in the Read from Server pane. • The variable are removed from the Network Variables pane. • A message indicates if a variable could not be mapped.
4	In the Select Memory Position dialog box, click the Memory Position menu and select a position to map the variable.	—
5	Click OK . NOTE: Click Cancel to close the Select a Memory Position dialog box without mapping the variable.	Result: <ul style="list-style-type: none"> • The variables are mapped and displayed in the Read from Server pane. • The variable are removed from the Network Variables pane. • A message indicates if a variable could not be mapped.

NOTE: To map a variable to write to the server, proceed as described above but use the **Variables** and **Write to Server** panes of the **Variable Mapping Editor**.

Changing the Position of a Mapped Variable

To change the position of a mapped variable, drag it to the new position.

You can select several variables to move them at once.

Moving variables mapped to a peer to peer communication channel affects the executables of both the **consumer** and the **counterpart owner** projects, page 463. To apply changes, you must start/stop the controllers where these executables are deployed in the appropriate order; otherwise, incorrect data may be read and/or written.

WARNING

LOSS OF CONTROL

- Deploy the executables to the controllers that communicate by using the same communication channel.
- Start/stop controllers in the appropriate order.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

To change the position of mapped network variables or variables with the **Position variables automatically** check box selected, select the variables in the **Read from Server** or **Write to Server** pane and drag them to the new compatible, empty position.

Result:

- The variables are removed from their existing positions and mapped.
- A message indicates if a variable could not be mapped.

To change the position of mapped network variables or variables with the **Position variables automatically** check box cleared, proceed as follows.

Step	Action	Result
1	Select the variables in the Read from Server or Write to Server pane and drag them to the new compatible, empty position.	Result: The Select Memory Position dialog box opens.
2	In the Select Memory Position dialog box, click the Memory Position men and select a position to map the variable.	-
3	Click OK . NOTE: Click Cancel to close the Select a Memory Position dialog box without mapping the variable.	Result: <ul style="list-style-type: none"> • The variables are mapped. • A message indicates if a variable could not be mapped.

Removing a Variable From a Communication Channel

To remove a mapped variable from the communication channel, unmap it.

Removing variables from a peer to peer communication channel affects the executables of both the [consumer](#) and the [counterpart owner projects](#), [page 463](#). To apply changes, you must start/stop the controllers where these executables are deployed in the appropriate order; otherwise, incorrect data may be read and/or written.

WARNING

LOSS OF CONTROL

- Deploy the executables to the controllers that communicate by using the same communication channel.
- Start/stop controllers in the appropriate order.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

To remove a network variable or variable from a communication channel, right-click the variable in the **Read from Server** or **Write to Server** pane and select **Unmap**.

Result:

- The variable is removed from the selected position.
- The variable is removed from the corresponding **Read from Server** or **Write to Server** pane.
- The variable is displayed in the corresponding **Network Variables** or **Variables** pane.

NOTE: You can also right-click the variable and press **Del**.

Deleting or Changing Mapped Variables

A network variable or variable is removed from the communication channel if you delete either of the following:

- The network variable itself.
- The corresponding variable in the owner project.

It is also removed from the communication channel if you change either of the its following properties in the owner or the consumer project:

- Name
- Type (ID or version)
- Size

Relevant information is displayed in the notification panel.

Mapping Hardware

Overview

To open the **Hardware Mapping Editor**, click the **Hardware Mapping** tab within the **Manage** window, page 721. The tab appears once you have performed the service mapping, page 723 with a controller.

It allows you to map the interfaces of instances of the application to compatible hardware interfaces representing the I/O and communication modules of the entity that you have service mapped to the Control executable of the project.

Application objects that expose mapping interfaces are instances of either:

- HAL templates (for example, *\$DISignal_UL*).
- Communication port templates (for example, *\$EMPortM*).
- Hardware application templates (for example, *\$TesysTE* or *\$TSignCptBmx*).

The mapping interfaces of application objects appear in the **Hardware Mapping Editor** once you assign their facets to the Control project; even without generating them. They are updated in the mapping editor when you make changes in the application (for example, rename an instance or disable a service) and/or generate the Control project (for example, after setting a facet to **Out Of Date** or **Deleted**).

You can map interfaces to modules that are connected:

- Physically (for example, in-rack I/O modules).
Their topological mapping interfaces are updated in the mapping editor when you *save changes*, page 592 in the Control Participant **Configuration** window.
- Logically (for example, Modbus TCP Device I/Os, PRMs, or STB islands).
Their topological mapping interfaces appear in the mapping editor when you *connect*, page 615 the device to the same logical network as the controller.

If you have created several executables, proceed with the hardware mapping for each one of them.

To facilitate the management of many mappings, you can export and import hardware mappings, page 754.

NOTE: EcoStruxure Process Expert marks variables and DFB instances, page 784 that are used for the hardware mapping when you build the logical Control Participant project.

Mapping Modbus TCP Devices, PRMs, and STB Islands

For the following entities communicating with the controller through Modbus TCP implicit messaging (I/O scanning) and that are connected to the same logical network as the controller, you can proceed with the hardware mapping as follows:

- Modbus TCP I/O devices: Does not require creating communication channels.
- STB islands and PRMs: Requires creating a communication channel, page 725.

NOTE: Instantiate the necessary application templates first. For details, refer to the topic describing how to implement communication, page 935.

Mapping Hardware with a Simulator

Hardware mapping is not supported and not required for executables that you have mapped to a workstation running the controller simulator.

Mapping Generic Device Interfaces

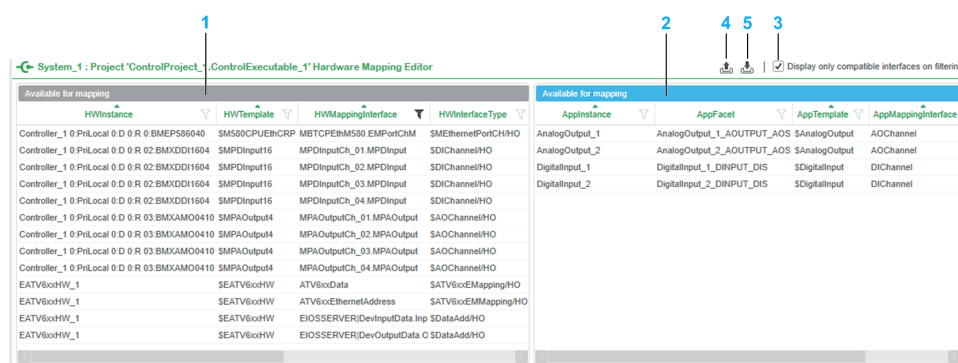
When a topological entity uses the *\$EGenericDeviceHW*, *\$EIPGenericDeviceHW*, or *\$EGenericPBSlave* template to model a device, a generic hardware mapping interface is available for this device in the **Hardware Mapping Editor**. You can use it to map the interface of a project facet to this I/O device. However, because of the generic nature of the interface, EcoStruxure Process Expert does not verify if both interfaces are compatible, page 751 before proceeding with the mapping.

Opening the Hardware Mapping Editor in Read-Only Mode

The editor can open in read-only mode, for example, if the executable is locked by another process such as the configuration of the controller. In this case, hardware mapping creation, modification, and import are not allowed.

Hardware Mapping Editor

The following figure shows an example of the **Hardware Mapping Editor** window.



Item	Description
1	Topological devices pane (left, gray title bar), which displays the interfaces of the controller entity that you have mapped to the executable of the Control project. It also displays interfaces of Modbus TCP/EtherNet/IP device, PRM, and STB island entities that are connected to the same logical network as the controller. A communication mapping for PRM and STB island entities needs to exist.
2	Project facet pane (right, blue title bar), which displays the mapping interfaces of the facets that you have assigned to the Control project, including interfaces of facets of Modbus TCP/EtherNet/IP explicit messaging device instances, Modbus serial device instances, and the corresponding communication port instances.
3	Check box to display only compatible interfaces, page 749 that meet the selected filter criteria in both panes. The functionality helps locate compatible interfaces when you have many different ones in either pane. The functionality is enabled by default.
4	Lets you export, page 754 to file (.csv) the existing hardware mappings and the interfaces of project facets that have not been mapped yet.
5	Lets you import, page 754 valid and compatible hardware mappings that you have created outside the Hardware Mapping Editor and that are contained in a file (.csv).

NOTE: When the executable of the Control project is mapped to a redundant controller, the topological devices pane displays only the hardware interfaces of the primary controller.

Editor section	Column header	Description
Topological devices pane	HWInstance	Identifier of the topological device. Displays: <ul style="list-style-type: none"> Controller devices. Modbus TCP and EtherNet/IP devices that are physically connected to the same network as the controller. PRM entities and STB island devices after you have completed the communication mapping.
	HWTemplate	Identifier of the template that is used by the topological device.
	HWMapping Interface	Identifier of the facet that models the hardware channel followed by the identifier of the facet template.
	HWInterfaceType	Type of the mapping interface and role.
	AppInstance	These columns are blank before you start the mapping process. The information contained in the corresponding columns of the project facet pane is moved to these columns when you map the project facet interface.
	AppFacet	
	AppTemplate	
	AppMapping Interface	
	AppInterface-Type	
Project facet pane	AppInstance	Identifier of the instance that references the facet.
	AppFacet	Identifier of the facet that is assigned to the Control project.
	AppTemplate	Identifier of the template that is used by the instance.
	AppMapping Interface	Identifier of the hardware mapping interface that is exposed by the facet.
	AppInterface-Type	Type of the mapping interface and role. The prefix of the type indicates the type of I/O hardware channel: <ul style="list-style-type: none"> DI: Digital input DO: Digital output AI: Analog input AO: Analog output

Customizing Column Display

In the various panes of the editor, you can choose which columns you want to see by right-clicking a column header and selecting **Customize**. Certain columns are always shown.

When you hide a column for which a filter is applied, the filter is cleared.

Displaying Compatible Interfaces

To show only interfaces that are compatible in both panes, proceed as follows.

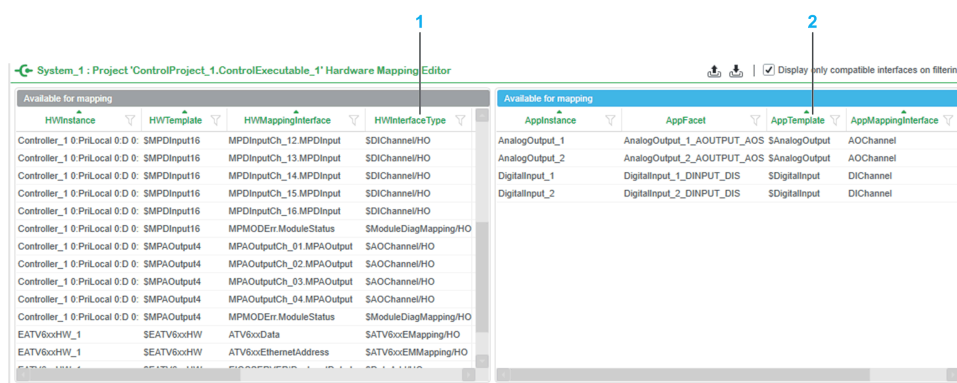
Step	Action
1	Verify that the Display only compatible interfaces on filtering check box, page 748 is selected.
2	In either pane, click the filter icon of the column that you want to use to filter interfaces. Result: The filter menu, page 120 opens.

Step	Action
3	Enter criteria to be used as filter and click Filter . Result: <ul style="list-style-type: none"> In the pane in which you have applied the filter, only interfaces that meet the filter criteria are shown. In the other pane, a filter is automatically applied to the *InterfaceType column to show only interfaces that are compatible with those that result of the filter that you have applied.
4	You can apply additional filters in either pane to further refine results.

NOTE:

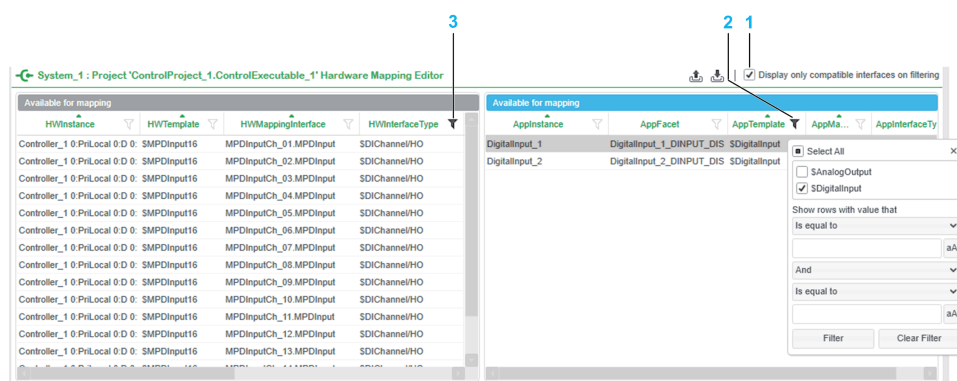
- When you clear the filter in either pane, a filter is applied to the ***InterfaceType** column in the other pane to show only interfaces that are compatible with those displayed in the pane where you have cleared the filter.
- To clear filters in both panes, clear the **Display only compatible interfaces on filtering** check box.

The following example illustrates the use of the automatic filter mechanism to display only compatible interfaces. The figures shows only a partial view of the **Hardware Mapping Editor** to simplify the example.



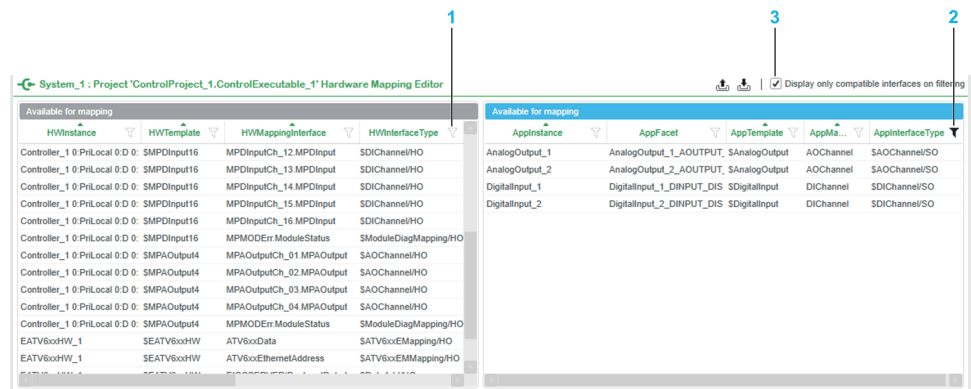
Item	Description
1	A controller is configured with an analog input, an analog output, and a digital input module. The corresponding interfaces appear in the HWInterfaceType column of the topological devices pane.
2	Facets of analog input, digital input and digital output templates are assigned to the project and the template identifiers appear in the AppTemplate column of the project facet pane.

In this step, the objective is to show only topological interfaces that are compatible with digital input interfaces to perform the hardware mapping between the two.



Item	Description
1	Verify that the check box, page 748 is selected.
2	The AppTemplate column, a filter was applied to show only interfaces of digital input templates. One interface meets this criteria.
3	In the topological devices pane, a filter is automatically applied to the HWInterfaceType column to show only interfaces of digital input modules. Interfaces of other modules are not shown anymore. You can map the interface of the project facet to any of these topological input channels.

In this step, the objective is to show only project facet interfaces that are compatible with the topological interfaces of the modules present in the controller configuration. This can be helpful to verify if any hardware-compatible project facet interfaces have not yet been mapped.



Item	Description
1	In the topological devices pane, the filter that was automatically applied to the HWInterfaceType column was cleared by the user. The column now shows all the interfaces of the various modules of the controller to which you can map project facet interfaces.
2	In the project facet pane, the filter of the AppTemplate column was cleared. Instead, a filter was applied to the AppInterfaceType column to show only interfaces that are compatible with the topological interfaces that are exposed by the configuration. As a result, the interface of the digital output template is not shown anymore because the configuration does not contain a digital output module.
3	The check box must be selected.

Mapping Hardware

To map facets of the Control project to topological interfaces of an entity, proceed as follows.

Step	Action
1	In the right-hand project facet pane, locate and select the facets of the Control project that you want to map. Use filters if necessary. Result: The selected facets are highlighted.
2	In the left hand, topological device pane, locate the topological interfaces to which you want to map the facets that you have selected in step 1. Use filters if necessary. Result: The selected instance is highlighted. NOTE: For the mapping to succeed, the type of the interface that is displayed in both *InterfaceType columns needs to be identical and the interface roles need to be compatible. For example, interface type <i>\$DChannel</i> with roles <i>HO</i> (Hardware Object) and <i>SO</i> (Signal Object).
3	From the project facet pane, drag the selected project facets onto the topological interfaces. Result: The information of the project facets is shifted to the respective columns of the topological device pane. NOTE: If the interfaces do not match, a tooltip opens and they are not mapped.

NOTE: To map an already mapped project facet interface to a different topological device with matching interface, drag the mapped project facet onto another compatible topological device interface.

Deleting Hardware Mappings

To delete the mapping of one or more already mapped project facet interfaces, proceed as follows.

Step	Action
1	In the topological device pane, select the rows that correspond to the mappings.
2	To proceed, either: <ul style="list-style-type: none"> • Drag the selection back to the project facet pane. • Press Del and confirm the command. • Right-click the selection, click Unmap, and confirm the command. Result: The selected facet appear again in the project facet pane. NOTE: To delete all the mappings at once, click inside the topological device pane, press Ctrl+A , and use Del or Unmap .

Mapping Several Project Facet Interfaces Simultaneously

You can drag more than one project facet interface onto the topological device pane at once by using multiple selection. The project facet interfaces are mapped to contiguous topological interfaces in the order they are displayed in the tooltip while dragging them.

The interfaces of each project facet/topological device pair need to match to complete the mapping process.

If there is a missing or incompatible topological interface in between, the first matching interfaces are mapped but not the remaining ones, which appear again in the right-hand project facet pane. A message informs you of the project facet interfaces that could not be mapped.

NOTE: Use the filters to display topological devices in the desired sequence.

Changing the Application and Projects After Mapping

If you make any of the following changes to a facet that you have already mapped by using the **Hardware Mapping Editor**, the mapping is deleted and the facet is available again for mapping:

- Unassigning, page 391 the facet and generating the project or section.
- Deleting the instance referencing the facet.
- Unselecting the element, page 183 that corresponds to the facet.

Changing the Topology After Mapping

If you make any of the following changes to topological devices whose interfaces you have already mapped by using the **Hardware Mapping Editor**, the mapping may be deleted. In this case, the corresponding project facets are available again for mapping:

- Deleting the controller entity.
- Changing the CPU module of the controller entity.
- Deleting the I/O device.
- Moving the I/O device.
- Deleting the communication channel with an STB island or a PROFIBUS PRM Master.
- Updating templates of topological entities.

Changing Hardware Mapping After Deploying

Consider a Control project that contains one or more facets representing signals that need to be mapped to I/O modules.

You build and deploy the Control project but without doing these hardware mappings.

If you then do the hardware mapping, build the Control project, and select the **Deploy changes / Undo Online Changes** command to [deploy the hardware mapping change, page 830](#), the new initial value of the variables that correspond to the mapped signals are deployed.

However, the existing value of these variables remains unchanged in the deployed Control project.

The change becomes effective, either through a cold start of the controller or by replacing the existing value of the variables in the animation table by their new initial value.

For example, you deploy a Control project without mapping the facet interface representing the signal quality of an *AnalogInput* instance to the facet interface representing the analog input module channel of the controller.

If you refine the deployed Control project online, the initial value of the variable assigned to the **ComStatus** pin of the corresponding DFB is displayed as *False* because the hardware mapping is not done. Its existing value is *0*.

If you proceed with the hardware mapping, build the Control project, and execute the **Deploy changes / Undo Online Changes** command, the initial value of this variable is changed to *True*.

However, its existing value remains *0*.

NOTE: The same applies when you change an existing hardware mapping after deploying it and deploy it, using the **Deploy changes / Undo Online Changes** command.

Exporting and Importing Hardware Mappings

Overview

The functionality lets you manage, in a three-step process, a large number of hardware mappings, page 747 by:

- Exporting the data that is shown in the **Hardware Mapping Editor** to comma-separated data format (CSV).
- Editing the file so that it contains the hardware mappings that you want to create or modify.
- Importing the file into the **Hardware Mapping Editor** of the same or a different Control executable.

Before proceeding with the import, EcoStruxure Process Expert validates the data and informs you of invalid entries, which will not be imported.

Exporting Hardware Mapping Data

Overview

In a first step, the export functionality lets you create export files that contain the data of the **Hardware Mapping Editor** for a given Control executable.

Good Practices

- Before modifying hardware mappings, back up all the existing hardware mappings by exporting them. If needed, it lets you restore the original configuration by importing the file.
- Apply a filter before exporting so that the export file contains only the topological interfaces whose hardware mapping you want to create or modify.

Export File Compatibility

Hardware mapping export files are forward compatible unless mentioned otherwise.

Export File Description

When you export hardware mapping data, two export files are created. Each one contains data in a comma-separated, page 236 format (CSV).

By default, export files are saved at the path %ProgramData%\Schneider Electric\Process Expert x\Db, where x represents the installed version. When this location does not exist (for example, if only clients are installed on the computer), files are saved at either location:

- At the path where you have last exported or imported any file of EcoStruxure Process Expert.
- If the above does not apply or is not reachable, in the Windows **Documents** folder.

Default file name	Description
<Executable identifier>_HWMapping where _HWMapping is appended to the file name that you enter once the file is created.	<p>The file contains the same data as shown in the topological devices pane, page 748 of the Hardware Mapping Editor at the time of export.</p> <p>It is a snapshot of:</p> <ul style="list-style-type: none"> • The mapping interfaces of the topological entity that is mapped to the Control executable (columns A to D⁽¹⁾). • The mapping interfaces of application objects that are already mapped (columns E to I⁽¹⁾). <p>If you have applied a filter, page 749 before exporting, only the filtered data is exported.</p> <p>The first row contains the column headers of the editor.</p> <p>You need to edit the content, page 757 of this file to create, modify, or delete hardware mappings.</p>
<Executable identifier>_Pending where _Pending is appended to the file name that you enter once the file is created.	<p>This supporting file contains the same data as shown in the project facet pane, page 748 of the Hardware Mapping Editor at the time of export.</p> <p>It is a snapshot of the mapping interfaces of application objects that are available for mapping.</p> <p>If you have applied a filter, page 749 before exporting, only the filtered data is exported.</p> <p>The first row contains the column headers of the pane in the editor.</p> <p>You can use this file to perform cut-and-paste operations to move to the <Executable identifier>_HWMapping.csv file the data that corresponds to hardware mappings that you want to create.</p>
(1) When you open the file by using Microsoft® Excel®	

Example of Export Files

The following figure shows an example of the **Hardware Mapping Editor** where:

- In the topological devices pane, four interfaces of digital input project facets are mapped to the corresponding hardware channel interfaces.
- In the project facet pane, four interfaces of analog output project facets are shown as available for hardware mapping.

System_1 : Project 'ControlProject_1.ControlExecutable_1' Hardware Mapping Editor

Available for mapping									
HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType	
Controller_1 0 PrtLocal 0 D 0: SM580CPUETHCRP	MBTCEPBM580 EMPortChM	SMEthernetPortCh/HO							
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_01 MPDInput	SDIChannel/HO	DigitalInput_1	DigitalInput_1	DigitalInput_1_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_02 MPDInput	SDIChannel/HO	DigitalInput_2	DigitalInput_2	DigitalInput_2_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_03 MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_04 MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPMODEnt ModuleStatus	SModuleDiagMapping/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_01 MPAOutput	SAOChannel/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_02 MPAOutput	SAOChannel/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_03 MPAOutput	SAOChannel/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_04 MPAOutput	SAOChannel/HO							

Available for mapping				
AppInstance	AppFacet	AppTemplate	AppMa...	AppInterfaceType
AnalogOutput_1	AnalogOutput_1_AOUTPUT	\$AnalogOutput	AOChannel	SAOChannel/SO
AnalogOutput_2	AnalogOutput_2_AOUTPUT	\$AnalogOutput	AOChannel	SAOChannel/SO
AnalogOutput_3	AnalogOutput_3_AOUTPUT	\$AnalogOutput	AOChannel	SAOChannel/SO
AnalogOutput_4	AnalogOutput_4_AOUTPUT	\$AnalogOutput	AOChannel	SAOChannel/SO

The following figure shows the content of the exported hardware mapping data file (*_HWMMapping.csv) that corresponds to the previous example.

	A	B	C	D	E	F	G	H	I
HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType	
Controller_1 0 PrtLocal 0 D 0: SM580CPUETHCRP	MBTCEPBM580 EMPortChM	SMEthernetPortCh/HO							
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_01 MPDInput	SDIChannel/HO	DigitalInput_1	DigitalInput_1	DigitalInput_1_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_02 MPDInput	SDIChannel/HO	DigitalInput_2	DigitalInput_2	DigitalInput_2_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_03 MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPDInput16	MPDInputCh_04 MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DICchannel	SDIChannel/SO	
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPMODEnt ModuleStatus	SModuleDiagMapping/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_01 MPAOutput	SAOChannel/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_02 MPAOutput	SAOChannel/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_03 MPAOutput	SAOChannel/HO							
Controller_1 0 PrtLocal 0 D 0: SMPAOutput4	MPAOutputCh_04 MPAOutput	SAOChannel/HO							

And the following figure shows the content of the supporting file (*_Pending.csv).

	A	B	C	D	E
AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType	
AnalogOutput_1	AnalogOutput_1_AOUTPUT_AOS	\$AnalogOutput	AOChannel	SAOChannel/SO	
AnalogOutput_2	AnalogOutput_2_AOUTPUT_AOS	\$AnalogOutput	AOChannel	SAOChannel/SO	
AnalogOutput_3	AnalogOutput_3_AOUTPUT_AOS	\$AnalogOutput	AOChannel	SAOChannel/SO	
AnalogOutput_4	AnalogOutput_4_AOUTPUT_AOS	\$AnalogOutput	AOChannel	SAOChannel/SO	

Exporting Hardware Mapping Data

To export the data shown in the **Hardware Mapping Editor**, proceed as follows.

Step	Action
1	In the Hardware Mapping Editor , select the data that you want to export by using the filters, page 120.
2	Click the export button, page 748. Result: The Save As dialog box opens.
3	Enter a file name, select a location, and click Save . Result: The two export files are created. NOTE: The export files do not contain the Control project and the system identifiers. If required, include this information in the file name.

Editing the Hardware Mapping Export File

Overview

In this second step, the objective is to edit the exported hardware mapping data file (**_HWMMapping.csv*) so that it contains the hardware mappings that you want to create or modify for a Control executable.

You can use the supporting file (**_Pending.csv*) which contains the project facet interfaces that are available for mapping.

It is a good practice to use a spreadsheet editor to facilitate the editing of data contained in the export files.

NOTE: Although you can delete hardware mappings by using the import functionality, you can delete them directly in the **Hardware Mapping Editor** by using the **Unmap** command, page 752.

Good Practices

- Before modifying hardware mappings, back up all the existing hardware mappings by exporting them. If needed, it lets you restore the original configuration by importing the file.
- Edit only an up-to-date export file to avoid data conflicts.
- Verify that the **list separator format**, page 236 is set to comma on the computer if you are using a spreadsheet editor.
- When you use data from the supporting file to create hardware mappings, cut the data rather than copying it to help avoid mapping the same project facet interface several times.

The other way around, add to the supporting file project facet interfaces that you are unmapping.

Compatibility With Hardware Mapping Files of Earlier Versions

You can use hardware mapping data files (*topologyexport.csv* and supporting file *projectexport.csv*) that have been exported by using the system server console of earlier versions if you edit the column headers to match the existing format.

The following table describes which column headers of the *topologyexport.csv* file need to be updated and their new value (not case-sensitive).

Column ⁽¹⁾	Header	Change to
A	Instance	HWInstance
B	Template	HWTemplate
C	HW Mapping IF	HWMappingInterface
D	HW Mapping Type	HWInterfaceType
E	Instance	AppInstance
F	Project Facet	AppFacet
G	Facet Type	AppTemplate
H	HW Interface	AppMappingInterface
I	HW Mapping Type	AppInterfaceType
(1) When you open the file by using Microsoft® Excel®		

Editing the Hardware Mapping Data File

Edit only rows that correspond to topological interfaces whose hardware mapping you want to create, modify, or delete. Refer also to the [examples](#), page 759.

Entries that exist in the **Hardware Mapping Editor** but not in the hardware mapping data file (for example, because a filter was applied before exporting or because they have been deleted from the file) are not modified.

The tables describe the result of various editing actions in the hardware mapping data file, page 755 (**_HWMMapping.csv*) once you import the file.

Editing to create, modify, and delete hardware mappings:

Edited data	Action	Result after import
Rows of columns E to I ⁽¹⁾	New, compatible ⁽²⁾ project facet interface data is added by cutting and pasting it from the <i>*_Pending.csv</i> file.	The corresponding hardware mapping is created. The hardware mapping is valid if the values of the two *InterfaceType columns match (same interface type and compatible roles).
	Existing project facet interface data is modified appropriately. For example: <ul style="list-style-type: none"> By mapping another project facet interface to the topological interface (the data is cut and pasted from the <i>*_Pending.csv</i> file). By swapping two mapped project facet interfaces. 	The corresponding hardware mapping is modified.
	Existing project facet interface data is deleted.	The corresponding hardware mapping is deleted.
(1) When you open the file by using Microsoft® Excel®		

Other editing:

Edited data	Action	Result after import
Rows of columns A to D ⁽¹⁾	Existing topology interface data is deleted.	If columns E to I ⁽¹⁾ for the same row: <ul style="list-style-type: none"> Are empty, no impact. This lets you remove from the file rows that you do not need to work with. Contain data, an invalid entry is reported.
The first row	The first row is deleted.	You cannot import the file.
	A header of columns A to E ⁽¹⁾ is deleted or modified.	
Other rows	A row is not edited.	No impact given the same row is still present in the Hardware Mapping Editor .
	An entire row is deleted.	No impact. NOTE: This lets you remove from the file rows that you do not need to work with.
	A row containing a string is added. Enter the string in column A ⁽¹⁾ and start with a semi-colon (for example, ;My comment).	No impact. NOTE: This lets you add comments to the file.
	An empty row is added.	No impact.
	The order of complete rows is changed.	No impact.
	The topology or project facet interface data (or both) of a row is duplicated.	You cannot import the original hardware mapping nor its copy.
Any columns	A column is added in first position (A ⁽¹⁾) or between columns A to E ⁽¹⁾ .	You cannot import the file.
	One or more columns are added after the last column (E ⁽¹⁾).	No impact. NOTE: This lets you add comments at the end of a row.
	The order of columns A to E ⁽¹⁾ is modified.	No impact.
(1) When you open the file by using Microsoft® Excel®		

Example of Editing to Create Hardware Mappings

The following example shows how to create hardware mappings with two channels of an analog output module by pasting the corresponding project facet interface data of the supporting file (shown outlined) to the hardware mapping data file.

	A	B	C	D	E	F	G	H	I	J	K
1	HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType		
2	Controller_1_0.PrLocal.0.D.0.0.8.MEP586040	SM580CPUEnCRP	MBTCEPM580.EMPorChM	SMEthernetPortCh/HO							
3	Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_01.MPDInput	SDIChannel/HO	DigitalInput_1	DigitalInput_1_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO		
4	Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_02.MPDInput	SDIChannel/HO	DigitalInput_2	DigitalInput_2_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO		
5	Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_03.MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO		
6	Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_04.MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO		
7	Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPMODr.ModuleStatus	SModuleDiagMapping/HO							
8	Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_01.MPAOutput	SAOChannel/HO							
9	Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_02.MPAOutput	SAOChannel/HO							
10	Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_03.MPAOutput	SAOChannel/HO							
11	Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_04.MPAOutput	SAOChannel/HO							
12	Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPMODr.ModuleStatus	SModuleDiagMapping/HO							

The following figure shows the result in the topological devices pane of the **Hardware Mapping Editor**. Also, the corresponding analog output project facet interfaces have been removed from the project facet pane of the editor (not shown).

System_1 : Project "ControlProject_1.ControlExecutable_1" Hardware Mapping Editor

Available for mapping	HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType
Controller_1_0.PrLocal.0.D.0.0.8.MEP586040	SM580CPUEnCRP	MBTCEPM580.EMPorChM	SMEthernetPortCh/HO						
Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_01.MPDInput	SDIChannel/HO	DigitalInput_1	DigitalInput_1_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO	
Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_02.MPDInput	SDIChannel/HO	DigitalInput_2	DigitalInput_2_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO	
Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_03.MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO	
Controller_1_0.PrLocal.0.D.0.0.2.BMXDD11604	SMPDInput16	MPDInputCh_04.MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO	
Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_01.MPAOutput	SAOChannel/HO	AnalogOutput_1	AnalogOutput_1_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO	
Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_02.MPAOutput	SAOChannel/HO	AnalogOutput_2	AnalogOutput_2_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO	
Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_03.MPAOutput	SAOChannel/HO						
Controller_1_0.PrLocal.0.D.0.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_04.MPAOutput	SAOChannel/HO						

Example of Editing to Delete Hardware Mappings

Continuing with the previous example, the figure shows how to delete hardware mappings with two channels of a digital input module by deleting the corresponding project facet interface data (shown outlined) from the hardware mapping data file.

	A	B	C	D	E	F	G	H	I
1	HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType
2	Controller_1_0.PrLocal.0.D.0.8.MEP586040	SM580CPUEnCRP	MBTCEPM580.EMPorChM	SMEthernetPortCh/HO					
3	Controller_1_0.PrLocal.0.D.0.2.BMXDD11604	SMPDInput16	MPDInputCh_01.MPDInput	SDIChannel/HO	DigitalInput_1	DigitalInput_1_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
4	Controller_1_0.PrLocal.0.D.0.2.BMXDD11604	SMPDInput16	MPDInputCh_02.MPDInput	SDIChannel/HO	DigitalInput_2	DigitalInput_2_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
5	Controller_1_0.PrLocal.0.D.0.2.BMXDD11604	SMPDInput16	MPDInputCh_03.MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
6	Controller_1_0.PrLocal.0.D.0.2.BMXDD11604	SMPDInput16	MPDInputCh_04.MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
7	Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_01.MPAOutput	SAOChannel/HO	AnalogOutput_1	AnalogOutput_1_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO
8	Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_02.MPAOutput	SAOChannel/HO	AnalogOutput_2	AnalogOutput_2_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO
9	Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_03.MPAOutput	SAOChannel/HO					
10	Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_04.MPAOutput	SAOChannel/HO					
11	Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPMODr.ModuleStatus	SModuleDiagMapping/HO					
12	Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPMODr.ModuleStatus	SModuleDiagMapping/HO					

NOTE: It is a good practice to cut the project facet interface data and paste it in the supporting file to keep track of project facet interfaces that are available for mapping.

The following figure shows the result in the topological devices pane of the **Hardware Mapping Editor**. Also, the corresponding project facet interfaces have been added to the project facet pane of the editor (not shown).

System_1 : Project "ControlProject_1.ControlExecutable_1" Hardware Mapping Editor

Available for mapping	HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType
Controller_1_0.PrLocal.0.D.0.8.MEP586040	SM580CPUEnCRP	MBTCEPM580.EMPorChM	SMEthernetPortCh/HO						
Controller_1_0.PrLocal.0.D.0.2.BMXDD11604	SMPDInput16	MPDInputCh_01.MPDInput	SDIChannel/HO						
Controller_1_0.PrLocal.0.D.0.2.BMXDD11604	SMPDInput16	MPDInputCh_02.MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO	
Controller_1_0.PrLocal.0.D.0.2.BMXDD11604	SMPDInput16	MPDInputCh_03.MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO	
Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_01.MPAOutput	SAOChannel/HO	AnalogOutput_1	AnalogOutput_1_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO	
Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_02.MPAOutput	SAOChannel/HO	AnalogOutput_2	AnalogOutput_2_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO	
Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_03.MPAOutput	SAOChannel/HO						
Controller_1_0.PrLocal.0.D.0.3.BMXAM00410	SMPAOutput4	MPAOutputCh_04.MPAOutput	SAOChannel/HO						

Example of Editing to Modify Hardware Mappings

Continuing with the previous example, the figure shows how to move an existing hardware mapping with the channel of an analog output module (*Ch_02*) to another channel (*Ch_01*). This is achieved by moving the corresponding project facet interface data of *AnalogOutput_2* to the other topological interface, thus overwriting the original interface data of *AnalogOutput_1*.

J	A	B	C	D	E	F	G	H	I
1	HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType
2	Controller_1 0 PrtLocal 0 D 0 R 0 BMEPS86040	SM580CPUethCRP	MBTCPethM580.EMPotChM	SMEthernetPortCh/HO					
3	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_01.MPDInput	SDIChannel/HO					
4	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_02.MPDInput	SDIChannel/HO					
5	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_03.MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
6	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_04.MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
7	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPMODer.ModuleStatus	SMModuleDiagMapping/HO					
8	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_01.MPAOutput	SAOChannel/HO	AnalogOutput_1	AnalogOutput_1_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO
9	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_02.MPAOutput	SAOChannel/HO	AnalogOutput_2	AnalogOutput_2_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO
10	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_03.MPAOutput	SAOChannel/HO					
11	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_04.MPAOutput	SAOChannel/HO					

NOTE: Before overwriting the project facet interface data of *AnalogOutput_1*, it is a good practice to cut the data and paste it in the supporting file to keep track of project facet interfaces that are available for mapping.

The following figure shows the result in the topological devices pane of the **Hardware Mapping Editor**. The interface of *AnalogOutput_2* is now mapped to *Ch_01*. The project facet interface of *AnalogOutput_1* has been unmapped and added to the project facet pane of the editor (not shown).

System_1: Project 'ControlProject_1.ControlExecutable_1' Hardware Mapping Editor

Available for mapping	HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType
	Controller_1 0 PrtLocal 0 D 0 R 0 BMEPS86040	SM580CPUethCRP	MBTCPethM580.EMPotChM	SMEthernetPortCh/HO					
	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_01.MPDInput	SDIChannel/HO					
	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_02.MPDInput	SDIChannel/HO					
	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_03.MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_04.MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO
	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_01.MPAOutput	SAOChannel/HO	AnalogOutput_2	AnalogOutput_2_AOUTPUT_AOS	SAAnalogOutput	AOChannel	SAOChannel/SO
	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_02.MPAOutput	SAOChannel/HO					
	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_03.MPAOutput	SAOChannel/HO					
	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_04.MPAOutput	SAOChannel/HO					

Example of Editing to Add Comments to the Export File

The following figure illustrates the various ways to add comments (shown in bold) to the hardware mapping data file (**_HWMMapping.csv*) so that they are not detected as invalid entries. Comments are not imported.

J	A	B	C	D	E	F	G	H	I	J	K
1	HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface	AppInterfaceType	Comments in columns after I	Comment header
2	You can add comments as rows - 1										
4	Controller_1 0 PrtLocal 0 D 0 R 0 BMEPS86040	SM580CPUethCRP	MBTCPethM580.EMPotChM	SMEthernetPortCh/HO							
5	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_01.MPDInput	SDIChannel/HO	DigitalInput_1	DigitalInput_1_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO	Comment 1	
6	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_02.MPDInput	SDIChannel/HO	DigitalInput_2	DigitalInput_2_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO		Comment 2
7	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_03.MPDInput	SDIChannel/HO	DigitalInput_3	DigitalInput_3_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO		
8	Controller_1 0 PrtLocal 0 D 0 R 02 BMXDDI1604	SMPDInput16	MPDInputCh_04.MPDInput	SDIChannel/HO	DigitalInput_4	DigitalInput_4_DINPUT_DIS	SDigitalInput	DIChannel	SDIChannel/SO		
9	You can add comments as rows - 2										
10	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_01.MPAOutput	SAOChannel/HO							
11	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_02.MPAOutput	SAOChannel/HO							
12	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_03.MPAOutput	SAOChannel/HO							
13	Controller_1 0 PrtLocal 0 D 0 R 03 BMXAM00410	SMPAOutput4	MPAOutputCh_04.MPAOutput	SAOChannel/HO							

NOTE: The CSV format does not allow saving special formatting.

Importing Hardware Mappings

Overview

In this third step, the data contained in the hardware mapping data file (**_HWMMapping.csv*) is imported into the topological devices pane **Hardware Mapping Editor** that is open for the Control executable.

Each row in the hardware mapping data file is compared to the same row in the **Hardware Mapping Editor** and if different, triggers an action that corresponds to the editing, page 758 that you have performed. Rows in the editor that are not present in the hardware mapping data file remain unchanged.

You can import a file into the same or a different system, Control, or executable as long as the prerequisites are fulfilled.

Prerequisites

To be able to import hardware mapping data, the following conditions must be fulfilled:

- You must have performed the service mapping, page 723 for the Control executable.
- Hardware mapping data files that were created by using earlier versions must be compatible, page 757.
- Both the mapping interfaces of the topological entity and of the application objects, page 755 that are in the hardware mapping data file must exist when you open the **Hardware Mapping Editor** for the Control executable. Additional interfaces of both types can appear in the editor and they have no effect on the import.

Invalid Hardware Mapping Log File

When you attempt to import a hardware mapping data file and invalid mappings are detected, a log file is created with these mappings.

- The file is located in the folder from where you have imported the hardware mapping data file.
- It has the same name as the imported file but with the *_log* suffix and .txt file extension.

Thereafter, each time you try to import the same hardware mapping data file (same file name), invalid mappings that are detected are added to the log file.

Importing Hardware Mapping Data

To import hardware mapping data in the **Hardware Mapping Editor**, proceed as follows.

Step	Action
1	In the Hardware Mapping Editor , click the import button, page 748. Result: The Open dialog box opens.
2	Select the file (<i>*_HWMMapping.csv</i>) that contains the hardware mapping data and click Open . Result: The Import Mappings dialog box opens.
3	Click Yes . Result: If the hardware mapping data is valid, the corresponding hardware mappings are performed in the Hardware Mapping Editor ; otherwise, if invalid hardware mappings, page 758 are detected in the imported file, a dialog box opens, which contains information about the invalid entries. Proceed to step 4.

Step	Action
	NOTE: Once the import operation is completed, you can see a summary in the notification pane.
4	<p>Click:</p> <ul style="list-style-type: none">• Yes to import only the valid data; invalid entries will be skipped.• No to cancel the entire import operation. <p>NOTE: If you proceed, you can see a summary in the notification pane once the operation is completed.</p>
5	If required, correct the invalid entries in the hardware mapping file or make the necessary adjustments in the Control project and import the file again.

Supervision Project Mapping Stage

Overview

This section describes how to use the **Project Explorer** to map the Supervision project of the system.

You must have completed [Generation](#), page 396 and the [Configuration](#), page 512 stages of the Supervision project to proceed with the mapping.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Creating Supervision Project Executables

Overview

The **Executables** node gives you access to the service and communication mapping functionality of the Supervision Participant project.

As such, it allows you to associate:

- The Supervision infrastructure (I/O, alarm, trend, and/or report servers, and/or runtime clients) that is defined in the Supervision Participant project to workstations of the topology modeling operation servers and operator stations.
- The I/O device to a controller and its communication module and/or to compatible Modbus TCP devices and their Modbus TCP server service by using the OFSOPC and/or OPCUA communication protocol.

You can create several executables for a Supervision Participant project, associating the Supervision infrastructure to various topological entities, and its I/O device to various controllers and Modbus TCP devices.

An executable is associated to an **Execution Domain**, which serves as a filter to select applicable services, for example, to define the boundaries for runtime navigation services.

Creating Executables

To create an executable, in the **Supervision Project Browser**, right-click the **Executables** node of the project and select **Create Executable**.

Result: A Supervision executable is created under the **Executables** node.

Executable Actions

The table describes the executable context menu commands.

Command	Description
Manage	<p>Opens the Manage window in the work area of the Project Explorer, which lets you perform the following actions:</p> <ul style="list-style-type: none"> • Service mapping, page 767: Associates the Supervision servers and clients to one or more workstations of the system topology. • View and edit the properties, page 765 of the executable. • Communication mapping, page 769: Maps the Supervision I/O device to: <ul style="list-style-type: none"> ◦ A controller and its communication module. ◦ Compatible Modbus TCP devices and their Modbus TCP server service.
Build , page 788	<p>Starts an automated process to integrate changes in the built Supervision Participant project by performing an incremental build. The command applies only for changes that you have made while editing Supervision pages (Edit command).</p> <p>This command becomes available once you have completed the service and communication mapping.</p> <p>NOTE: If the Supervision Participant project is not built yet, you can use this command in place of Build All indifferently.</p>
Build All	<p>Starts an automated process to create, page 788 the built Supervision Participant project the first time or to recreate it to integrate certain changes, page 792 that you have made to the Supervision Participant project.</p> <p>This command becomes available once you have generated the Supervision project and completed the service and communication mapping.</p>
Generate and Build , page 399	<p>Starts a generation of the Supervision Participant project followed by a build of the project executable.</p> <p>If you have not built the executable yet, the build process that is executed is the same as when you select the Build All command.</p>

Command	Description
	<p>If the generation process does not complete successfully, no build is performed and information is displayed in the notification panel.</p> <p>NOTE: After selecting the command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task.</p>
Deploy Built Project , page 841	Allows you to deploy the built Supervision project to the workstations that are mapped to the executable.
Deploy Changes , page 851	The command is available only if you have already deployed the built Supervision project to at least one workstation that is mapped to the executable and if the number and type of changes are supported.
Run Supervision Client	<p>Launches the Supervision runtime locally with the corresponding Supervision project after you confirm the command.</p> <p>The command is available only if the following conditions are satisfied:</p> <ul style="list-style-type: none"> The executable is mapped to a workstation whose NIC is configured with the IP address of the local computer. The build state of the executable is Built or Out Of Date. <p>For the Supervision project to launch in runtime (see <i>EcoStruxure Process Expert, Supervision Participant Services, User Guide</i>), some actions need to be performed first, such as deploying the executable.</p> <p>If the runtime is already launched, you can restart it by using the command.</p> <p>The runtime is not launched if the Supervision Participant is being used by EcoStruxure Process Expert (for example, if a Supervision project is being refined or its executable built).</p>
Download Projects , page 791	<p>Opens the Download Projects tab, which allows you to select Supervision project files individually to save them on the computer, such as:</p> <ul style="list-style-type: none"> The master project (.ctz) with the name <i>ProjectIdentifier_ExecutableIdentifier</i>, which was created by the last successful build process. Included projects (.ctz) that are related to animated graphics assigned to Supervision pages and those that you have added manually, page 510. OFS/OPC UA configuration files (.xml).
Delete	Deletes the executable including the associated configurations.
Rename	Lets you to enter a new identifier for the executable.
Properties	Opens the Properties tab in which you can view and edit properties of the executable.

Executable Properties

To open the properties of the executable, double-click it.

You can view and/or edit the following items.

Property	Description
Identifier	The identifier must be unique within a Supervision project.
Description	<p>Optional.</p> <p>You can enter a description for the Executable with free form text.</p>
Execution Domain	<p>Optional.</p> <p>String.</p> <p>Default value: Blank.</p> <p>You can use runtime navigation services (see EcoStruxure™ Process Expert, Runtime Navigation Services , User Guide) only if the executable of the Control Participant project and the executable of the Supervision Participant project have the same execution domain value.</p> <p>You can assign a logical execution domain to the executable with free form text.</p> <p>Values that you enter remain in the menu and you can select them from executables of other Control and Supervision Participant projects.</p>

Property	Description
	You can modify the Execution Domain parameter at any time during the engineering life cycle. NOTE: Leaving the Execution Domain parameter empty is considered as a value.
Build State	Indicates the status, page 790 of the executable. Default value: Not Built
Built On	Date and time when you last built the executable successfully by using a build command.

Assigning an Execution Domain

To assign an execution domain to an executable, proceed as follows.

Step	Action
1	Double-click the executable to open its properties.
2	Select a value from the Execution Domain menu or enter a new value.

NOTE: To remove or change the associated execution domain, select the empty row at the top of the menu.

Mapping Services

Overview

The **Service Mapping Editor** opens within the **Manage** window when you double-click the Supervision executable, page 764.

The service mapping defines the workstations that model the operation server and operator station computers to which you deploy, page 841 the Supervision project.

Before you proceed with the service mapping, create the necessary workstations in the topology of the system and configure them with the services as needed:

- A Supervision service, page 598.
- An **OFS** service, page 598.

You must map the existing services of the Supervision Participant project to workstations in order to complete the build stage, page 788.

NOTE: Install OPC Factory Server or OPC UA Server Expert on the workstation that you map to the *IOServer* service.

Mapping Workstations Configured with Several NICs

The service mapping does not let you select a specific NIC for each workstation.

The IP address of each NIC is created as a network address, page 315 in the Supervision Participant project and associated to the I/O server.

For details on the behavior when deploying to a workstation that is configured with more than one NIC, refer to the topic describing the target computer setup, page 842.

Changing the Service Mapping of Deployed Executables

If you have already deployed a Supervision executable to one or more workstations of the system, when you change the service mapping of this executable, the related entry is cleared in the **Last Deployed On** field, page 847 of the workstations that you have unmapped. As a result, if you recreate this mapping right away and deploy, it is handled like a first deployment.

NOTE: The service mapping of a workstation is unmapped if you delete its Supervision service. It is unmapped if you delete its **OFS** service, page 598 and the workstation was mapped to the *IOServer* service of the Supervision project.

Mapping Services

To map a Supervision server or client to a workstation, proceed as follows.

Step	Action
1	Double-click the executable of the Supervision project that you want to map and select Manage . Result: The Service Mapping Editor opens.
2	Click the Engine menu next to the server or client that you want to map. Result: The menu displays the identifiers of compatible workstations, page 767.
3	Select a workstation. Result: The service is mapped to the selected workstation. NOTE: Once you have mapped the I/O server, the Communication Mapping tab is shown at the bottom of the window.

NOTE: To remove a mapping, in the **Service Mapping Editor**, right-click a service and select **Unmap**.

Mapping Communication

Overview

To open the **Communication Mapping Editor**, click the **Communication Mapping** tab within the **Manage** window, page 764. The tab appears once you have performed the service mapping, page 767 with an I/O server.

The communication mapping defines from which topological entity I/O devices acquire data during runtime by using the selected driver, page 312.

The editor lets you map the I/O devices of the Supervision project to:

- A controller of the system and its communication modules. This includes modules that embed an OPC UA server.
- Modbus TCP devices of the **EPE Managed Devices** product range that feature the Modbus TCP server service (for example, Altivar drives, power meters, STB islands, or PRMs).
- Modbus serial devices by using a topological Modbus TCP device acting as gateway (**EPE Managed Devices** product range).
- The Control service of a workstation with a Controller simulator to which a Control project has been deployed.

For each I/O server that you map, a protocol-dependent OFS/OPC UA configuration file, page 788 is created. Exceptions apply.

NOTE: Changing the protocol of a tag container, page 354 whose associated I/O device is already mapped to a topological entity removes the mapping.

Mapping Modbus Devices

Before you can map a Modbus TCP or Modbus serial device of the **EPE Managed Devices** product range, to monitor data directly from the Supervision infrastructure that is defined in the Supervision project, you must perform some additional steps.

For details, refer to the topic describing how to monitor Modbus device data directly from Supervision, page 984.

Mapping Modules Embedding an OPC UA Server

If **Protocol** of the tag container that is associated to the I/O device is set to *OPCUA Embedded*, **Data Server (Address B)** is disabled. You can map only one module per I/O device.

If modules embedding the OPC UA server are mounted in both local racks of a redundant configuration, either module can be selected under **Data Server (Address A)**.

To also map the module embedding an OPC UA server located in rack B, create a second I/O device configured as standby, page 309 and associate it to the same tag container.

Mapping BMEXNOR0200 Modules

BMEXNOR0200 modules automatically appear in **Data Server (Address A)** and/or **Data Server (Address B)** when you select the controller containing the module in the **Topological Entity** column of the **Communication Mapping Editor**.

Mapping Redundant Communication Ports

For controllers and workstations, you can map up to two communication ports or NICs to a same I/O device if the following conditions are satisfied:

- The communication ports/NICs belong to the same topological entity.
- **Direct TCP/IP Addressing Mode** of the tag container, page 354 that is associated to the I/O device is set to *Unallocated/Allocated Data, PAC Modicon (/U)*.

NOTE: If **Protocol** of the tag container that is associated to the I/O device is set to *OPCUA Embedded*, communication port redundancy for the same I/O device is not possible.

Communication Mapping Tab

The following figure shows an example of the **Communication Mapping Editor** in which one controller and two Modbus TCP devices are mapped to I/O devices through their communication modules.

Assignment(Supervis... x Manage(Executable_1... x				
System_3 : Project 'Supervision_1' : Executables > Executable_1 : Communication Mapping Editor				
IO Device	Topological Entity	Data server (Address A)	Data server (Address B - Optional)	Modbus Unit ID
IODevice_1	Controller_1	Controller_1 0:PrLocal 0:D 0:R 0:BMEPS86040	Controller_1 0:PrLocal 0:D 0:R 02:BMENOC0321	
IODevice_3	EGenericDeviceHW_1	EGenericDeviceHW_1:EIOSSERVER	Not Assigned	10
IODevice_2	EATV6xHW_1	EATV6xHW_1:EIOSSERVER	Not Assigned	255

Header	Description
IO Device	I/O devices that you have created in the cluster and which is associated to a tag container.
Topological Entity	<p>Topological entities that have at least one communication module (data server) connected to a logical network. These can be controllers, workstations, Modbus TCP devices⁽¹⁾, and generic Modbus TCP devices⁽¹⁾ acting as gateways for Modbus serial devices.</p> <p>The selected entity becomes the data source.</p> <p>NOTE: Restrictions apply if the protocol of the tag container that is associated to the I/O device has the following value:</p> <ul style="list-style-type: none"> <i>OPCUA</i>: Controllers of the Quantum platform are not shown. <i>OPCUA Embedded</i>: Only controllers that have a module embedding an OPC UA server and that is connected to a logical network are shown.
Data Server (Address A)	<p>Communication modules of the selected topological entity that are connected to a logical network.</p> <p>(The IP address of the module is set as the primary address in the OFS/ OPC UA configuration file.)</p> <ul style="list-style-type: none"> For Modbus I/O devices⁽¹⁾, it is the Modbus TCP server service of the topological entity. For a workstation, if more than one Control Expert service exists, it is the NIC followed by the controller simulator port that is configured for the service. Otherwise, only the NIC appears. <p>NOTE: If the protocol of the tag container that is associated to the I/O device is set to <i>OPCUA Embedded</i>, only modules that embed an OPC UA server are shown.</p>
Data Server (Address B)	<p>Optional. The menu is available only if the topological address A has been mapped.</p> <p>Communication modules of the selected topological entity that are connected to a logical network.</p> <p>The IP address of the module is set as the secondary address in the OFS/ OPC UA configuration file.</p> <p>NOTE: The menu is disabled in either case:</p> <ul style="list-style-type: none"> The topological entity is a Modbus TCP device⁽¹⁾. Protocol of the tag container that is associated to the I/O device is set to <i>OPCUA Embedded</i>.
Modbus Unit ID	<ul style="list-style-type: none"> For Modbus TCP devices⁽¹⁾, it is the ID parameter of the topological entity. The value corresponds to the default value of the template or to the value that you configured (if available) before selecting the device and its data server. You can change the value for the device. The change is not propagated to the topological entity. Changing the value in the topological entity is not propagated to devices that are already mapped. For Modbus serial devices⁽¹⁾, if you already configured an application template of the device (for example, an instance of <i>\$ATV71MB</i>), enter the ModbusAddress parameter value of the instance. Otherwise, enter the unique serial slave ID of the device. <p>Valid range: 0 to 255</p> <p>Refer to the documentation of the device for information on the parameter configuration.</p> <p>NOTE: The uniqueness of entries is not verified.</p> <p>NOTE: The parameter is disabled for workstations and controllers.</p>
(1) Devices of the EPE Managed Devices product range.	

Mapping Workstations

If you map an I/O device to the Control service of a workstation, you can simulate runtime services.

The workstation is available in the **Communication Mapping Editor** only if its NIC is connected to an logical network and if you have created at least one Control Expert service, page 598 in the workstation.

NOTE: Map, page 772 the Control executable of the corresponding Control Participant project to the Control service of the workstation that uses the same port (service mapping).

Mapping Communication

To map the I/O device to the communication module/card of a topological entity, proceed as follows.

Step	Action
1	Select the Communication Mapping tab within the Manage window.
2	Click the Topological Entity menu next to the I/O device you want to map. Result: The menu displays the identifiers of the compatible entities of the system.
3	Select an entity.
4	Click the Data Server (Address A) menu. Result: The menu displays the identifiers of the available communication modules/cards of the selected entity.
5	Select a communication module/card.
6	If needed, click the Data Server (Address B) menu and select a second communication module/card of the same entity.

NOTE: To remove the entire communication mapping for an I/O device, right-click a communication module/card in either **Data Server** column and select **Unmap**.

To remove the mapping of one data server only, select **Not Assigned** from the menu.

Build Stage

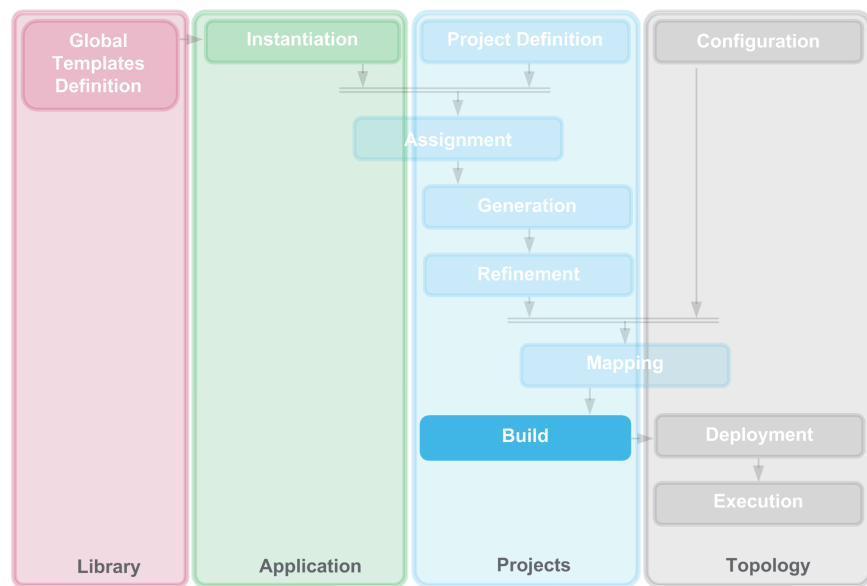
What's in This Chapter

Control Project Build Stage	774
Supervision Participant Project Build Stage	787

Overview

This chapter describes how to create the built Control and Supervision Participant projects, which are the executables and project files that you deploy to controllers and/or workstations of the system.

The following figure shows the position of the **Build** stage within the system engineering life cycle.



Refer to the Build stage, page 65 for a description of the purpose of this stage.

Control Project Build Stage

Overview

This section describes how to use the **Project Explorer** to create the built Control Participant project for the first time and, in a second step, to integrate changes that you have made to the Control Participant project and to the topology of the system.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Building the Logical Control Participant Project

Overview

In this automated process, EcoStruxure Process Expert creates a unique built Control Participant project, which merges:

- The logical Control Participant project information, including changes made during refinement.
- The mapping information, which is associated to the Control executable of the logical project:
 - Services
 - Hardware
 - Communication (if applicable)
- The topological information, which is associated to the mapped services

The resulting program file (.stu) is stored in the [content repository](#), [page 1121](#).

The build process also:

- Creates the necessary DDDT variables with the number of input and output words that you have configured to allocate memory.
- Creates the I/O scanner lines for the communication channels that exist in the communication mapping given that sufficient memory is available, [page 595](#).
- Defines the value of REF_TO data types (see EcoStruxure™ Process Expert, Control Participant Services, User Guide):
 - As *REF(%MWx)* or *REF(%CH<@mod>.<c>)* (IODDT) for direct addressing data instances (HAL2.0 variables) that are used in hardware mapping.
 - As *REF(DDDT)* for variables that are used in communication mapping.
 - As *REF(PES_CONST_TRUE)* for variables that need to be initialized as *TRUE*.
- Introduces the REF_TO ANY_BOOL data type because of the legacy feature to convert BOOL to EBOOL, and the other way around becoming obsolete. If you have updated existing Schneider Electric HAL templates in systems with HAL2.0 templates (see *EcoStruxure Process Expert, Control Participant Services, User Guide*), both data types are converted to REF_TO ANY_BOOL.

NOTE: The conversion does not apply to templates created by users except if they reference Schneider Electric HAL templates (for example, *\$DISignal_UL*).

To perform a subsequent build or if the **Build** command has become unavailable, refer to the topic [Changing the Built Control Participant project](#), [page 781](#).

NOTE: You may need to update the firmware version of topological entities to support the ANY_BOOL data type during build. For information on compatible firmware versions, refer to the platform release notes.

Properties of M580 Controllers Mapped to the Executable

If you modified a property of an M580 controller that is mapped to an executable but did not apply the change to the controller configuration, a message is displayed and you cannot build the executable. You must [apply changes](#), [page 592](#) in the **Topology Explorer** before using a build command.

Firmware and Web Diagnostic/Data Storage Passwords

If you receive a message about the **Firmware** and/or **Web Diagnostic/Data Storage** passwords not being set or having default values when you build an executable, set the passwords in the Participant as indicated in the following table.

Topological entity mapped to the executable	Where to set the passwords
Workstation	Set both passwords in the project Properties of the Control project by refining it, page 447 from the Project Explorer .
M580 controller with firmware version 4.00 and later.	Set the Firmware password in the project Properties of the controller by configuring it from the Topology Explorer .

Device Type Manager and Device DDT Variables for Modbus TCP Devices

During build, EcoStruxure Process Expert adds a generic **Modbus Device** device type manager (DTM) for each Schneider Electric **Ethernet Modbus TCP device**, page 603 that you have configured in the topology of the system and which communicates with a controller of the M580 platform or by using a NOC communication module. You do not need to add it.

Modbus TCP devices can be:

- An STB Island
- A PRM
- A Modbus TCP I/O device

The DTM name is composed of four elements. It has the following syntax.

<I/O device name><Counter>_<Client communication module><Position>

Where:

- <I/O device name> is the identifier of the topological entity, page 562. The identifier is truncated not to exceed 10 characters.
- <Counter> is an incremental 1 or 2-digit number generated automatically so that truncated device names are unique.
- <Client communication module> is the commercial reference of the communication module of the client controller that is used to communicate with the device. The reference is truncated to not exceed 12 characters.
- <Position> is the position of the communication module on the rack.

For example, EATV71HW_11_BMENOC030122 (1 in 11th position from the left is the counter and 2 in last position is the position of the module. EATV71HW_1 is the I/O device name. BMENOC03012 is the commercial reference of the communication module).

The Control Participant creates the corresponding DDDT variable.

Device Type Manager and Device DDT Variables for Peer to Peer Communication

During build, EcoStruxure Process Expert adds a generic **Modbus Device** device type manager (DTM) in the consumer (client) Control project for each communication module of a server counterpart with which a peer to peer communication channel exists. The DTM is added if the controller acting as client is of the M580 platform or if communication client-side is implemented by using a NOC communication module.

The DTM name is composed of four elements. It has the following syntax.

<Server entity><Counter>_<Server communication module><Position>

Where:

- <Server entity> is the identifier, page 592 of the topological entity acting as server, page 726. The identifier is truncated to not exceed 10 characters.
- <Counter> is an incremental 1 or 2-digit number generated automatically so that truncated server entity names are unique.

- <Server communication module> is the commercial reference of the communication module of the controller acting as server. The reference is truncated to not exceed 12 characters.
- <Position> is the position of the communication module on the rack.

For example, `Controller1_140NOE771012` (1 in 11th position from the left is the counter and 2 in last position is the position of the module. `Controller_1` is the server entity name, which has been truncated. 140NOE77101 is the commercial reference of the communication module server side).

The Control Participant creates the corresponding DDDT variable.

Device DDT Variables Created in the Logical Control Project

If you have created a device DDT variable, page 455 in the logical Control project and this variable does not exist in the controller that is mapped to the executable (for example, because you have removed the module to which it is associated), the resulting built Control project contains it in unmapped (unlocked) state.

If a device DDT variable with the same name exists in both the logical Control project and the controller that is mapped to the executable but with a different data type or type version, the resulting built Control project contains the variable of either type based on the following rule.

DDDT variable in the controller	DDDT variable in the logical Control project	Result in the built Control project
Mapped	Unmapped	Variable of the same type as in the controller mapped to the executable.
Unmapped	Unmapped	Variable of the same type as in the logical Control project.

Validity Indicators

If a validity icon, page 336 is displayed, you may not be able to build the project successfully. Some exceptions apply, page 401.

In this case, Refine the Control Participant project, page 446 to fix the issue. The Control Participant features tools that help you identify validity issues.

When you save the Control Participant project after refining, page 449 it and project analysis is enabled, page 272, the project is analyzed. The status of the Control project is indicated by a validity icon.

Building the Control Project for the First Time

Use the **Build** or **Build All** command indifferently to create the built Control Participant project for the first time.

NOTE: If you have not built the Control Participant project yet, you can also use the **Generate and Build** command. For information on the generation process, refer to the topic describing the generation of Participant projects, page 398.

Control Participant Project Settings

When you build a logical Control Participant project by using the **Build** or **Build All** command, EcoStruxure Process Expert may override some project settings (see *EcoStruxure Process Expert, Control Participant Services, User Guide*) that you have set during controller configuration or project refinement.

To view the **Project Settings** of the built Control Participant project, use the **Open Built Project** command, page 780.

NOTE: This applies also when you are using the **Generate and Build** command to build a Participant project executable.

Allocation of Memory in I/O Scanner Lines

The build process allocates the memory in the following order: First write, then read.

Read and write addresses are created contiguously for clients of the M580, M340, and Quantum platforms.

The read/write memory allocation varies depending on the platform.

Platform	Memory allocation process
Quantum	<p>The write address should be the end of the read address of the previous channel. However, for example, if 2 channels exist for entity A, and 1 channel exists for entity B, then the memory allocation is the following:</p> <ol style="list-style-type: none"> 1. Write memory allocation for any existing channels of entity A. 2. Read memory allocation for any existing channels of entity A. 3. Write memory allocation for the channel of entity B. 4. Read memory allocation for the channel of entity B. <p>NOTE: The first 8 words are allocated for the device control block.</p>
M340	1. Allocates the memory for writing for any existing channels.
M580	2. Allocates the memory for reading for any existing channels.

NOTE: EcoStruxure Process Expert detects an error in the configuration during build if you have not configured sufficient memory for a client controller based on the Quantum platform with NOC communication module (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).

Allocation of Peer to Peer Communication Variables

The build process allocates addresses or values to the network variables that you have mapped, page 738 to communication channels.

The start address of a variable in the channel needs to be even and contiguous with the channel. Therefore, if the start address of the communication channel created in the client is:

- Odd:
 - No variable is allocated to the first word of the client communication channel.
 - Addressing of variables in this channel starts from the second word.
 - The offset of the server read/write memory counterpart is the first word of the server communication channel.
- Even:
 - Addressing of variables in this channel starts from the first word.
 - Read and write size of the memory is equal to the value of the **ServerMemoryLength** parameter - 1.
 - The offset of the server read/write memory counterpart is the second word of the server communication channel.

NOTE: If by using the **Refine** command, you delete or change the name, type (ID or version), or size of any variable or network variable that is mapped to a communication channel, the build process cannot complete successfully. Diagnostic information is displayed in a dialog box. In this case, open the **Variable Mapping Editor** to remove, page 746 the inconsistent variable or undo the changes to the variable. Start the build process over.

The figure shows examples of entries created during build in the **consumer** project, page 463 (client based on the M580 platform) for the **Value** attribute of two sample peer to peer communication variables that are used to read data from the server (*DISignal_UL_1V*) and write data to the server (*Valve_1_VALVE_OPV*).

Name	Type	Value	Address
DISignal_UL_1V	REF_TO ANY_BOOL	REF(M580_Server_BMENOC030122.Inputs.Free[0].0)	
Valve_1_VALVE_OPV	REF_TO ANY_BOOL	REF(M580_Server_BMENOC030122.Outputs.Free[0].0)	

The figure shows examples of entries created during build in the **owner** project (server based on the M580 platform) for the **Value** attribute of the two counterpart sample peer to peer communication variables.

Name	Type	Value	Address
DISignal_UL_1V	REF_TO ANY_BOOL	REF(%MW2.0)	
Valve_1_VALVE_OPV	REF_TO ANY_BOOL	REF(%MW52.0)	

Aborting Build Tasks

After selecting a build command or confirming it, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. No build is performed.

Only the user who selected the command is allowed to cancel it.

Using the Build Commands

To build the Control Participant project, modify the built Participant project through an incremental build, or build it again, proceed as follows.

Step	Action	Result
1	Right-click the executable and select Build , Build All , or Generate and Build depending on the situation, page 783.	<p>If no validity icon, page 777 is displayed, the following actions are executed; otherwise a confirmation dialog box opens. In this case, proceed to step 2.</p> <ul style="list-style-type: none"> The built Control Participant project is created or updated. The program file (.stu) is stored in the content repository. Information appears in the notification panel. A dialog box containing a summary of the build process opens.
2	In the confirmation message, click No . NOTE: Click Yes to proceed with the build anyway; the process may not complete successfully.	The confirmation dialog box closes without building the Control Participant project.
3	Refine the Control Participant project, page 449 to fix the issue.	—
4	Start the build process over.	—

NOTE: If the build process does not complete successfully, a notification is displayed to inform you and provides indications about the cause. In this case, make the necessary corrections and build the project again.

Build Status

The table describes the build status of an executable, page 722.

Status	Description
Not Built	No built Participant project exists. Default status when you create an executable.
Built	The built Participant project is up to date.
Out Of Date	The built Participant project is out of date and needs to be built again, page 783.

NOTE: You may need to build the Control Participant project again if you have made certain changes, page 781 to the system even though the **Build State** is displayed as **Built**.

Viewing the Built Participant Project

Once you have built the Control Participant project, you can open the project in the Control Participant to view it.

You may also be able to open the built Control Participant project if the errors are detected during the build process to help you with the troubleshooting.

NOTE: Do not make changes to the built Control Participant project from within the Control Participant as the software discards any changes upon closing the Participant window.

To make changes to the system, you can use the following methods:

- Open the system using the corresponding explorer.
- Refine, page 446 the logical Control project.

To open the built Control Participant project in the Control Participant, proceed as follows.

Step	Action
1	Right-click the Control executable and select Open Built Project . Result: The Control Participant opens extracted, page 118 and maximized. You can view the contents of the built executable. NOTE: For more information, refer to Viewing the Built Control Participant project (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).
2	Close the Control Participant window.

NOTE: To save a copy of the built Control Participant project, use the **Save as** command of the **Content Repository** explorer, page 1123.

Changing the Built Control Participant Project

Overview

This automated process allows you to change the built Control Participant project by integrating changes that you have made to the Control Participant project, or to the topology of the system.

Depending on the modifications you made, the status of a built Control Participant project changes and you may need to use a specific command to integrate changes.

Actions Changing the Build Status

The following table describes the impact of various actions on the state of a Control executable that is **Built**.

Change description	Resulting build state
Changes to the topology of the system (for example, entity properties, controller rack configuration, IP addresses, physical and network connections, replacing templates of I/O devices, deleting STB islands, functional safety password). NOTE: For M580 controllers, the build state changes only after you apply the changes to the configuration by using the Configure command or the Synchronize Changes command if the Participant is open.	Out Of Date
Changing the logical Control Participant project and generating it.	
Deleting a section in the Project Explorer .	
Selecting the Refine command after adding, renaming, or changing the order of sections in the Project Explorer .	
Refining the Control Participant project.	
Selecting Built Participant Project for an executable when importing a Control Participant project in the Project Explorer . It applies in the following cases: <ul style="list-style-type: none">• An executable with the same identifier already exists in the target project and its state is Built.• You are importing an executable that does not exist yet in the target project and its state was Built at the time of export.	
Updating, page 897 the logical Control Participant project.	
Changing the service, communication, or hardware mapping of the Control executable. For communication mapping, if a peer to peer communication channel exists, changes include creating, modifying, or deleting a variable mapping, page 738 ⁽¹⁾ .	Not Built
Removing the service mapping.	
Deleting a controller or workstation that is selected in the service mapping, or deleting the Control Expert service, page 598 of such a workstation.	
(1) When a communication channel and variable mappings for peer to peer communication exist between two built Control project executables, if you make a change that impacts the communication channel or variable mapping of one executable, which sets its build state to Out Of Date , it also changes the build state of the other executable to Out Of Date . Such a change can be, for example, deleting the Control project or its executable, deleting a mapped variable, deleting a section containing a mapped variable, or disconnecting a mapped controller from the logical network.	

NOTE: After making a change that modifies the build status, build the logical Control Participant project again by using the appropriate command, page 783.

Building after Creating, Updating, or Deleting Device DDT Variables in the Logical Control Project

If you have created a device DDT variable, page 455 in the logical Control project and this variable does not exist in the built Control project, the updated built Control project contains it in unmapped (unlocked) state.

The following table describes the result when building changes if a device DDT variable with the same name exists in both the logical and the built Control project but with a different data type or type version.

DDDT variable in the built Control project	DDDT variable in the logical Control project	Result in the updated built Control project
Mapped	Unmapped	Variable of the same type as it was in the built Control project.
Unmapped	Unmapped	Variable of the same type as in the logical Control project if the type that is updated is not used by a mapped device DDT variable.

The following table describes the result when building changes if a device DDT variable that exists in the built Control project has been deleted from the logical Control project.

DDDT variable in the built Control project	DDDT variable in the logical Control project	Result in the updated built Control project
Mapped	Variable is deleted and references to it in code are removed.	<ul style="list-style-type: none"> The device DDT variable is not removed. Code is updated.
Unmapped		<ul style="list-style-type: none"> The device DDT variable is removed. Code is updated.

Using the Appropriate Command to Integrate Changes

The table indicates which command to use to update the built Control Participant project depending on the change that you have made.

Change description	Command	Action
<ul style="list-style-type: none"> Adding, deleting, or changing: <ul style="list-style-type: none"> Any section (including the order of sections) Variables⁽¹⁾ (including device DDT variables, page 455) The hardware mapping CCOTF-compatible changes, page 585 in the configuration of a controller⁽²⁾ 	Build ⁽⁵⁾	Performs an incremental build.
Changing the topology of the system ⁽³⁾ .	Build All	Performs a complete build.
Changing the communication or network variable mapping ⁽³⁾ .		
After an import into a Control Participant project ⁽³⁾ .		
After regenerating a Control Participant project ⁽³⁾ .		
Adding, deleting, or changing: <ul style="list-style-type: none"> Animation tables⁽⁴⁾ Operator screens⁽⁴⁾ DDTs or DFBs⁽³⁾ 		
Changing the order of FBD or non-FBD sections in the Control Participant project by using the Refine command ⁽⁴⁾ .		
Moving FBD or non-FBD sections from one task folder to the other (for example, from MAST to FAST) while refining the Control Participant project (Refine command).		
Changes that set the build status to Not Built , page 781.	Build or Build All	Either command performs a complete build.

- (1) To integrate changes to attributes of variables used in hardware mapping, page 784, use the **Build All** command.
- (2) Except changes to the S908 architecture, which require using the **Build All** command to integrate these changes even if the **Build** command is available.
- (3) Such changes disable the **Build** command. Except for CCOTF-compatible changes.
- (4) Using the **Build** command after such change does not update the built Control Participant project.
- (5) You can also use the **Generate and Build** command, page 399.

NOTE: When you modify an existing peer to peer communication channel, rebuild the owner and the consumer, page 463 Control Participant projects.

Performance Considerations

When you have made changes to a built Control Participant project and the appropriate command to update it is **Build**, in certain cases you may be able to reduce the time required to update the built Participant project by using the **Build All** command instead.

This can be the case, for example when you make changes to many sections of a built Control Participant project at once.

Managing Hardware Abstraction Layer (HAL) Variables and DFBs

Overview

The Global Templates Library contains HAL application templates, which generate elementary variables representing the value and/or quality of hard-wired signals.

HAL is a mechanism that emulates access to hardware resources to maintain an application device-independent. By using such templates, you can use the elementary variables that they generate although the hardware channels are not yet part of the logical Control Participant project. Data related to the topology is associated to the logical Control project during build.

The following types of templates exist:

- HAL templates generating elementary digital or analog Control Participant variables (for example, signal conditioning templates).
- Special card templates containing DDTs, DFBs, and/or IODDTs, allowing to use the data exchanged with the special in-rack modules (for example, Modicon M340 high speed counter module device templates).

Facets referenced by these templates are used during the hardware mapping.

For details on these templates, refer to the EcoStruxure Process Expert help, page 104.

Marking HAL Variables and DFBs

When you build the logical Control Participant project, EcoStruxure Process Expert marks the following code that is used in the hardware mapping with the *Custom* attribute **HAL**:

- Elementary variables coming from mapped facets of hardware abstraction layer templates or special card templates.
- DFB instances and IODDTs coming from mapped facets of special card templates and that are added to a section during build.

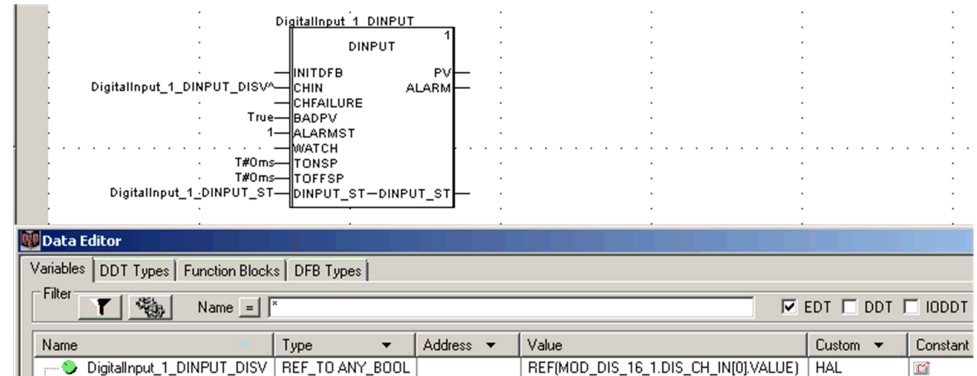
Limitations can apply when you:

- Refine sections containing marked DFB instances, using the **Refine Online** command (see EcoStruxure™ Process Expert, Control Participant Services, User Guide).
- Update the logical Control Participant project, page 897 with changes made to HAL variables or sections containing marked DFB instances.

NOTE: DFB instances and IODDTs are marked with the attribute **HALFB** to distinguish them from EDTs.

Properties of HAL Variables

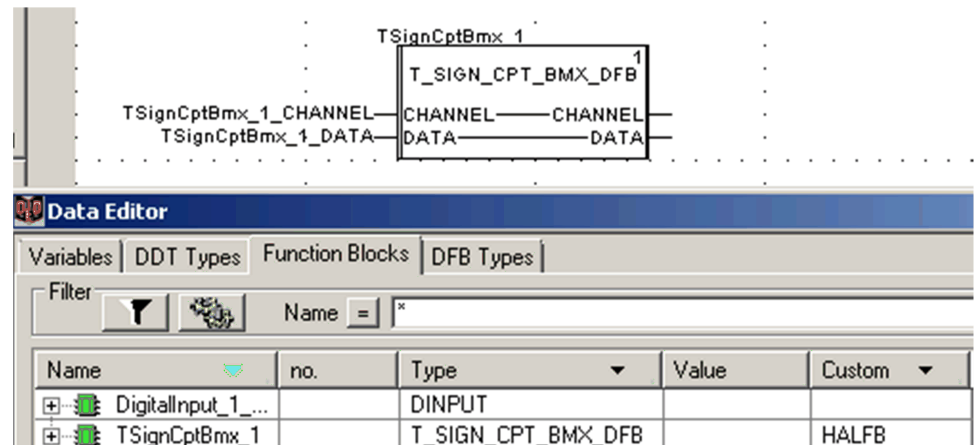
In this example, facets of instance **DigitalInput_1** have been assigned to an FBD section and generated. In the hardware mapping, the **DigitalInput_1_DINPUT_DIS** facet is mapped to a channel of the digital input module of the controller. The controller entity is based on the M580 CPU. The following figure shows the built Control Participant project where the **DigitalInput_1_DINPUT_DISV** variable that is used in the hardware mapping is marked.



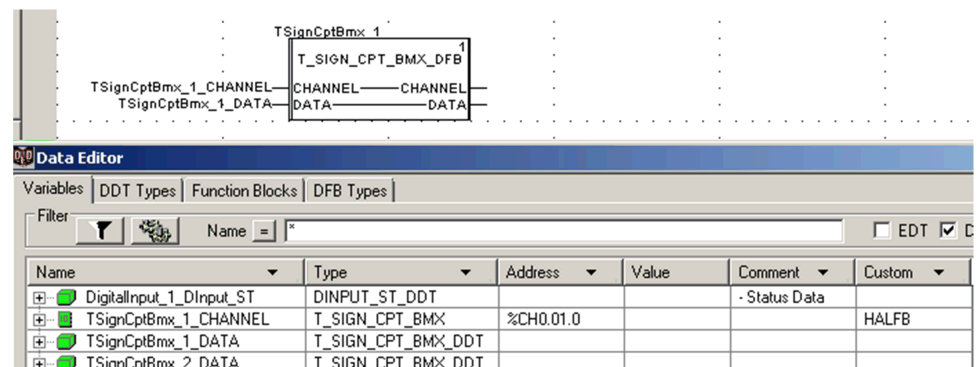
Properties of HAL IODDT Variables and DFBs

In this example, the facet of instance **TSignCptBmx_1** (modeling a high speed counter module) has been assigned to an FBD section and generated. In the hardware mapping, the **TSignCptBmx_1** facet is mapped to a channel of the high speed counter module of the controller. The following figure shows the built Control Participant project where the **TSignCptBmx_1** DFB encapsulated in this facet is added to the section and marked.

NOTE: The data type and addressing method for variables shown in this example may vary depending on the version of the template generating them.



The following figure shows the same built Control Participant project where the corresponding IODDT **TSignCptBmx_1_CHANNEL** was created and marked.



NOTE: A **TSignCptBmx_2_DATA** facet of another high speed counter instance, for the second channel of the counter module, has been assigned to the FBD section and generated. However, the hardware mapping for the second channel has not been done, thus the second DFB instance and the corresponding IODDT have not been added to the section.


Viewing Sections Containing HAL DFBs and Variables

Once you have built the Control Participant project, use the **Open Built Project** command to view the **HAL** attribute, allowing you to identify variables and DFB instances that were marked.

NOTE: After deployment, you can also view this attribute by using the **Refine Online** command.

NOTE:

This attribute is not visible when you refine the logical Control Participant project by using the **Refine** command.

Variables DDT Types Function Blocks DFB Types					
Filter		Name = *		<input checked="" type="checkbox"/> EDT <input type="checkbox"/> DDT <input type="checkbox"/> IODDT <input type="checkbox"/> Device DDT	
Name	Type	Address	Value	Custom	
 DigitalInput_1_DINPUT_DISV	REF_TO ANY_BOOL				

Changing HAL Variables

To integrate into the built Control Participant project changes that you make to variables that are marked as **HAL** by using the **Refine** command, refer to [Using the Right Command to Integrate Changes](#), page 783.

Supervision Participant Project Build Stage

Overview

This section describes how to use the **Project Explorer** to create the built Supervision Participant project for the first time and, in a second step, to integrate changes that you have made to the Supervision Participant project.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Building the Supervision Participant Project

Overview

In this automated process, by using the compilation mechanism of the Supervision Participant, EcoStruxure Process Expert creates a unique built Supervision Participant project, which merges:

- The logical Supervision Participant project information.
- The mapping information, which is associated to the executable of the logical Participant project:
 - Services
 - Communication
- The topology information, which is associated to the mapped services.

The resulting Supervision project file (.ctz) is stored in the content repository, page 1120.

This is the file that you can deploy to computers running Supervision software.

I/O Device Addresses

When the protocol for a tag container is **OPCUA** or **OPCUA Embedded**, the build process replaces the temporary IP address and port value, page 309 in the address of the I/O device that is associated to the container in order to create the actual I/O device address in the Supervision Participant project.

Protocol for tag container	I/O device address format
OPCUA	<p><i>opc.tcp://<IP address>:<TCP listening port>/OPCUAServerExpert</i></p> <p>Where:</p> <ul style="list-style-type: none"> • <IP address> is the IP address of the workstation, page 598 that is mapped, page 767 to the <i>IOServer</i> service. • <TCP listening port> is the value that is configured for TCP Port for this workstation.
OPCUA Embedded	<p><i>opc.tcp://<IP address>:<TCP listening port></i></p> <p>Where:</p> <ul style="list-style-type: none"> • <IP address> is the IPv4 address (see EcoStruxure™ Process Expert, Control Participant Services, User Guide) of the communication module embedding the OPCUA server and that is selected in the communication mapping, page 769 for the I/O device associated to the tag container. • <TCP listening port> is the value that is configured for this module in the Control Participant.

NOTE: For other protocols, refer to the **Address** property, page 309 of the I/O device.

OFSOPC and OPC UA Configuration Files

During the build process, EcoStruxure Process Expert generates the necessary OPC Factory Server and OPC UA Server Expert configuration files and stores them in the contents repository.

One configuration file is created for each workstation that is mapped, page 767 to an *IOServer* service. The name of the configuration file contains a suffix that identifies the protocol that is used.

In case more than one protocol is used in the Supervision project, at least one configuration file per protocol is generated.

Protocol used	Configuration file name	Configuration file aliases
OFSOPC	<p><i><Topological entity>_DeviceConfig_OFSDA.xml</i></p> <p>For example, <i>OperationServer_DeviceConfig_OFSDA.xml</i></p>	<p>One device alias exists for each I/O device of the same I/O server.</p> <p>The alias name is <i>Alias_<I/O device></i> where <i><I/O device></i> is the identifier of the I/O device that is mapped, page 769 to the data source.</p> <p>For example, <i>Alias_IODevice_1</i>.</p>
OPCUA	<p><i><Topological entity>_DeviceConfig_OPCUA.xml</i></p> <p>For example, <i>OperationServer_DeviceConfig_OPCUA.xml</i></p>	<p>One device alias exists for each tag container.</p> <p>The alias name is the tag container identifier, page 354.</p> <p>For example, <i>TagContainer_1</i>.</p> <p>NOTE: The User authentication and X509 user token properties are enabled by default.</p> <p>For details, refer to <i>User Authentication and Authorization</i> in the help of OPC UA Server Expert.</p>
OPCUA with OPC UA server embedded in module of controller, page 354	No configuration file is generated.	<p>Not required.</p> <p>The OPC UA server is embedded in a module of the controller that is the data source.</p> <p>The variable tag address has the format <i>ns=2;s=0:<variable name></i> where <i>ns</i> indicates the namespace and 2 corresponds to the OPC UA server as defined by Schneider Electric.</p> <p>For example, <i>ns=2;s=0:InputVar1</i></p>
Where <i><Topological entity></i> is the identifier of the workstation to which the <i>IOServer</i> service is mapped in the service mapping, page 767		

If you rename the workstation, the name of the configuration file is updated without the need to build the Supervision Participant project.

NOTE: For information on how to generate symbol table files (.xvm), refer to the topic describing the Control project deployment process, page 798.

Network Addresses of Supervision Project Components and Properties of Workstations

When you build a Supervision Participant project, and you have not done the service mapping, page 767 of a server and/or client of the project, an IP address is not generated for this component. Because of this, the build process of the Supervision Participant project cannot complete successfully.

Proceed with the service mapping of any Supervision Participant project component that you have created before building the project.

For example, if you have not mapped the alarm server of the project, the network address of the associated workstation is not generated into the Supervision Participant project.

NOTE: If you see a notification about TLS exchange after the build completes, verify that you have entered a value in the **Computer Name** property, page 598 of the workstations that are used in the service mapping with the Supervision executable.

Building When Instance Identifiers Start with a Digit

When the logical Supervision Participant project that you want to build contains facets referenced by instances whose identifier starts with a digit, you need to set

the *tagstartdigit* parameter of the Supervision Participant configuration file (.ini) to 1 for the build process to complete successfully.

Building the Supervision Project for the First Time

Use the **Build** or **Build All** command indifferently to create the built Supervision Participant project for the first time.

NOTE: If you have not built the Supervision Participant project yet, you can also use the **Generate and Build** command. For information on the generation process, refer to the topic describing the generation of Participant projects, page 398.

Aborting Build Tasks

After selecting a build command, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. No build is performed.

Only the user who selected the command is allowed to cancel it.

Using the Build Commands

To create the Supervision Participant project, modify it, or to create it again, right-click the executable and select **Build**, **Build All**, or **Generate and Build** depending on the situation, page 792.

Result: The following actions are executed:

- The Supervision Participant project is created or updated.
- The project files are stored in the contents repository.
- Information is displayed in the notification panel.
- A dialog box containing a summary of the build process opens.

NOTE: Building the Supervision project may take time.

NOTE: If the build process does not complete successfully, a notification is displayed to provide indications about the cause. In this case, make the necessary corrections and build the project again.

Build Status

The table describes the build status of an executable, page 765.

Status	Description
Not Built	No built Participant project exists. Default status when you create an Executable .
Built	The built Participant project is up to date.
Out Of Date	The built Participant project is out of date and needs to be built again.

NOTE: You may need to build the Supervision Participant project again if you have made certain changes, page 790 to the system even though the **Build State** is displayed as up to date.

Changes Modifying the Build Status

The table describes how changes affect the status of the executable of a Supervision Participant project that is already built.

Change	Resulting build state
Renaming an I/O device or the Supervision project.	Out Of Date
Generating the project after making changes.	
Creating a page.	
Changing the project through refinement.	
Any import operation in the Project Explorer .	
Changing service or communication mapping of the Supervision executable.	
Changes to the topology of the system, for example: <ul style="list-style-type: none"> Changing the IP address of the NIC of a workstation or of a module that is selected in the service or communication mapping. Deleting the Control Expert or OFS service, or the NIC instance of a workstation that is selected in the communication mapping. Deleting the Supervision service of a workstation that is selected in the service mapping. Deleting a controller that is selected in the communication mapping. Changing the TCP Port of a workstation that is selected in the service mapping for the I/O server when the protocol that is used for at least one tag container is OPCUA. 	Not Built
Removing all service mappings of the Supervision executable.	
Deleting a workstation that is selected in the service mapping.	

Downloading Supervision Project Files

After the build process completes successfully, the following files are available for download in the **Download Projects** window, page 764:

- The master project (.ctz) with the name *ProjectIdentifier_ExecutableIdentifier* (for example, *Supervision_1_Executable_1*).
The project also contains any user files, page 508 that are attached to the Supervision Participant project.
- The included projects (.ctz) that are related to animated graphics, which have been added to Supervision pages, page 508.
- Any other included project that you have added manually, page 510 to the Supervision Participant project.
- The OFSOPC/OPC UA configuration files (.xml) if you have configured the communication board, page 312 driver as OFSOPC or OPCUA.

NOTE: You can download included projects as soon as they are added to the Supervision Participant project.

To download a project file, proceed as follows.

Step	Action
1	Right-click the executable that you have built and select Download Projects . Result: The Download Projects window opens and shows the project files associated to the built Supervision Participant project.
2	Select one file and click Download . Result: The Save As dialog box opens.
3	Browse to the location where you want to save the file and click Save .

Changing the Built Supervision Participant Project

Overview

This automated process allows you to change the built Supervision Participant project by integrating changes that you have made to the Supervision Participant project.

For performance considerations, when possible, use the **Build** command. It requires less time to complete than **Build All**. However, you can use the **Build All** command in any case.

NOTE: You can also use the **Generate and Build** command, page 399.

Using the Right Command to Integrate Changes

The table indicates which command to use to update the built Supervision Participant project depending on the type of change that you have made.

Change	Command	Action
Changes involving Supervision pages: <ul style="list-style-type: none"> Creating or deleting a page. Editing a page (Edit command): <ul style="list-style-type: none"> Adding or deleting one or more genies or graphical elements. Editing page properties. Generating a page that contains a genie facet with the status Out Of Date or Deleted. 	Build ⁽¹⁾⁽²⁾	Performs an incremental compilation.
Changes involving Supervision data facets ⁽²⁾ : <ul style="list-style-type: none"> Assigning or unassigning a facet from a tag container. Moving an existing facet to another tag container. Generating a tag container that contains a facet with the status Out Of Date or Deleted. 		
Modifying a parameter of the Message category, page 183 of a Supervision element of an instance and generating the corresponding Supervision project (for example, modifying the description of an interlock condition of the InterlockTags element under the Supervision node of an instance).		No compilation is performed.
Other changes, for example, to: <ul style="list-style-type: none"> Clusters. The trend, alarm, report, and/or I/O server. Service and communication mappings. Advanced project settings. Generating or refining the Supervision Participant project (except when generating only changes that involve a Supervision data facet as described in this table). Topological entities, for example, changing the TCP Port of the I/O server workstation when the OPCUA protocol is used or changing the configuration of a module that embeds an OPC UA server. 	Build All ⁽²⁾⁽³⁾	Performs a complete compilation.
(1) You are limited in the number of changes that you can make per Supervision Participant project to be able to use the Build command: <ul style="list-style-type: none"> A maximum of 5 changes involving Supervision pages. A maximum of 30 changes involving data facets. When you exceed either limit, you must use the Build All command.		
(2) You can also use the Generate and Build command, page 399.		
(3) When you make this type of change, the Build command is not available.		

Deployment Stage

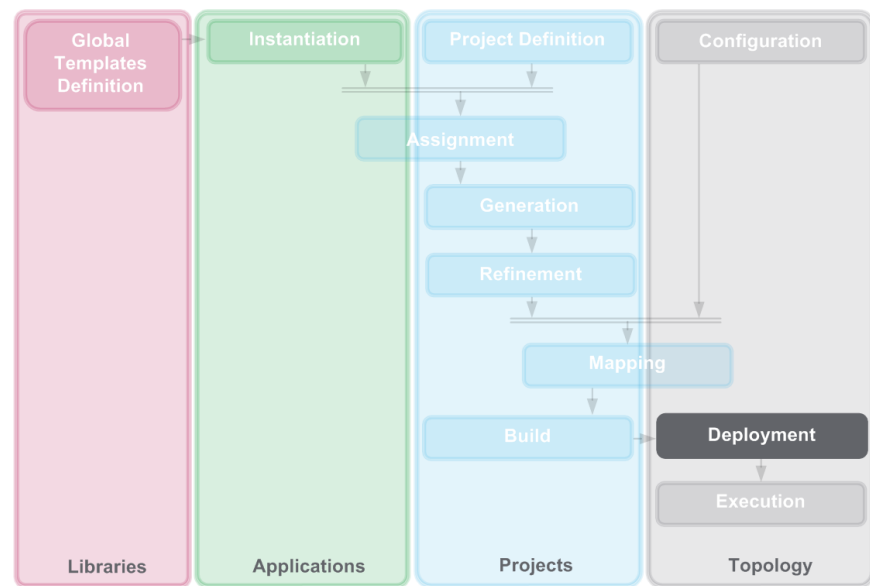
What's in This Chapter

Control Project Deployment Stage	794
Supervision Project Deployment Stage	841

Overview

This chapter describes how to deploy the built Control and Supervision Participant projects to entities that exist in the topology of the system.

The following figure shows the position of the **Deployment** stage within the system engineering life cycle.



Refer to Deployment Stage, page 67 for a description of the purpose of this stage.

Control Project Deployment Stage

Overview

This section describes how to use the **Topology Explorer** to deploy entire built Control Participant projects or parts of them to physical controllers or to PCs emulating simulators, which are modeled by engines in the topology of the system.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Control Participant Project Deployment Process

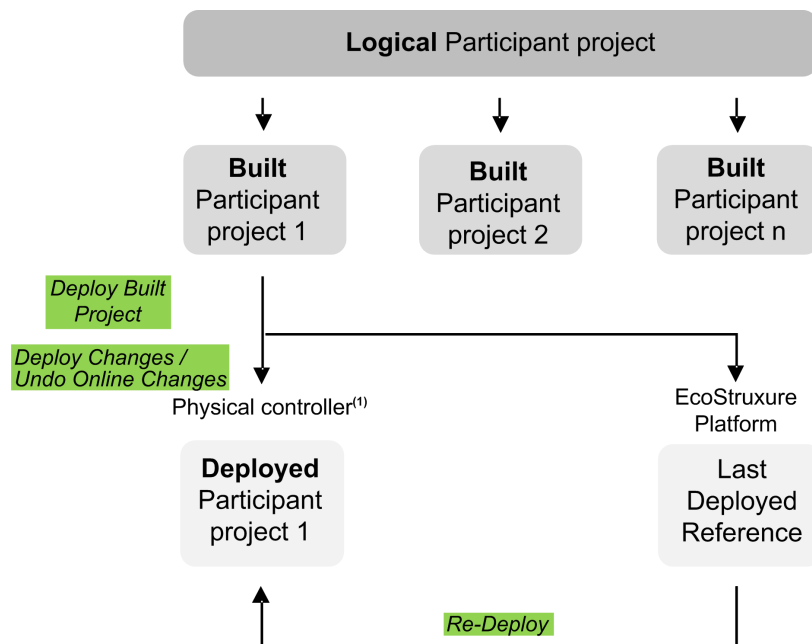
Overview

You can deploy a built Control Participant project, page 775 to engines by using the following commands:

- **Deploy Built Project**
- **Re-Deploy Last Project**
- **Deploy Changes / Undo Online Changes**

Deployment Flowchart

The following figure illustrates the different steps of the deployment process of the built Control Participant project and their associated commands.



(1) Or controller simulator (Control service of a workstation).

Prerequisites

The following table describes the actions that are prerequisite to using a command.

Command	Prerequisites
Deploy Built Project	Build the Control Participant project and configure the corresponding topological entity.
Re-Deploy Last Project	Deploy the built Control Participant project.
Deploy Changes / Undo Online Changes	<p>Update the project, page 897 with hardware changes made online (if applicable).</p> <p>In addition, the following are prerequisites for specific actions:</p> <p>For the Deploy Changes action:</p> <ul style="list-style-type: none"> • Deploy the built Control Participant project. • Change the logical Control Participant project and build it. <p>For the Undo Online Changes action:</p> <ul style="list-style-type: none"> • Deploy the built Control Participant project. • Change the deployed Control Participant project by using the Refine Online command. <p>NOTE: Certain changes cannot be deployed, page 834 by using the Deploy Changes command.</p>

For the other actions that are required, for example, mapping the Control executable to a controller, refer to the description of the system engineering life cycle in this document.

Supported Engines for Deployment

You can deploy/re-deploy a built Control Participant project and deploy changes to engines modeling:

- A controller
- A redundant controller of the M580 or Quantum platform.
- A computer running a controller simulator, page 597 (modeled by the Control service of a workstation in the topology of the system).

Engines must exist in the topology of the system and be mapped to a Control executable of the project.

Identifying Engines

An engine is identified on the logical network by using an IP address.

⚠ WARNING
<p>UNINTENDED EQUIPMENT OPERATION</p> <p>Ensure that you are connected to the controller you intend to deploy the Control Participant project.</p> <p>Failure to follow these instructions can result in death, serious injury, or equipment damage.</p>

NOTE: To obtain a list of IP addresses of nodes that are connected to the same physical network and their corresponding physical address (Media Access Control (MAC) address), use the Address Resolution Protocol (ARP).

The procedure describes how to use the ARP to obtain a list of IP addresses of nodes connected to the same physical network and their corresponding MAC address.

Step	Action
1	Write down the MAC address that is printed on the communication module to which you want to deploy or execute.
2	Connect the computer to the logical network to which this communication module is connected.
3	On the computer, execute the <i>cmd.exe</i> program. Result: A command prompt opens.
4	Type <code>arp -a</code> and press Enter . Result: The local ARP cache table opens.
5	In the table, locate the MAC address matching that printed on the communication module.
6	In the table, ensure that the IP address that corresponds to this MAC address matches the IP address that you have assigned to the engine in the Control Participant; otherwise, proceed to step 7.
7	Assign to the communication module the IP address that you have assigned to the corresponding engine in the Control Participant and repeat the procedure, starting from step 2.

Client/Server Connection

After an unexpected interruption of the client/server connection, the notification panel of engineering clients that were connected to the system server may contain information about operations that did not complete successfully because of this interruption. It may also indicate corrective actions.

After an interruption of the client/server connection, the states of the engineering client and the system server may be inconsistent if a deployment operation was in progress.

NOTICE
DATA CORRUPTION <p>After an interruption of the client/server connection, ensure that the last operation executed by EcoStruxure Process Expert completed successfully and if necessary, repeat the last operation.</p> <p>Failure to follow these instructions can result in equipment damage.</p>

NOTE: For more information on client/server connection interruption, refer to the topic describing the [client/server connection](#), page 76.

Start and Stop Commands

EcoStruxure Process Expert can send start and stop commands to the controller.

⚠ WARNING
UNKNOWN OPERATIONAL STATE OF EQUIPMENT <p>Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.</p> <p>Failure to follow these instructions can result in death, serious injury, or equipment damage.</p>

Generating Symbol Table Files

You cannot generate symbol table files (.xvm) by selecting the *saveXVM* property in the project settings of the Control Participant. Instead, you must edit a dedicated configuration file before deploying a Control project.

To generate a symbol table file (.xvm) of the Control Participant and retrieve it from the content repository, proceed as follows.

Step	Action
1	Open the <i>ControllerSettings.xml</i> file that is located at the path <i>C:\Program Files\Schneider Electric\EcoStruxure\Process Expert\System Server</i> on the computer on which the system server is installed.
2	Set the value for the <i>SaveXVM</i> attribute to 1 and save changes. Result: When you deploy changes, page 830 to a Control project or refine it online, page 874, the symbol table file is added to the content repository.
3	In the content repository, open the <i>XvmHolder</i> folder located under Systems > <System name> > Project > Upro.
4	Save the file, page 1123.

Deploying to the Controller Simulator

Overview

You can deploy built Control Participant projects to computers for simulation purposes. EcoStruxure Process Expert supports deployment to multiple Controller simulator instances running on a single or several computers.

In the topology, each of these computers is modeled by a [workstation](#), [page 597](#) on which one or more Control Expert services (engine) are created, modeling simulator instances.

By default, each simulator instance is configured to start with a password-protected Control project (see *EcoStruxure Process Expert, Installation and Configuration Guide*) provided with EcoStruxure Process Expert. You can start a simulator instance with a different password-protected project file or change the password, [page 808](#) of the project that is loaded by default.

NOTE: If your Windows session is not the one that was used to install EcoStruxure Process Expert, you must [configure the simulator manually](#), [page 801](#) to load a password-protected Control project at startup.

Before Deploying to the Simulator

The following table describes the actions you must perform to deploy a Control project that is mapped to a workstation/Control Expert service, depending on the computer on which the simulator is running.

Computer selected for deployment	Actions
Computer on which Ecostruxure Process Expert is installed.	Start a simulator instance.
Other computer on which Ecostruxure Process Expert is not installed.	<ol style="list-style-type: none">1. Install the simulator.2. Start a simulator instance. Refer to the topic describing how to install the Controller Simulator. (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>)

Starting the Controller Simulator

To start an instance of the controller simulator, proceed as follows.

Step	Action
1	<p>Double-click the shortcut of the controller simulator on the desktop.</p> <p>Result: The Process Expert Controller Simulator dialog box opens.</p> <p>NOTE: If you installed the simulator manually, double-click the <i>PLCSimulatorStarter.exe</i> file.</p> <p>NOTE: You can also start a simulator instance by using the command prompt. Refer to the topic describing how to load a project in the simulator in the help of the simulator.</p>
2	<p>Edit the port number that you want the simulator instance to use (port 502 by default) and click Start.</p> <p>Result: The simulator instance starts minimized as an icon in the notification area and uses the configured port. By default the password-protected Control project (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>) is loaded.</p> <p>NOTE: To deploy a Control project, the port number that is used by the simulator instance must match the value of the Port property of the Control Expert service that is used in the service mapping of the executable of the project.</p>
3	To start another instance, edit the port number and click Start again.

NOTE: For information on using the simulator, refer to *EcoStruxure Control Expert, PLC Simulator*, which is located in the PLC_Simulator folder. You can also open the help of the simulator from the simulator panel after starting it.

Starting the Simulator With Another Password-Protected Control Project

You can modify the startup setting and have simulator instances load your own password-protected Control project file (.sta).

Once you have configured the simulator to start by loading your own Control project, you must enter the password to use it, page 806.

NOTE: The change does not apply to simulator instances that are already running. You must restart them.

To start simulator instances by loading a specific password-protected Control project, proceed as follows.

Step	Action
1	Start a simulator instance.
2	<p>Right-click the simulator icon in the Windows taskbar and select Options...</p> <p>Result: The Simulator Panel Options dialog box opens.</p>
3	Select the Use default application to start simulator (enforce security) check box.
4	Click the browse button and select your password-protected Control Expert Control project file (.sta).
5	<p>Click OK.</p> <p>Result: The simulator stops.</p>
6	<p>Start the simulator.</p> <p>Result: The simulator starts and loads the selected Control Expert Control project file.</p>

Starting the Simulator Without Password-Protected Control Project

You can modify the startup setting and start simulator instances without loading a password-protected Control project file (.sta). However, this leaves the Ethernet port of the computer that is used by the simulator vulnerable to cyberattack.

NOTE: The change does not apply to the other simulator instances that are already running. You must restart them.

To start simulator instances without loading a password-protected Control project, proceed as follows.

Step	Action
1	Start a simulator instance.
2	Right-click the simulator icon in the Windows taskbar and select Options.... Result: The Simulator Panel Options dialog box opens.
3	Clear the Use default application to start simulator (enforce security) check box.
4	Click OK . Result: You can start the simulator without entering a password.

Starting the Simulator Under Another Windows session

Once you have configured simulator instances to start by loading a password-protected Control project, you must enter the password to use it, page 806.

To start simulator instances by loading the default Control project provided with EcoStruxure Process Expert or another one when you are logged into a Windows session that is different from the one used to install EcoStruxure Process Expert, proceed as follows.

Step	Action
1	Start a simulator instance.
2	Right-click the simulator icon in the Windows taskbar and select Options.... Result: The Simulator Panel Options dialog box opens.
3	Select the Use default application to start simulator (enforce security) check box.
4	Click the browse button and select either of these password-protected Control project files (.sta): <ul style="list-style-type: none"> <i>Simulatorprofile.sta</i> that is provided with EcoStruxure Process Expert and that you copied to the computer. The file is located in the <i>AFS</i> folder in the root of the installation package. Your own Control Expert Control project file.
5	Click OK . Result: The simulator stops.
6	Start a simulator instance. Result: The simulator instance starts and loads the selected Control project file.

Obtaining the IP Address of the Simulator

If you leave 127.0.0.1 as IP address for the Network Interface Card (NIC) of the workstation, page 598 where the simulator is running, it automatically defaults to the simulator that is installed on the system server computer. To use the simulator that is installed on a computer where no system server is installed (for example, an engineering station), you must enter the IP address of the local computer on the subnet (host IP address).

To obtain the host IP address used by a simulator instance, proceed as follows.

Step	Action
1	Start a simulator instance.
2	Double-click the simulator icon in the notification area. Result: The Controller Simulator Panel is displayed.
3	Read the IP address in the Host IP address field.

Using Deployment and Execution Commands With the Simulator

The way to use the controller simulator is similar to using deployment and execution commands with a controller. You must use the [commands, page 598](#) that appear in the **Control** context menu of the workstation that you have mapped to the Control project.

If you have created several Control Expert services on the workstation and mapped them to Control projects, a dialog box lets you select the Control project to which you want the command to apply.

Communication Between the Computer and Engines

Controllers

To deploy a built Control Participant project, ensure that the computer that is running the engineering client can communicate through the network with the controller or its communication module that is modeled by the entity that you are deploying to.

For more information on connecting controllers to a logical network, refer to the Control Participant help.

Workstations

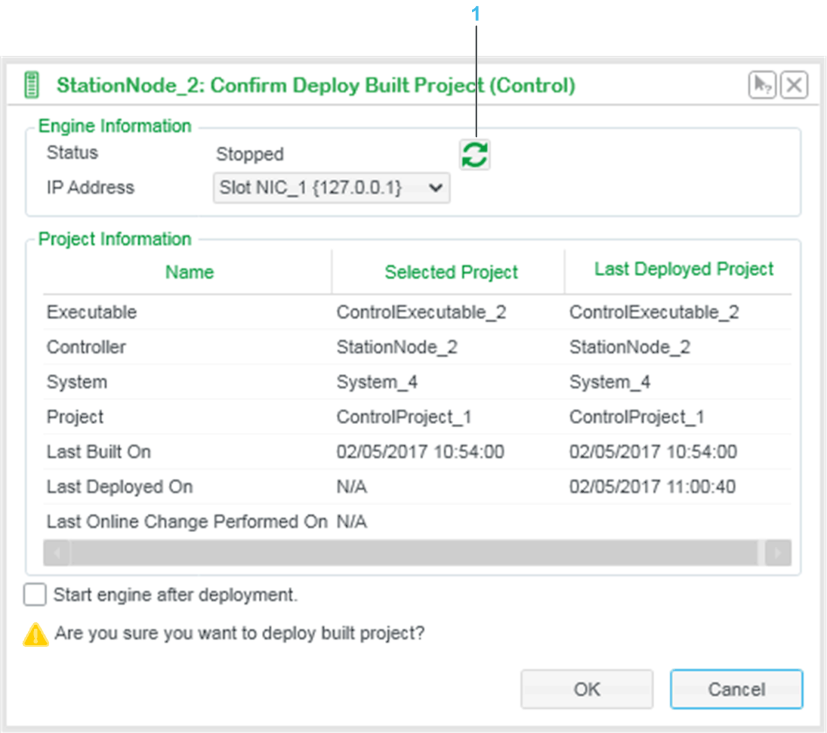
To deploy a built Control Participant project, ensure that the computer that is running the system server can communicate through the network with the computer that is modeled by the workstation that you are deploying to.

Refreshing Engine Status and Reconnecting Engines

When you use the various deployment and execution commands, a confirmation dialog opens where you must select the IP address of the target engine.

The engine must be reachable to deploy or execute a command. Once you have selected the IP address, the status of the engine is indicated in the **Status** field. If the status of the engine changes while the confirmation dialog box is open, the engine status is not refreshed. If the status of the engine is detected as **Not Reachable**, click the reconnect button to refresh the status of the engine and attempt to establish the connection.

The following figure shows an example of deployment confirmation dialog box featuring the reconnect button.



1 Reconnect button

Managing Passwords of Deployed Control Participant Projects

Overview

Using the **Controller/Simulator** password protection restricts deployment and execution operations for Control Participant projects that you perform by using an engineering client. It also restricts operations performed by using the operation client.

This topic describes how you can perform the following actions on controllers and the simulator running on a workstation:

- Setting a password for the first time.
- Changing a password. This includes changing a controller password that was set by using an application other than EcoStruxure Process Expert.
- Clearing a password.
- Resetting the password if you forgot it.

Setting a password is mandatory if the **Controller Access Password at Creation**, page 139 (formerly **Optional Security Services By Default**) property of the system is enabled; otherwise, you cannot deploy Control Participant projects.

For an overview of the entire Control project password management functionality, refer to the topic describing the system and Control facet template passwords (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

NOTE: For information on the default simulator password (to be entered in the **Current Password** text box to change it), refer to the topic describing how to install the controller simulator (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

NOTE: Deploying a Control Participant project after setting or changing the password takes slightly longer than deploying with an existing or without password.

Scope of the Password Protection

The **Controller** password that you can set by using the commands in the topological entity context menu corresponds to the application password of the Control Participant project that is deployed.

The application password and file encryption settings are disabled in the Control Participant (**Protection** tab of the properties of the project).

NOTE:

The management of the other passwords of the Control Participant is not supported by EcoStruxure Process Expert, except for the following:

- The **System Access Password**, which is the application password, page 141 of Control Participant projects that are managed from the **Projects Explorer** (offline). This password does not replace the **Controller** password when you deploy a Control project.
- The functional safety password, page 925 for M580 safety controllers.

The following table describes which Control projects benefit of the **Controller/Simulator** password protection.

Control Participant project	Password protection applies
Logical Control project	No ⁽¹⁾
Built Control project	No ⁽¹⁾

Control Participant project	Password protection applies
Deployed Control project	Yes ⁽²⁾
<p>(1) You must use the System Access Password property, page 139 of the system to enable password protection and file encryption.</p> <p>(2) File encryption is not enabled. To enable it, deploy a Control project after enabling system password protection.</p>	

Actions That Require Providing the Password

The table outlines the various actions for which you must provide a password if **Controller/Simulator** password protection is enabled for the controller or simulator running on a workstation respectively.

Topological entity	Action	Password that is required
Workstation/ simulator only	Setting the Simulator password.	<p>The password of the Control project file (.sta) that is loaded when the simulator starts.</p> <p>By default, this is the Control project file that is provided with EcoStruxure Process Expert (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>).</p> <p>NOTE: The simulator may have been configured, page 800 to load another Control project file when starting or the password may have been changed, page 808.</p>
	First deployment.	You must set a Simulator password first.
	Subsequent deployment and execution operations.	No password required as long as you keep the engineering client open.
Controller only	Setting the Controller password.	<p>No password is required.</p> <p>If a password is already set for the controller (for example, by using an application other than EcoStruxure Process Expert), the password is required to set a new one.</p> <p>NOTE: Setting the password when no Control Participant project is deployed to the controller requires stopping the controller.</p>
	First deployment.	You must set the Controller password first.
	Deployment and execution operations after setting or changing the password.	The Controller password that is set.
	Subsequent deployment and execution operations.	No password required as long as you keep the engineering client open.
	Navigating to a section by using Runtime Navigation Services (operation client).	<p>The password that is set.</p> <p>If you close the operation client or lock it and another user logs in, you must re-enter the password.</p>
	Resetting the password.	The temporary password provided by Schneider Electric support.

Topological entity	Action	Password that is required
Controller or workstation/ simulator/Control Expert service	Changing the Controller/Simulator password.	The Controller/Simulator password that is set.
	Clearing the Controller/Simulator password.	The Controller/Simulator password that is set.
	Using the Re-Deploy last Project command.	The Controller/Simulator password that is set. NOTE: If before redeploying the initial project (on which password A is set), another project on which password B is set and belonging to a different system was deployed from a different client to the same engine, during re-deployment of the initial project, you must enter password B. The deployment operation performed from a different system and client has changed the password that was set during the initial deployment (password A).
	Using the Back Up Data command for the first time after deploying a Control project.	The Controller/Simulator password that is set.
	Deployment and execution operations after importing the controller or workstation.	The password Controller/Simulator that is set. If no password is set, you must set one (or, for a workstation, enter the password of the Control project file (.sta) that is loaded when the simulator starts).
	Deployment and execution operations after restoring a system containing the controller or workstation.	
	Deployment and execution operations after restoring a database containing a system with the controller or workstation.	
	Recovering from a signature mismatch.	The Controller/Simulator password that is set.
	Disabling the Controller or Simulator password property after a password was set.	The Controller/Simulator password that is set. For a workstation, if several Control Expert services exist, you must enter the password for each one. NOTE: Disabling the password property opens the Clear Password dialog box. Entering the password there clears it.

Password Requirements

To be valid, the **Controller** and **Simulator** passwords must contain the following:

- From 8 to 16 characters.
- At least one uppercase letter from the classical Latin alphabet (A...Z).
- At least one lowercase letter from the classical Latin alphabet (a...z).
- At least one base-10 digit (0...9).
- At least one special character (~, !, @, \$, %, ^, &, *, _ , +, -, =, ` , |, \, (,), [,] , : , ; , ' , < , > , { , } , , , #).

Managing Application Passwords With Redundant Controllers

The following are the password management rules that apply to redundant controllers:

- Both primary and standby engines of the topological entity must be reachable by using the IP addresses that is configured and the high-speed cable must be connected.
- The same application password is set for the Control project of both engines. Changing or clearing the password applies to both engines.
- If the **ControllerExist** property, page 812 of a redundant controller is set to **Primary**, passwords are managed the same way as for a standalone controller.

The following describes the operating mode for redundant controllers if a password mismatch occurs:

- If setting, changing, or clearing the password for the Control project of the standby controller did not succeed, you must synchronize standby controller by using the **Sync (Primary→Standby)** command. You can proceed with the deployment to the primary controller.
- When you are prompted to enter the password, the password is compared with that of the Control project in both engines. If the password of the project in the standby engine does not match, you cannot perform the operation on the standby engine.

Setting the Application Password for a Control Project

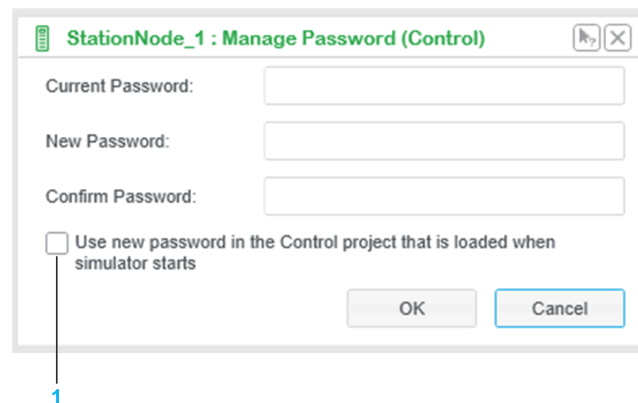
The following are prerequisites to setting the application password for a Control project that will be deployed:

- You created a topological entity of type controller or workstation.
- The topological entity is configured as follows:
 - For a controller, you used the **Configure** command and saved changes.
 - For a workstation, you created a Control Expert service.
- The **Controller** or **Simulator** password property of the topological entity is enabled.
- The engine of the topological entity must be reachable by using the configured IP address. In case of the simulator, it must be running.

You can set a password even if you already deployed a Control project while the password protection for the topological entity was disabled. The password is set for the already deployed Control project without stopping the controller.

NOTE: You can also use this functionality to change the password of the Control project that is loaded when you start the simulator.

The following figure shows the **Manage Password** dialog box for the controller simulator.



1 Check box that lets you change the password, page 810 of the Control project that is loaded when you start the simulator.

To set the application password for a Control project, proceed as follows.

Step	Action
1	Right-click the topological entity and select the following: <ul style="list-style-type: none"> For a controller: Manage Password For an M580 safety controller, select Controller in the submenu. For a workstation (simulator): Control > Manage Password If more than one Control Expert service exists, you can select the one to use. Result: The Manage Password dialog box opens.
2	Enter the password in the Current Password text box if required, page 806.
3	Enter the new password and confirm it.
4	To use this password also in the Control project that is loaded when the simulator starts, select the check box. It replaces the password that is set for this Control project file.
5	Click OK . Result: The password is saved and set as application password.

Verifying the Password to Perform Deployment and Execution Tasks

To enter the password that is set for a controller or simulator, proceed as follows.

Step	Action
1	In the Verify Password dialog box, enter the required password. NOTE: Alternatively, you can select the Use last entered password check box. It fills the text box with the last password that you entered in a Verify Password dialog box since the engineering client session was opened. The text box becomes read-only.
2	Click OK . Result: If the password is correct, the command is executed. If not, an information message is displayed, prompting you to re-enter the password. You can enter the password three times before the command is canceled. NOTE: Click Cancel to cancel the command you had selected.

Changing the Application Password of a Control Project

To change the password, the engine must be reachable at the configured IP address.

To change the application password for a Control project, proceed as follows.

Step	Action
1	Right-click the topological entity and select the following: <ul style="list-style-type: none"> For a controller: Manage Password For an M580 safety controller, select Controller in the submenu. For a workstation (simulator): Control > Manage Password If more than one Control Expert service exists, you can select the one to use. Result: The Manage Password dialog box opens.
2	Enter the password in the Current Password text box.
3	Enter the new password and confirm it.
4	To use this password also in the Control project that is loaded when the simulator starts, select the check box, page 808. It replaces the password that is set for this Control project file.
5	Click OK . Result: The password is saved and set as application password.

Clearing the Application Password of a Control Project

To clear the application password, the engine must be reachable at the configured IP address.

NOTE: Clearing the application password does not clear the password of the Control project that is loaded when the simulator starts.

To clear the application password for a Control project, proceed as follows.

Step	Action
1	<p>Right-click the topological entity and select the following:</p> <ul style="list-style-type: none"> For a controller: Clear Password For an M580 safety controller, select Controller in the submenu. For a workstation (simulator): Control > Clear Password If more than one password-protected Control Expert service exists, you can select the one to use. <p>Result: The Clear Password dialog box opens.</p>
2	Enter the password in the Current Password text box.
3	<p>Click OK.</p> <p>Result: The password is cleared.</p>

Resetting a Password

To reset the application password of a Control project, proceed as follows.

Step	Action
1	<p>Right-click the topological entity and select Forgot Password.</p> <p>For an M580 safety controller, select Controller in the submenu.</p> <p>Result: The Forgot Password dialog box opens and an authentication code is displayed.</p> <p>NOTE: If you close the Forgot Password dialog box and open it again, the same authentication code is displayed.</p>
2	<p>Contact your local Schneider Electric service representative and provide the authentication code.</p> <p>Result: You receive a temporary password.</p> <p>NOTE: You can use the temporary password as long as you do not modify the program that is deployed to the controller.</p>
3	<p>Right-click the topological entity and select Manage Password.</p> <p>For an M580 safety controller, select Controller in the submenu.</p> <p>Result: The Manage Password dialog box opens.</p>
4	Follow the procedure to change the password, page 809 and enter the temporary password in the Current Password text box.

Changing the Password of the Control Project Loaded at Simulator Start-Up

You can change the password of the default Control project file or your own file, [page 800](#) that is loaded when the simulator starts if the functionality is enabled.

The folder containing the project file must not be write-protected.

The following table indicates from where you can change the password of the Control project file depending on the location of the controller simulator.

Computer where the controller simulator is running	Software component where you must perform the password change
System server computer.	An engineering client. The IP address of the NIC of the workstation in the Topology Explorer must correspond to the system server computer.
Engineering client computer.	The local engineering client.
Operation client computer.	<ol style="list-style-type: none"> 1. Copy the project file loaded in the controller simulator to the local engineering client computer (if the project file loaded in both simulators is different; otherwise, proceed to step 3). 2. Configure the simulator, page 800 on the local engineering client computer to load this project file at start-up. 3. Change the password of the project file by using the local engineering client. 4. Replace the project file on the operation client or on the other computer.
Other computer where the controller simulator was installed manually.	

To change the password of the Control project file that is loaded when the simulator starts, proceed as follows.

Step	Action
1	Start the controller simulator or restart it.
2	<p>Right-click the workstation that represents the computer on which the controller simulator is running and select Control > Manage Password.</p> <p>If more than one Control Expert service exists, you can select the one to use.</p> <p>Result: The Manage Password dialog box opens.</p>
3	In the Current Password text box, enter the password of the Control project file that is loaded when the simulator starts.
4	Enter the new password and confirm it.
5	Select the check box, page 808.
6	<p>Click OK.</p> <p>Result: The password is changed in Control project file.</p> <p>NOTE: The change does not apply to other simulator instances that are running until you restart them.</p>

Deploying to Redundant Controllers

Overview

This topic describes the aspects that are specific to the deployment of Control Participant projects to redundant controllers.

The deployment commands that are used on standalone configurations are also used for redundant controllers.

A description of these commands is given further in the chapter.

For a description of the sequence of actions that EcoStruxure Process Expert performs on the controllers, refer to the topic describing [redundant controller scenarios](#), page 889.

NOTE: For M580 controllers and BMENOC communication modules, you can only use the main IP address, page 679.

ControllerExist Property

Ensure that the value of the **ControllerExist** property, page 689 of the redundant controller matches the physical configuration of the hardware.

If a mismatch is detected, you cannot deploy.

NOTE: The property also affects [execution commands](#), page 855.

Controller Status for Redundant Controllers

Depending on the command that you select for a redundant controller composed of a primary and a standby engine to which IP addresses *IP* and *IP + 1* are assigned respectively, various dialog boxes open before deploying.

Each of these dialog boxes features the **Status** field that informs you of the detected state of each engine role (A and B). The states can be as follows.

States for M580 controllers	States for Quantum controllers
Run Primary	Run Primary
Run Standby	Run Standby
Wait	Run Offline
Stop	Stop Offline
No Conf (non-configured state)	No Conf (non-configured state)

NOTE: If **ControllerExist** is set to **Primary**, the status of the standby engine is not available.

Deployment Workflows for Redundant Controllers

When you deploy to a redundant controller:

- EcoStruxure Process Expert deploys to the primary and the standby engines unless the **ControllerExist** property of the controller is set to **Primary**. You can view and/or select the IP address for the primary engine (*IP*) and the IP address *IP + 1* is automatically selected for the standby engine.
- If you select to start the engines after deployment, both engines are started.

If deployment to the first engine (*IP* or *IP + 1*, depending on the status of the engine, page 889) does not complete successfully, the entire deployment process is canceled.

If deployment to the second engine (*IP* or *IP + 1*, depending on the status of the engine) or if synchronization does not complete successfully, logic that was deployed successfully to the first engine is retained.

NOTE: For M580 controllers and BMENOC communication modules, you can only use the main IP address, page 679.

Redundant Controller Stop Sequence

If deployment requires stopping the controller and both are running, controllers are stopped in the following order:

- The standby controller
- The primary controller

Deploying Changes to Redundant Controllers

If you deploy changes that do not require stopping the controller by using the **Deploy Changes / Undo Online Changes** command, page 830 and the primary controller is in **Running Primary** state, EcoStruxure Process Expert performs the following actions in sequence:

- It enables the configuration mismatch feature of the controller so that the standby controller does not enter the offline operating mode when changes are deployed to the primary controller.
- It deploys to the primary controller first and upon successful completion of the deployment, it automatically synchronizes the standby controller.
- It disables the configuration mismatch feature.

When you deploy changes made online to a redundant controller and deployment to the primary controller succeeds, the synchronization of the standby controller, page 892 may not complete. In this case, a message appears in the notification panel to inform you. If an interruption of the connection to the system server is the cause, the notification panel also contains additional information about steps that you must complete when the connection to the system server is re-established.

For example, if EcoStruxure Process Expert cannot disable the configuration mismatch feature and a switchover occurs after deployment to the primary controller, the standby controller operates with a program that is not synchronized with the changes that you deployed to the primary controller.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: If the primary controller is not in **Running Primary** state, synchronization of the standby controller cannot complete successfully. In this case, starting, page 861 the primary controller synchronizes the standby one.

Deploying the Control Participant Project

Overview

The **Topology Explorer** allows you to deploy the Control Participant project to a controller or controller simulator that is modeled by an engine in the topology by using the following commands:

- **Deploy Built Project:** Deploys the built Control Participant project to the controller or Control service of the workstation that is mapped to the corresponding executable, page 723.
- **Re-Deploy Last Project:** Re-deploys the built Control Participant project that you last deployed to the controller/workstation. It deploys a mirror of the deployed Participant project, called the last deployed reference, which is stored in the content repository, page 1119.

To deploy or redeploy a built Participant project, the controller or the controller simulator that you are deploying to must be in **Stop** state. Otherwise, EcoStruxure Process Expert informs you of it and stops the engine when you deploy.

You can start the controller following the deployment.

Built Participant Project Build Status

You can deploy a built Control Participant project whose executable is either **Out Of Date** or that you did not re-build after making changes to the system, page 781, which affect the built Participant project.

A message is shown in the confirmation dialog boxes of deployment commands if the **Build State**, page 779 of the associated executable is **Out Of Date**.

Selecting Participant Project and Control Executable

When you select the **Deploy Built Project** command, you can select the following in the **Deploy Built Project** dialog box:

- **Project:** A Control project that exists in the system and whose **Controller family**, page 269 property matches the controller platform.
The project must contain a built or out of date executable that is mapped to the controller.
- **Executable:** For each project, a Control **executable**, page 721 that is mapped to the engine, page 723.

NOTE: When you select **Deploy Built Project** for a workstation, you can select a Control project that has an executable that is mapped to a Control service of this workstation.

Deploying to a Redundant Controller

For information that applies to the deployment to redundant controllers, refer to the corresponding topic, page 812.

Control Participant Project Information

Before deploying or redeploying a built Participant project, information is displayed in the **Confirm Deploy Built Project** dialog box about:

- The engine you are deploying to.
- The built Participant project to be deployed.
- The Participant project deployed last to the engine.

StationNode_2: Confirm Deploy Built Project (Control)

Engine Information

Status: Stopped

IP Address: Slot NIC_1 {127.0.0.1} ▼

Project Information

Name	Selected Project	Last Deployed Project
Executable	ControlExecutable_2	ControlExecutable_2
Controller	StationNode_2	StationNode_2
System	System_4	System_4
Project	ControlProject_1	ControlProject_1
Last Built On	02/05/2017 10:54:00	02/05/2017 10:54:00
Last Deployed On	N/A	02/05/2017 11:00:40
Last Online Change Performed On	N/A	

☐ Start engine after deployment.

Are you sure you want to deploy built project?

OK Cancel

The figure shows an example of the **Confirm Deploy Built Project** dialog box for a redundant controller.

Q_HSBY: Confirm Deploy Built Project

Engine Information

Role: A B

Status: Running Primary Running Standby

IP Address: Slot 05:140NOE77111 {192.168.10.80} Slot 05:140NOE77111 {192.168.10.81}

Project Information

Name	Selected Project	Last Project Deployed To Primary Engine	Last Project Deployed To Standby Engine
System	System_2		
Project	Q_HSBY		
Executable	q		
Controller	Q_HSBY		
Last Built On	13-11-2014 16:15:45		
Last Deployed On	N/A		
Last Online Change Performed On	N/A		
Last Sync Performed On	N/A		

☐ Start primary and standby engines after deploying built project.

Primary and standby engines are running. Are you sure you want to stop both engines to continue with deploy built project?

OK Cancel

The following table describes the information that is displayed in the **Engine Information** section of the **Confirm Deploy Built Project** dialog box.

Label	Description
Role⁽¹⁾	Role of the engine: A or B. If EcoStruxure Process Expert cannot communicate with the engine, the role is indicated as Not Detected .
Status	Status of the engine that is selected for deployment. If the status is Not Reachable you cannot proceed with the deployment. In this case, verify the IP address, the connection to the engine, and the status of the engine. NOTE: If the status is Running , a notification is displayed in the dialog box to inform you that the engine will be stopped if you deploy.

Label	Description
IP Address	<p>Name of the communication module through which the built Participant project is deployed and the IP address that is configured in the Topology Explorer.</p> <p>If the topological entity has several communication modules, you can select one of them.</p> <p>NOTE: For M580 controllers and BMENOC**** communication modules, you can only use the main IP address, page 679.</p> <p>NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.</p>
(1) Displayed only for a redundant controller.	

NOTE: To refresh the engine status and attempt to connect to it, click the reconnect button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Deploy Built Project** dialog box.

Namecolumn	Selected Project column	Last Deployed Project column ⁽²⁾
System	Identifier of the system that the engine belongs to.	Identifier of the system that the engine from which you last deployed the project belongs to.
Project	Identifier of the Control Participant project that you selected.	Identifier of the Control Participant project associated to the executable that you last deployed.
Executable	Identifier of the Control executable that you selected and that contains the built Participant project to deploy.	Identifier of the Control executable that contains the built Participant project that you last deployed.
Controller	Name of the topological entity to which you are deploying the built Participant project.	Name of the topological entity to which you last deployed the built Participant project.
Last Built On	Date and time when you successfully built the Control Participant project to deploy.	Date and time when you successfully built the Control Participant project that you last deployed.
Last Deployed On	Not applicable	Date and time when you last deployed the Participant project to the engine or you last completed the Deploy Changes / Undo Online Changes command, independently if there changes were deployed or not.
Last Online Change Performed On	Not applicable	Date and time when you last changed the deployed Participant project by using the Refine Online command; otherwise the field is empty.
Last Sync. Performed On ⁽¹⁾	Not applicable	Date and time when you last synchronized the standby controller with the primary one by using the Synchronize (Primary→Standby) command.
Start engine after deployment	Select the option to start the engine after deployment is complete; otherwise the engine remains in stop state. NOTE: The engine is stopped before deploying.	
(1) Displayed only for a redundant controller.		
(2) When you deploy to a redundant controller, the dialog box shows separate columns for the primary and the standby engines.		

Deploying the Control Participant Project

To deploy the Control Participant project, proceed as follows.

Step	Action
1	Ensure that the engine can communicate with the computer, page 803 and that it can be stopped.
2	<p>In the Topology Explorer, right-click the entity where you want to deploy the built Participant project and select Deploy Built Project.</p> <p>Result: The Deploy Built Project dialog box opens.</p>

Step	Action
3	Click the menu next to Project and select the Control project.
4	Click the menu next to Executable and select the executable associated to the built Control project that you want to deploy.
5	Click OK . Result: The Confirm Deploy Built Project dialog box opens.
6	Verify the information that is displayed.
7	Verify the status of the engine in the Status field.
8	Verify the communication module identification or select one in the IP Address field. NOTE: When deploying to a controller simulator, ensure that it is started, page 799 and uses the same port number as the one shown in the IP Address field.
9	Select Start engine after deployment to start the engine after the deployment is complete.
10	Click OK . Result: The Reconfirm Deploy Built Project dialog box opens.
11	Click OK . Result: <ul style="list-style-type: none"> The engine is stopped. The Control Participant project is deployed. Complete is displayed in the notification panel once deployment is completed. <ul style="list-style-type: none"> For a redundant controller, the notification panel displays separate messages for the primary and the standby engines by using their respective IP address to identify them. If deployment to either engine does not complete successfully, a message appears in the notification panel. Updates the related date/time field. Starts the engine (cold start) if you selected the option. NOTE: Click Cancel to revert to the Confirm Deploy Built Project dialog box without deploying.

NOTE: If the deployment process does not complete successfully and the controller was stopped, it remains in **Stopped** state.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that you are connected to the controller you intend to deploy the Control Participant project.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For more information, refer to Identifying Engines, page 796.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that you have selected the correct project file before deploying it to a controller.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠ WARNING**UNKNOWN OPERATIONAL STATE OF EQUIPMENT**

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

When you deploy to a redundant controller, the deployment to the standby controller may not complete successfully. In this case, a message is displayed in the notification panel to inform you. For more information, refer to [Redundant Controller Reservation and Connection Scenarios](#), page 892.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For information on the **Sync (Primary→Standby)** command, refer to the [Execution stage](#), page 855.

Re-Deploying the Control Participant Project

Overview

The **Re-Deploy Last Project** command is available once you have deployed a project to the engine.

The process of redeploying the Control Participant project is similar to the deployment process, page 819 except that you cannot select a Control project and an associated Control executable. A mirror image of the last deployed Participant project is deployed to the controller or controller simulator, which is the last deployed reference.

To redeploy a built Participant project, the engine must be in **Stopped** state. Otherwise, a message informs you of it and the engine is stopped when you deploy.

NOTE: If you made changes to the deployed Participant project by using the **Refine Online** and/or if you deployed changes by using the **Deploy Changes** commands, the Participant project that is redeployed to the controller contains these changes.

Redeploying to a Redundant Controller

For information that applies to the redeployment to redundant controllers, refer to the corresponding topic, page 812.

Control Participant Project Information

Before re-deploying the last deployed reference, information is displayed in the **Confirm Re-Deploy Last Project** dialog box about:

- The engine you are re-deploying to.
- The project to be re-deployed.
- The Participant project deployed last to the engine.

StationNode_1: Confirm Re-Deploy Last Project

Engine Information

Status: Running

IP Address: Slot NIC_1 {127.0.0.1} ▼

Project Information

Name	Selected Project	Last Deployed Prc
System	System_1	System_1
Project	Quantum_1	Quantum_1
Executable	ControlExecutable_2	ControlExecutable_2
Controller	StationNode_1	StationNode_1
Last Built On	13-11-2014 09:53:45	13-11-2014 09:53:45
Last Deployed On	N/A	13-11-2014 11:59:31
Last Online Change Performed On	N/A	

☒ Start engine after deployment

Engine is already running. Are you sure you want to stop the engine to continue with Deploy Built Project?

OK Cancel

The figure shows an example of the **Confirm Re-Deploy Last Project** dialog box for a redundant controller.

Q_HSBY: Confirm Re-Deploy Last Project


Engine Information

Role	B	A
Status	Running Primary	Running Standby
IP Address	Slot 05:140NOE77111 {192.168.10.80}	Slot 05:140NOE77111 {192.168.10.81}

Project Information

Name	Selected Project	Last Project Deployed To Primary Engine	Last Project Deployed To Standby Engine
System	System_2	System_2	System_2
Project	Q_HSBY	Q_HSBY	Q_HSBY
Executable	q	q	q
Controller	Q_HSBY	Q_HSBY	Q_HSBY
Last Built On	13-11-2014 16:15:45	13-11-2014 16:15:45	13-11-2014 16:15:45
Last Deployed On	N/A	13-11-2014 16:27:57	13-11-2014 16:27:57
Last Online Change Performed On	N/A		
Last Sync Performed On	N/A	13-11-2014 16:27:57	N/A

☐ Start primary and standby engines after deploying built project.

 Primary and standby engines are running. Are you sure you want to stop both engines to continue with deploy built project?

OK Cancel

The following table describes the information that is displayed in the **Engine Information** section of the **Confirm Re-Deploy Last Project** dialog box.

Label	Description
Role⁽¹⁾	Role of the engine: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the engine, the role is indicated as Not Detected .
Status	Status of the engine that is selected for deployment. If the status is Not Reachable you cannot deploy. In this case, verify the IP address, the connection to the engine, and the status of the engine. NOTE: If the status is Running , a notification is displayed in the dialog box to inform you that the engine will be stopped if you deploy.
IP Address	Name of the communication module through which the built Participant project is redeployed and the IP address that is configured in the Topology Explorer . If the topological entity has several communication modules, you can select one of them. Default value: IP address that was selected for the last deployment operation to this engine. NOTE: For M580 controllers and BMENOC.... communication modules, you can only use the main IP address, page 679. NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.
(1) Displayed only for a redundant controller.	

NOTE: To refresh the engine status and attempt to connect to it, click the reconnect button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Re-Deploy Built Project** dialog box.

Name column	Selected Project column	Last Deployed Project column ⁽²⁾
System⁽¹⁾	Identifier of the system that the engine belongs to.	
Project⁽¹⁾	Identifier of the Control Project that you selected.	
Executable⁽¹⁾	Identifier of the Control executable that you selected and that contains the built Participant project to deploy.	
Controller⁽¹⁾	Name of the controller/workstation to which you are deploying the built Participant project.	
Last Built On	Date and time when you successfully built the Control Participant project to deploy.	Date and time when you successfully built the Control Participant project that you last deployed.
Last Deployed On	Not applicable	Date and time when you last deployed the Participant project to the engine or you last completed the Deploy Changes / Undo Online Changes command, independently if changes were deployed or not.
Last Online Change Performed On	Not applicable	Date and time when you last changed the deployed Participant project by using the Refine Online command; otherwise the field is empty.
Last Sync. Performed On⁽³⁾	Date and time when you last synchronized the standby controller with the primary one by using the Synchronize (Primary→Standby) command.	


Namecolumn	Selected Project column	Last Deployed Project column ⁽²⁾
Start engine after deployment	Select the option to start the engine after the deployment is complete; otherwise the engine remains in stop state. NOTE: The engine is stopped before deploying.	
<p>(1) The values of the Selected Project and Last Deployed Project fields are identical.</p> <p>(2) When you redeploy to a redundant controller, the dialog box shows separate columns for the primary and the standby engines.</p> <p>(3) Displayed only for a redundant controller.</p>		

Redeploying the Last Deployed Reference

To redeploy the last deployed reference, proceed as follows.

Step	Action
1	Ensure that the engine can communicate with the computer, page 803 and that it can be stopped. NOTE: When redeploying to a controller simulator, ensure that it is started, page 799 and uses the same port number as the one shown in the IP Address field.
2	In the Topology Explorer , right-click the entity where you want to redeploy the built Participant project and select Re-Deploy Last Project . Result: The Confirm Re-Deploy Last Project dialog box opens.
3	Verify the information that is displayed.
4	Verify the status of the engine in the Status field.
5	Select Start engine after deployment to start the engine after the redeployment is complete.
6	Click OK . The Reconfirm Re-Deploy Last Project dialog box opens.
7	Click OK . Result: <ul style="list-style-type: none"> The last deployed reference is redeployed. Complete is displayed in the notification panel once redeployment is completed. <ul style="list-style-type: none"> For a redundant controller, the notification panel displays separate messages for the primary and the standby engines by using their respective IP address to identify them. If redeployment to either engine does not complete successfully, a message appears in the notification panel. The related date/time field is updated. The engine is started if you selected the option. NOTE: Click Cancel to revert to the Confirm Re-Deploy Last Project dialog box without re-deploying.

NOTE: If the redeployment process does not complete successfully and the controller was stopped, it remains in **Stopped** state.

 WARNING
<p>UNINTENDED EQUIPMENT OPERATION</p> <p>Ensure that you are connected to the controller you intend to deploy the Control Participant project.</p> <p>Failure to follow these instructions can result in death, serious injury, or equipment damage.</p>

NOTE: For more information, refer to *Identifying Engines*, page 796.

⚠ WARNING**UNKNOWN OPERATIONAL STATE OF EQUIPMENT**

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

When you redeploy to a redundant controller, the redeployment to the standby controller may not complete successfully. In this case, a message is displayed in the notification panel to inform you. For more information, refer to [Redundant Controller Reservation and Connection Scenarios](#), page 892.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For information on the **Sync (Primary→Standby)** command, refer to the [Execution stage](#), page 855.

Deploying Data to Controllers and Controller Simulators

Overview

The **Deploy Data** command lets you deploy data contained in a backup file, page 880 that was created by using the **Back Up Data** command.

You can select one file from a list of backup files that were created for the executables of the same Control Participant project. The selection dialog shows the information that was associated to each backup file at the time of creation, including the type of data that was selected.

By default, backup files are listed in descending order of time stamp (most recent file on top).

NOTE: For more information, refer to *Save/Restore Data Between a File and the PLC* in the help of the Control Participant, page 95.

Deploying Data to Redundant Controllers

For redundant controllers, you can deploy data only to the primary controller.

To deploy data to the primary controller, the standby controller must be reachable at the configured IP address.

Only if the **ControllerExist** property, page 689 is set to **Primary**, you can deploy data to the primary controller while the standby controller is not reachable.

NOTE: The status of the standby controller is not shown in the **Confirm Deploy Data** dialog box.

Data Backup File Information

Before deploying controller data, information is displayed in the **Confirm Deploy Data** dialog box about:

- The deployed Control Participant project.
- The executable that is deployed.
- Executables of this Control project for which a backup file exists.
- The type of data contained in each backup file.
- The backup file identification.

NOTE: If you change the name of a controller or executable, or the role of a controller, the identification information of the corresponding backup file is not modified.

You can sort and filter, page 120 the information that is displayed.

The following figure shows an example of the **Confirm Deploy Data** dialog box for a standalone controller.

StationNode_3: Confirm Deploy Data (Control)

Engine Information

Status: Stopped
IP Address: Slot NIC_1 {127.0.0.1}

Project Information

Name	Last Deployed Project
Control Project	ControlProject_1
Executable	ControlExecutable_3
Last Deployed On	12/05/2017 11:46:50

Select Backup Data

Time Stamp	Description	User	Executable	Controller	%M	%MW	Unlocated Variables	Unlocated FBs
<input checked="" type="radio"/> 20170512121021	Test_1 data	Pascal LAVALLE...	ControlExecutable_4	StationNode_4	0 - 512	0 - 2048	False	False
<input type="radio"/> 20170512120738	Initial data	Pascal LAVALLE...	ControlExecutable_3	StationNode_3	0 - 512	0 - 2048	True	True

⚠ Controller is in STOP state. Deploying data may change the program data and cause unexpected equipment operation when the controller goes to RUN state.

OK Cancel

The following figure shows an example of the **Confirm Deploy Data** dialog box for a redundant controller.

R_04_1: Confirm Deploy Data

Engine Information

Controller: Primary
Role: A
Status: Running Primary
IP Address: Slot 0:BMEH584040 {173.20.4.10}

Project Information

Name	Last Project Deployed To Primary Engine
Control Project	ControlProject_1
Executable	ControlExecutable_1
Last Deployed On	5/12/2017 10:28:31 AM

Select Backup Data

Time Stamp	Description	User	Executable	Controller	%M	%MW	Unlocated Variables	Unlocated FBs
<input checked="" type="radio"/> 20170512103316	BUP_1	Damien HUGOT...	ControlExecutable_1	R_04_1 \A	0 - 512	0 - 6000	True	True

⚠ Controller is in RUN state. Deploying data when the controller is in this state may take more than one scan cycle. This operation may corrupt the program data and cause unexpected equipment operation. It is strongly recommended to deploy data when the controller is in STOP state.

OK Cancel

The following table describes the information that is displayed in the **Engine Information** section of the **Deploy Data** dialog box.

Label	Description
Controller⁽¹⁾	You can deploy controller data only to the primary controller. Default value: Primary
Role⁽¹⁾	Role of the engine: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the engine, the role is indicated as Not Detected .
Status	Status of the engine that is selected for deployment of data. If the status is Not Reachable , you cannot deploy. In this case, verify the IP address, the connection to the engine, and the status of the engine.
IP Address	Name of the communication module through which data is deployed and the IP address that is configured in the Topology Explorer . If the topological entity has several communication modules, you can select one of them. Default value: IP address that was selected for the last deployment operation to this engine. NOTE: For M580 controllers and BMENOC.... communication modules, you can only use the main IP address, page 679. NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.
(1) Displayed only for a redundant controller.	

NOTE: To refresh the engine status and attempt to connect to it, click the **reconnect** button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Deploy Data** dialog box.

Label	Description
Control Project	Identifier of the Control Participant project associated to the executable that you last deployed.
Executable	Identifier of the Control executable to which you are deploying data.
Last Deployed On	Date and time when you last deployed the Participant project to the engine or you last completed the Deploy Changes / Undo Online Changes command, independently if changes were deployed or not.

The following table describes the information that is displayed in the **Select Backup Data** section of the **Confirm Deploy Data** dialog box.

Column header	Description
Time Stamp	Date and time of the backup file creation.
Description	Description that was entered when the backup file was created.
User	Name of the user who was logged into the client when the backup file was created.
Executable	Identifier of the Control executable whose data is contained in the backup file.
Controller	Name of the controller that was running the executable whose data was backed up. For redundant controllers, it also indicates the role of the controller that was selected when the backup file was created.
%M	Indicates the memory range of located variables of boolean data type that are contained in the backup file. If this type of data was not selected, the value shown is N/A .
%MW	Indicates the memory range of located variables of WORD data type that are contained in the backup file. If this type of data was not selected, the value shown is N/A .
Unlocated Variables	True indicates that this type of data was selected during data backup. Possible values: <ul style="list-style-type: none"> • True • False
Unlocated FBs	

Control Project Signature Verification

Before executing the command, EcoStruxure Process Expert compares the Control Participant project that is deployed in the engine with the *last deployed reference*, page 795, which is the Control Participant project residing in EcoStruxure Process Expert content repository.

If they are identical, the command is executed. Otherwise, a recovery process starts, which may take several minutes to complete.

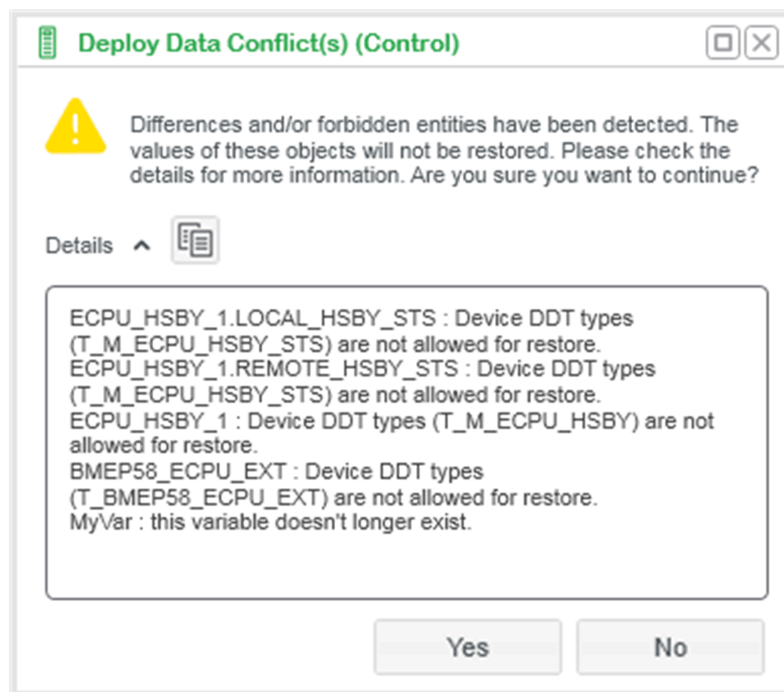
If a notification about a signature mismatch is displayed, the command cannot be executed. Contact your local Schneider Electric service representative.

Deployed Data Conflicts

If conflicts in the data to be deployed are detected, a dialog box opens before the deployment of data. It displays the data instances that cannot be deployed.

You can deploy without the conflicting data instances or cancel the deployment operation.

The following figure shows an example of the **Deploy Data Conflicts** dialog box.



Deploying Data Contained in a Backup File

Unintended situations can occur if the operations that are in progress are not known when acting on a controller.

Deploying data when the controller is in **RUN** state may take several scan cycles and corrupt the program data. This adversely impacts the execution of the program.

Deploying data to the controller may change program data. This affects the execution of the program when the controller is in **STOP** state and then switched to **RUN** state after the deployment.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Before deploying data to the controller:

- Ensure that the controller state does not jeopardize data integrity when the data is deployed.
- Verify the impact of the deployment on the execution of the program.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: Deploy data while the controller is in **STOP** state.

To deploy data contained in a backup file, proceed as follows.

Step	Action
1	Ensure that the engine can communicate with the computer, page 803. NOTE: When deploying data to a controller simulator, ensure that it is started, page 799 and uses the same port number as the one shown in the IP Address field.
2	In the Topology Explorer , right-click the entity where you want to deploy data and select Deploy Data . Result: The Confirm Deploy Data dialog box opens.
3	In the IP Address field, ensure that the communication module identification is correct.
4	Verify the status of the engine in the Status field. NOTE: Deployment is possible only if the state of the engine is indicated as either RUN , STOP , or WAIT (OFFLINE) for Hot Standby controllers of the Quantum platform).
5	Verify the information that is displayed in the Project Information section.
6	Select a backup file.
7	Click OK . Result: The Reconfirm Deploy Data dialog box opens.
8	Click OK . Result: If conflicts in the data to be deployed are detected, the Deploy Data Conflicts dialog box, page 827 opens and you can deploy without the conflicting data (Yes) or cancel the deployment (No). Otherwise, the data is deployed and information about the status of the deployment operation is provided. NOTE: Click Cancel to close the Reconfirm Deploy Data dialog box without deploying data.

Deploying Control Project Changes

Overview

Following a deployment, the **Deploy Changes / Undo Online Changes** command allows you to deploy changes that you made to the associated Control Participant project.

It lets you also deploy Configuration Change On The Fly (CCOTF)-compatible changes, page 585 made to the configuration of the associated controller.

You can deploy changes pertaining to the following categories:

- Animation tables
- Operator screens
- Sections
- Types
- Variables
- Hardware

In addition, for Control projects of M580 safety controllers, page 908, you can deploy changes related to:

- Process (not safety-related) program interfaces
- The Safe program
- The functional safety password

To deploy other changes, refer to the topic describing changes that cannot be deployed, page 834.

When you select the **Deploy changes / Undo Online Changes** command, EcoStruxure Process Expert compares the Participant project that is deployed in the controller with the corresponding built Participant project to identify the following:

- Control project and configuration changes that you can deploy without stopping the controller.
- Control project changes that require stopping the controller, page 840. A dialog box opens informing you that a controller stop is required in order to deploy these changes.

You can select which of the identified changes you want to deploy.

The deployment of changes executes the CCOTF transaction and resets the counter.

You can proceed with a cold start of the controller following the deployment of changes that require stopping the controller.

NOTE: You can use the **Deploy changes / Undo Online Changes** command to discard changes that you made to the deployed Participant project.

NOTE: You must build the Control project, page 781 before you can deploy changes by using the **Deploy changes / Undo Online Changes** command.

Control Project Signature Verification

Before deploying changes, EcoStruxure Process Expert compares the Control Participant project that is deployed in the controller with the last deployed reference, page 795, which is the Control Participant project residing in EcoStruxure Process Expert content repository.

If they are identical, the command is executed. Otherwise, a recovery process starts, which may take several minutes to complete.

If a notification about a signature mismatch is displayed, the command cannot be executed. Contact your local Schneider Electric service representative.

Undoing Online Changes

If you changed the logic of the deployed Participant project by using the **Refine Online** command, you can use **Deploy changes / Undo Online Changes** to undo these changes. This is possible only if the following conditions are fulfilled:

- You did not change the configuration online.
- You did not update the Control project with these online changes.

EcoStruxure Process Expert identifies the changes you made online and displays them in the **Deploy Changes / Undo Online Changes** dialog box. After executing the command, the deployed Control Participant project is identical to the built Participant project.

NOTE: If you changed and built the Control Participant project since the deployment, these changes are applied to the deployed Participant project unless you unselect them. It is not possible to unselect CCOTF-compatible hardware changes.

Deploying Changes to a Redundant Controller

For information that applies to the deployment to redundant controllers, refer to the corresponding topic, page 812.

Deployed Project Information

When you select the **Deploy Changes / Undo Online Changes** command, the **Confirm Deploy Changes / Undo Online Changes** dialog box opens, which contains information about:

- The controller you are deploying changes to.
- The Control Participant project that is deployed to the controller.

StationNode_1: Confirm Deploy Changes / Undo Online Changes

Engine Information

Status: Running

IP Address: Slot NIC_1 {127.0.0.1} ▼

Project Information

Name	Selected Project	Last Deployed Prc
System	System_1	System_1
Project	Quantum_1	Quantum_1
Executable	ControlExecutable_2	ControlExecutable_2
Controller	StationNode_1	StationNode_1
Last Built On	13-11-2014 09:53:45	13-11-2014 09:53:45
Last Deployed On	N/A	13-11-2014 11:59:31
Last Online Change Performed On	N/A	

Are you sure you want to Deploy Changes / Undo Online Changes?

OK Cancel

The figure shows an example of the **Confirm deploy changes / Undo Online Changes** dialog box for a redundant controller.

Q_HSBY: Confirm Deploy Changes / Undo Online Changes

Engine Information

Role	B	A
Status	Running Primary	Running Standby
IP Address	Slot 05:140NOE77111 {192.168.10.80}	Slot 05:140NOE77111 {192.168.10.81}

Project Information

Name	Selected Project	Last Project Deployed To Primary Engine	Last Project Deployed To Standby Engine
System	System_2	System_2	System_2
Project	Q_HSBY	Q_HSBY	Q_HSBY
Executable	q	q	q
Controller	Q_HSBY	Q_HSBY	Q_HSBY
Last Built On	13-11-2014 16:15:45	13-11-2014 16:15:45	13-11-2014 16:15:45
Last Deployed On	N/A	13-11-2014 16:27:57	13-11-2014 16:27:57
Last Online Change Performed On	N/A		
Last Sync Performed On	N/A	13-11-2014 16:27:57	N/A

⚠ Are you sure you want to Deploy Changes / Undo Online Changes to primary and standby engines?

OK Cancel

The following table describes the information that is displayed in the **Engine Information** section of the **Confirm deploy changes / Undo Online Changes** dialog box.

Label	Description
Role⁽¹⁾	Role of the controller: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the controller, the role is indicated as Not Detected .
Status	Status of the controller that is selected. If the status is Not Reachable you cannot deploy. In this case, verify the IP address, the connection to the controller, and the status of the controller. NOTE: If the status is Running , a notification is displayed in the dialog box to inform you that the controller will be stopped if you deploy.
IP Address	Name of the communication module through which the built Participant project is deployed and the IP address that is configured in the Topology Explorer . If the topological entity has several communication modules, you can select one of them. Default value: IP address that was selected for the last deployment operation to this controller. NOTE: For M580 controllers and BMENOC... communication modules, you can only use the main IP address, page 679. NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.
(1) Displayed only for a redundant controller.	

NOTE: To refresh the controller status and attempt to connect to it, click the reconnect button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Deploy Changes / Undo Online Changes** dialog box.

Name column	Selected Project column	Last Deployed Project column ⁽²⁾
System	Identifier of the system that the controller belongs to.	Identifier of the system that the controller from which you last deployed the Participant project belongs to.
Project	Identifier of the Control Project that you selected.	Identifier of the Control Project associated to the deployed Participant project that you last deployed.
Executable	Identifier of the Control executable that you selected and that contains the built Participant project to deploy.	Identifier of the Control executable that contains the built Participant project that you last deployed.
Controller	Name of the controller/workstation to which you are deploying the built Participant project.	Name of the controller/workstation to which you last deployed the built Participant project.
Last Built On	Date and time when you successfully built the Control Participant project to deploy.	Date and time when you successfully built the Control Participant project that you last deployed.
Last Deployed On	Not applicable	Date and time when you last deployed the Participant project to the controller or you last completed the Deploy Changes / Undo Online Changes command, independently if changes were deployed or not.
Last Online Change Performed On	Not applicable	Date and time when you last changed the deployed Participant project by using the Refine Online command; otherwise the field is empty.
Last Sync. Performed On ⁽¹⁾	Date and time when you last synchronized the standby controller with the primary one by using the Synchronize (Primary→Standby) command.	
(1) Displayed only for a redundant controller.		
(2) When you deploy changes to a redundant controller, the dialog box shows separate columns for the primary and the standby controller.		

Deploying Changes Other Than FBD Language

The following table describes the effect of the **Deploy changes / Undo Online Changes** command on items other than Function Block Diagram (FBD) language.

Item	Effect
Non-FBD sections	Details of changes inside non-FBD sections that were identified are not shown. Only the names of sections that were added, removed, or modified are displayed. Nevertheless, the changes inside non-FBD sections are applied to the Control Participant project.
Animation tables	Details of changes to these items that were identified are not shown. Only the name of the animation table and/or operator screen is displayed.
Operator screens	<p>Animation tables and/or operator screens that the built Participant project contains, including modified and/or new ones are shown. Animation tables and/or operator screens that you deleted from the Participant project are not shown.</p> <p>You cannot select individual items but only the category.</p> <p>When selected, animation tables and operator screens of the deployed Participant project are overwritten with those contained in the built Participant Project.</p> <p>By default, both categories are unselected.</p> <p>NOTE: Overwriting an existing operator screen creates a copy of the screen in the deployed Participant project. Following deployment, use the Refine Online command to edit the categories and make their content consistent with the Control Participant project.</p> <p>NOTE: Animation tables and/or operator screens that you removed from the Participant project are not removed from the deployed Participant project by using the command. Use the Refine Online command to delete them.</p>
Hardware	<p>Details of CCOTF- compatible configuration changes are shown.</p> <p>The Hardware category is selected by default (if changes are detected) and is read-only.</p>

Considerations When Deploying Hardware Mapping Changes

The information that is provided in the following pages applies when you use the **Deploy changes / Undo Online Changes** to deploy changes to the hardware mapping that affect the following:

- IP addresses, page 948
- Signals, page 753

Changes That Cannot Be Deployed

EcoStruxure Process Expert only identifies and deploys changes that pertain to a category displayed in the **Name** column of the **Deploy Changes / Undo Online Changes** dialog box. Refer also to the section describing the deployment of changes other than FBD language, page 833.

WARNING

UNINTENDED EQUIPMENT OPERATION

Perform **Deploy Built Project** operation to apply changes that are not identified in the **Deploy Changes / Undo Online Changes** dialog box.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

The following table indicates the types of changes that are not identified and the command that you must use to apply them to the deployed Control Participant project.

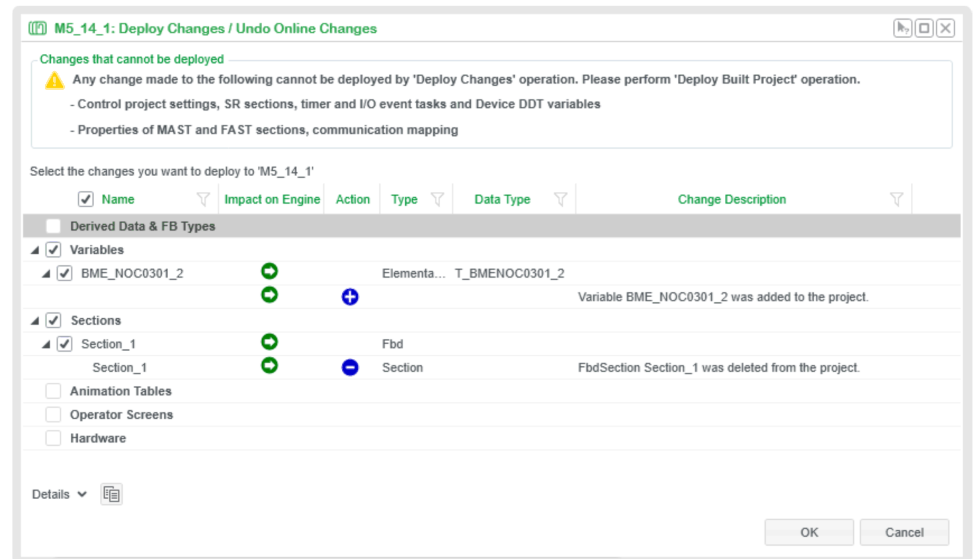
Type of change	Command to use
Control project settings (refer to the Control Participant help, <i>Operating Modes</i>)	Refine Online , page 874 or Deploy Built Project , page 814
SR sections	
Properties of MAST and FAST tasks	
Timer and I/O event tasks	
I/O Derived Data Type (DDT) variables	Refine Online , page 874
Device DDT variables	Refer to the topic describing the deployment of changes related to device DDT variables, page 836
Changes to the topology, page 562 made in the Topology Explorer and that are not CCOTF-compatible.	Deploy Built Project , page 814
Changes to the communication mapping, page 725.	

Identifying Project and Hardware Changes

Before deploying, the **Deploy Changes / Undo Online Changes** dialog box opens, which indicates the differences that it has identified between the built Participant project and the deployed Participant project. The dialog box allows you to select the changes that you want to deploy.

NOTE: If you select a section in the deployed Control Participant project, EcoStruxure Process Expert overrides the online changes that you made in this section.

The following figure shows an example of the **Deploy Changes / Undo Online Changes** dialog box.



Header	Description
Name	<p>Identifier of the items that are changed.</p> <p>The items that are selected are those that were identified as changed and that will be deployed.</p> <p>Items are grouped by categories of the Control Participant project.</p> <p>NOTE: A black square in a selection box indicates that at least one sub-item is unselected.</p>
Impact on Engine	<p>Indicates if the deployment of the item requires stopping the controller, page 840.</p> <ul style="list-style-type: none"> Green dot with an arrow: You can deploy the item without stopping the controller. Red dot with a square: The controller must be stopped to deploy the item.
Action	<ul style="list-style-type: none"> Minus icon: The item will be removed from the deployed Participant project. (Because the item was removed from the built Participant project or it was added to the deployed Participant project but not updated in the Participant project.) Plus icon: The item will be added to the deployed Participant project. (Because the item was added to the built Participant project or it was removed from the deployed Participant project during online refinement and the change was not updated in the Participant project.) Pencil icon: The item will be modified in the deployed Participant project. <p>NOTE: If you select a section, the changes that were identified for this section are deployed. You cannot select/unselect changes individually inside a section.</p>
Type	Indicates the type of the item identified as a change.
Data Type	Indicates the data type that is affected by the change, if applicable.
Description of Change	For FBD sections, variables, types, and hardware, indicates whether the change that was identified is an addition, a modification, or a deletion compared to the deployed Participant project.

NOTE: A notification is displayed if no change that can be deployed is detected.

Deploying Changes Related to Device DDT Variable Creation, Update, or Deletion

The following table describes the information that is displayed if a device DDT variable, page 455 is present in either the built or the deployed Control project.

Device DDT variable in the deployed Control project	Device DDT variable in the built Control project	Information shown in the Deploy Changes / Undo Online Changes dialog box
Mapped/unmapped	Not present	No information. Use the Refine Online , page 874 or Deploy Built Project , page 814 command to remove the variable in the deployed Control project
Not present	Mapped/unmapped	EcoStruxure Process Expert detects the following as candidate for deployment: <ul style="list-style-type: none"> • The device DDT variable • Logic related to the variable • Sections containing the logic NOTE: The variable is unmapped in the deployed Control project.

The following table describes the information that is displayed if the following applies:

- A device DDT variable with the same name exists in both the built and the deployed Control project but with a different data type or type version.
- The built Control project contains updated code that uses the device DDT variable.

Device DDT variable in the deployed Control project	Device DDT variable in the built Control project	Information shown in the Deploy Changes / Undo Online Changes dialog box
Mapped	Mapped/unmapped	The device DDT variable is not shown as candidate for deployment. Modifications to the DDT variable type are not deployed. Only the changes in the code are shown as candidate for deployment. NOTE: A notification is displayed in the Details section of the dialog box. NOTE: Deployment of changes may not succeed because not all changes are deployed.
Unmapped	Unmapped	EcoStruxure Process Expert detects the following as candidate for deployment: <ul style="list-style-type: none"> • The device DDT variable • Logic related to the variable • Sections containing the logic

The following table describes the information that is displayed if the following applies:

- A device DDT variable with the same name exists in both the built and the deployed Control project but with a different data type or type version.
- The device DDT variable is not used in the code of either Control project.

Device DDT variable in the deployed Control project	Device DDT variable in the built Control project	Information shown in the Deploy Changes / Undo Online Changes dialog box
Mapped	Mapped/unmapped	The device DDT variable is not shown as candidate for deployment. modifications to the DDT variable type are not deployed. NOTE: A notification is displayed in the Details section of the dialog box.
Unmapped	Unmapped	EcoStruxure Process Expert detects the device DDT variable as candidate for deployment.

The following table describes the information that is displayed if the following applies:

- A device DDT variable that exists in the deployed Control project was deleted from the built Control project.
- The device DDT variable is either:
 - Used in the code.
 - Not used in the code.

Device DDT variable in the deployed Control project	Device DDT variable in the built Control project	Information shown in the Deploy Changes / Undo Online Changes dialog box
Mapped	Variable is deleted and references to it in the code are removed.	The device DDT variable is not shown as candidate for deployment (removal). Only the changes in the code are shown as candidate for deployment.
	Variable is deleted. It is not used in the code.	no information
Unmapped	Variable is deleted and references to it in the code are removed.	EcoStruxure Process Expert detects the following as candidate for deployment: <ul style="list-style-type: none"> • The device DDT variable (removal) • Logic related to the variable • Sections containing the logic
	Variable is deleted. It is not used in code.	EcoStruxure Process Expert detects the device DDT variable as candidate for deployment (removal).

Deploying Changes

Before deploying a CCOTF-compatible modification, refer to the topics describing the CCOTF functionality in the help of the Control Participant for your specific controller platform and follow the instructions.

To deploy changes, proceed as follows.

Step	Action	Result
1	Ensure that the computer is connected, page 803 to the controller. NOTE: To deploy changes to a controller simulator, ensure that it is started, page 799 and uses the same port number as the one shown in the IP Address field.	—
2	In the Topology Explorer , right-click the controller to which you want to deploy changes and select Deploy Changes / Undo Online Changes .	The Confirm Deploy Changes / Undo Online Changes dialog box opens.
3	Verify the information that is displayed.	—
4	Click OK .	The Deploy Changes / Undo Online Changes dialog box opens.
5	Select the changes that you want to deploy.	—
6	Click OK . NOTE: Click Cancel to revert to the previous Deploy Changes / Undo Online Changes dialog box.	The Deploy Changes / Undo Online Changes confirmation dialog box opens.
7	If the deployment requires stopping the controller, in the Deploy Changes / Undo Online Changes confirmation dialog box, select Start engine after deploying built project to start the controller after the deployment of changes is complete.	—
8	Click OK or Deploy (only either button is displayed depending on the deployment use case.) NOTE: Click Cancel to revert to the selection of changes dialog box without deploying changes.	<ul style="list-style-type: none"> • The controller is stopped (if applicable). • The changes are deployed to the deployed Control Participant project. • Completed is displayed in the notification panel once the deployment of changes is successfully executed. <ul style="list-style-type: none"> ◦ For a redundant controller, the notification panel displays separate messages for the primary and the standby controllers by using their respective IP address to identify them. If deployment to either controller does not complete successfully, a message is displayed in the notification panel. • The related date/time field is updated. • The controller is started (cold start) if you selected the option (if applicable).

NOTE: If the deployment of changes does not complete successfully and the controller was stopped, it remains in **Stopped** state.

WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that you are connected to the controller you intend to deploy the Control Participant project.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For more information, refer to *Identifying controllers*, page 796.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

Ensure that you have selected the correct project file before deploying it to a controller.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠ WARNING**UNKNOWN OPERATIONAL STATE OF EQUIPMENT**

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

When you deploy changes to a redundant controller, the synchronization of the standby controller may not complete successfully. In this case, a message is displayed in the notification panel to inform you. For more information, refer to *Redundant Controller Reservation and Connection Scenarios*, page 892. If an interruption of the connection to the system server is the cause, the notification panel also contains additional information about steps that you must complete when the connection to the system server is re-established.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For information on the deployment of changes to redundant controllers, refer to the topic describing *deployment to redundant controllers*, page 813.

Changes That Require Stopping the Controller

Summary of Changes

The following table indicates if changes that you deploy by using the **Deploy Changes / Undo Online Changes** command or apply by using the **Refine Online**, page 874 command require stopping the controller.

Type of change	Requires stopping the controller	
	When using the Deploy Changes command	When using the Refine Online command
General		
Modifying Control Expert project settings.	N/A	Refer to <i>Project Settings</i> for a list of settings that you can change and to <i>Synthesis of Authorized and Unauthorized Modifications</i> in the help of the Control Participant.
Configuration		
Adding a drop with a BMXCRA...../BMXCRA..... adapter module to an M580 controller.	N/A	Yes. Refer to Using Configuration Change On The Fly (CCOTF), page 585.
Program: Task Sections, SR, SFC Action & Transition		
Modifying/replacing partially the code of an SFC section (SFC chart, transition, action).	Yes	Refer to <i>Synthesis of Authorized and Unauthorized Modifications</i> in the help of the Control Participant.
Deleting SFC chart, transition, action.	Yes	
Adding an SFC section.	No	
Modifying or deleting an LL984 section.	Yes	
Adding an LL984 section.	No	
Modifying, deleting, or adding a section in either of the following languages: <ul style="list-style-type: none">ILLDST	No	
Used Variables (also used in Animation Tables or Operator Screens)		
Modifying the <i>Retain</i> or <i>Exchange On STBY</i> attribute of a used variable.	Yes	Yes
Modifying the time stamp attribute of a used variable.	Yes	Refer to <i>Synthesis of Authorized and Unauthorized Modifications</i> in the help of the Control Participant.
Modifying the structure of a used DDT.	Yes	
Used DFBs		
Modifying the name of an input, output, or input/output pin (formal parameter) of a used DFB.	Yes	Refer to <i>Synthesis of Authorized and Unauthorized Modifications</i> in the help of the Control Participant.
Adding an input, output, or input/output pin (formal parameter) to a used DFB.	Yes	
Deleting an input, output, or input/output pin (formal parameter) from a used DFB.	Yes	

Supervision Project Deployment Stage

Overview

This section describes the methods that you can use to deploy built Supervision Participant projects to computers (for example, operation servers, operator stations), which are connected to the logical network. These computers are modeled by the corresponding workstations in the topology of the system.

It also describes how to deploy only the changes that you have made following a first deployment.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Preparing for Deployment

Overview

You can deploy a built Supervision Participant project and related files to the workstations, [page 597](#) that you have defined in the topology of the system and mapped, [page 767](#) to the executable that is associated to the Supervision project.

You can deploy either:

- In the **Project Explorer**: From the executable of the Supervision project. This deploys a selection of files to the workstations that are mapped to the executable in the service mapping.
- In the **Topology Explorer**: From a workstation that is mapped to the executable in the service mapping. This deploys a selection of files to this workstation only.

EcoStruxure Process Expert detects roles of workstations depending on the services, [page 598](#) that are configured in the **Topology Explorer** and processes a workstation as:

- An I/O server when configured with Supervision and **OFS** services.
- An alarm, trend, or report server, or a Supervision client when configured with the Supervision service only.

The deployment is performed to the computer that has an IP address matching the IP address of the workstation configured in the **Topology Explorer**.

For information on the required Supervision software, refer to the topic describing software requirements (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

Target Computer Setup

To prepare for the deployment, you must:

- Ensure that the configuration of the workstations in the topology of the system matches the configuration of the target computers (for example, ensure that the workstation containing the **OFS** service has the same IP address as the computer that runs the OPC Factory Server or OPC UA Server Expert).

If a workstation is configured with several NICs, each having an IP address:

- When deploying from the executable, the first IP address that is reachable from the system server computer is used.
- When deploying from the workstation, you can select which IP address to use among those that are reachable from the system server computer.
- Create shared folders on each computer to which you deploy the Supervision project.
- Ensure that the system server can access these shared folders; otherwise, the deployment cannot complete successfully.

For information on how to set up target computers, refer to the topic describing how to prepare for deployment (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).

NOTE: When sharing a folder, restrict access by giving permissions only to the user who needs to access the folder. In this case, the user who is logged into the system server computer.

Deploying the Built Supervision Participant Project From the Executable

Overview

By using the **Deploy Built Project** command, you can transfer selected files of the built Supervision Participant project to the operation server, and/or operator stations that are connected to the logical network. The deployment is performed to the workstations that you have mapped to the executable of the Supervision project in the service mapping and which represent these computers.

After a first deployment, you must restore the deployed Supervision Participant project on each target computer.

Information related to subsequent deployments from the same executable is provided in the topic describing how to deploy the built Supervision Participant project again, page 844.

Deployed Files

You can deploy the following files:

- The Supervision master project file (.ctz), for example, *Supervision_1_Executable_1.ctz*, which includes the Supervision computer setup file (*citect.ini*) (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).
- Attached user files, page 511. These are located inside the master project (.ctz) on the target computer.
- OFSOPC/OPC UA configuration files (.xml), page 788: Deployed only to workstations mapped to the *IO Server* service of the Supervision project. If there are several configuration files, each file is deployed to its respective workstation only.
- The included projects, page 510.

You can select not to deploy the configuration file and/or the included project in the **Deployment File Selection** dialog box, page 848, which is accessible during the deployment process.

NOTE: Supervision project files that are deployed are already compiled by using the Supervision Participant. To verify the version, open a Supervision Participant window (for example, by refining a Supervision project), and open the **About** dialog box.

Deploying Out Of Date Participant Projects

You can deploy a Supervision Participant project whose status is **Out Of Date**.

Before deploying a built Supervision Participant project, verify its **Build State**, page 790. Only Supervision Participant projects whose status is **Built** contain the latest changes that you have made, page 790 to the system.

Aborting Deployment Tasks

After confirming the command by clicking **OK** in the **Deploy Built Project** or **Deployment File Selection** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task.

Files that have already been transferred to workstations at the moment you abort the task are not removed from the target locations. Open the shared folders on each workstation to confirm which files have been transferred.

Only the user who selected the command is allowed to cancel it.

Deploying the Supervision Files

To deploy selected files pertaining to the built Supervision Participant project to the target computers from the executable, proceed as follows.

Step	Action
1	Ensure that the target computers are prepared, page 842.
2	Right-click the executable that you want to deploy and select Deploy Built Project . Result: The Deploy Built Project dialog box opens, which gives you the possibility to modify the selection of deployed files.
3	Click: <ul style="list-style-type: none"> • Yes to open the Deployment File Selection dialog box, page 848; proceed to step 4. • No to deploy only the master Supervision project file (.ctz) but no OFSOPC/OPC UA configuration file and no included projects. • Cancel to cancel the deployment. Result: If you clicked No : <ul style="list-style-type: none"> • The files are deployed to the appropriate folders on each target computer. • Deployment information is displayed in the notification panel for each target computer, including the IP addresses that were used. • The Deploy Built Project dialog box opens, which contains a summary of the deployment process per target computer.
4	In the Deployment File Selection dialog box, select the files that you want to deploy and click OK . Result: <ul style="list-style-type: none"> • The files are deployed to the appropriate folders on each target computer. • Deployment information is displayed in the notification panel for each target computer, including the IP addresses that were used. • The Deploy Built Project dialog box opens, which contains a summary of the deployment process per target computer. NOTE: Click Cancel to cancel the deployment.

Restoring the Supervision Project

After a first deployment, you must restore the deployed Supervision Participant project on each target computer.

Restoring the project creates a Supervision project folder on the target computer that has the same name as the deployed Supervision master project file (.ctz).

For information on restoring the deployed Supervision project, refer to topic describing how to restore Supervision projects (see *EcoStruxure™ Process Expert, Supervision Participant Services, User Guide*).

Deploying the Built Supervision Participant Project Again

To perform a subsequent deployment of the complete Supervision project from the same executable after you have modified and built it, follow the procedure describing how to deploy the project files, page 844.

The existing files in the *SoCoDeploy*, *SoCoOfsDeploy*, and/or *SoCoOpcUaDeploy* shared folders (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*) are replaced on the target computers with the version that is contained in the Supervision project that you are deploying again.

Then, to apply the changes, the steps that you must complete on each of the target computer vary, depending on the changes that you have deployed (for example, updating the OFSOPC/OPC UA configuration data manually). Refer to the topic describing how to execute the deployed Supervision project (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).

NOTE: If you had already deployed attached user files, page 511 and you redeploy the same Supervision project after removing one or more of user files, these files remain in the *SoCoDeploy* shared folder where they had been deployed initially. You must remove them manually from the shared folder.

Deploying the Built Supervision Participant Project from the Workstation

Overview

The **Deploy Built Project** command that is available from the workstation provides an alternative deployment method. By selecting it, you can transfer selected files of the built Supervision Participant project to a specific computer connected to the logical network (for example, the operation server or an operator station). The deployment is performed from the workstation that models this computer. The workstation needs to be mapped ([service mapping](#), [page 767](#)) to the executable of the Supervision project that you are deploying.

After a first deployment, you must restore the deployed Supervision Participant project on the target computer.

Information related to subsequent deployments of the same Supervision project is provided in the topic describing how to deploy the built Supervision Participant project again, [page 844](#).

Deployed Files

You can deploy the following files:

- The Supervision master project (.ctz file, for example, *Supervision_1_Executable_1.ctz*), which includes the Supervision cocomputer setup file (*citect.ini*) (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).
- Attached user files, [page 511](#). These are located inside the master project (.ctz) on the target computer.
- The OFSOPC/OPC UA configuration files (.xml), [page 788](#) that are related to the workstation: Deployed only to workstations mapped to the *IOServer* service of the Supervision project.
- The included projects, [page 510](#).

You can select not to deploy the configuration files and/or the included project in the **Deployment File Selection** dialog box, [page 848](#), which is accessible during the deployment process.

NOTE: Supervision project files that are deployed are already compiled by using the Supervision Participant. To verify the version, open a Supervision Participant window (for example, by refining a Supervision project), and open the **About** dialog box.

Selecting Participant Project and Control Executable

When you select the **Deploy Built Project** command for a workstation, the **Deploy Built Project** dialog box opens, which allows you to select:

- **Project:** Any Supervision project that exists in the system whose executable is mapped to at least one [service](#), [page 767](#)
- **Executable:** For each project, any built or out of date Supervision executable that is mapped to the workstation (service mapping).

Built Participant Project Build Status

A message informs you if the built Supervision Participant project that you want to deploy is out of date. You can still proceed with the deployment.

Before deploying a built Supervision Participant project, ensure that its **Build State**, [page 790](#) is up to date and/or that it contains the latest changes that you have made, [page 790](#) to the system.

Supervision Participant Project Information

Before proceeding with the deployment of a built Participant project, the **Confirm Deploy Built Project** dialog box opens and informs you about:

- The workstation that you are deploying to.
- The built Participant project to be deployed.
- The Participant project that was deployed last to the engine.

The following figure shows an example of the **Confirm Deploy Built Project** dialog box.

The tables describe the information that is displayed in the **Confirm Deploy Built Project** dialog box.

Engine Information section	Description
Status	Status of the NIC of the workstation that is selected for deployment. If the status is Not Reachable , you cannot proceed with the deployment. In this case, verify the IP address and the connection to the workstation. NOTE: To refresh the engine status and attempt to connect to it, click the reconnect button, page 803.
IP Address	Name of the NIC through which the built Participant project is deployed and its IP address that is configured in the Topology Explorer . If the workstation has several NICs, select the one to use for deployment.

Project Information section	Selected Project column	Last Deployed Project column
Executable	Identifier of the Supervision executable that you have selected and that contains the built Participant project to be deployed.	Identifier of the Supervision executable that contains the built Participant project that you last deployed.
Workstation	Name of the topological entity to which you are deploying the built Participant project.	Name of the topological entity to which you last deployed the built Participant project.
System	Identifier of the system that the workstation and the Supervision project, which you have selected belong to.	Identifier of the system that the engine and the Supervision project, from which you have last deployed the Participant project belong to.
Project	Identifier of the Supervision Participant project that you have selected.	Identifier of the Supervision Participant project associated to the executable that you last deployed.
Last Built On	Date and time when you successfully built the Supervision Participant project to be deployed.	Date and time when you successfully built the Supervision Participant project that you last deployed.

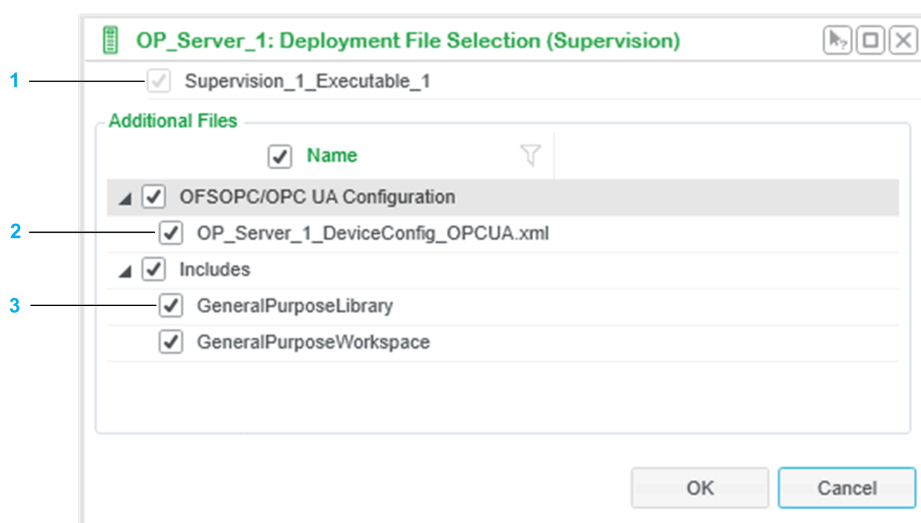
Project Information section	Selected Project column	Last Deployed Project column
Last Deployed On	Not applicable	Date and time when you last deployed the Participant project from the workstation or from the executable associated to the Supervision project.
Select OFSOPC/OPC UA configuration files and/or included projects	Check box. When selected, opens the Deployment File Selection dialog box, where you can select the OFSOPC/OPC UA configuration files and included projects to be deployed in addition to the Supervision master project file (.ctz). When cleared, the dialog box does not open and only the Supervision master project file (.ctz) is deployed.	

Check box	Description
Select OFSOPC/OPC UA configuration files and/or included projects	When selected, opens the Deployment File Selection dialog box, where you can select the OFSOPC/OPC UA configuration files and included projects to be deployed in addition to the Supervision master project file (.ctz). When cleared, the dialog box does not open and only the Supervision master project file (.ctz) is deployed.

NOTE: When you deploy for the first time to a workstation, the fields of the **Last Deployed Project** section are empty.

Selecting the Files to Deploy

The following figure shows an example of the **Deployment File Selection** dialog box, which opens when you select the **Select OFSOPC/OPC UA configuration files and/or included projects** check box and click **OK** in the **Confirm Deploy Built Project** dialog box.



Item	Description
1	Supervision master project file (.ctz), which is selected by default. You cannot clear the check box.
2	OFSOPC or OPC UA configuration files, page 788 that are related to the workstation and generated depending on the selected protocol for the tag container. Configuration files can be selected only for workstations that are mapped to an I/O server.
3	Included projects, page 510 that are associated to the master project.

Aborting Deployment Tasks

After confirming the command by clicking **OK** in the **Reconfirm Deploy Built Project** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task.

Files that have already been transferred to the workstation at the moment you abort the task are not removed from the target locations. Open the shared folders on the workstation to confirm which files have been transferred.

Only the user who selected the command is allowed to cancel it.

Deploying the Supervision Files

To deploy selected files pertaining to the built Supervision Participant project from a workstation, proceed as follows.

Step	Action
1	In the Topology Explorer , identify the workstation that represents the computer you want to deploy to.
2	Ensure that the target computer is prepared, page 842.
3	Right-click the workstation and click Supervision > Deploy Build Project . Result: The Deploy Built Project dialog box opens.
4	Click the Project list and select the Supervision project.
5	Click the Executable list and select the executable associated to the built Supervision project that you want to deploy.
6	Click OK . Result: The Confirm Deploy Built Project dialog box opens. NOTE: Click Cancel to close the dialog box without deploying.
7	Verify the status of the engine in the Status field.
8	Verify the NIC identification and IP address or select one in the IP Address field.
9	Verify the information that is displayed in the Project Information section of the Confirm Deploy Built Project dialog box.
10	To modify the selection of files to deploy, select the Select OFS configuration files and/or included projects check box, page 847 and click OK ; otherwise, to exclude configuration files and included projects from the deployed files, clear the check box, click OK , and proceed to step 12. Result: The Deployment File Selection dialog box opens.
11	In the Deployment File Selection dialog box, select the files you want to deploy to the workstation and click OK . Result: The Reconfirm Deploy Built Project dialog box opens. NOTE: Click Cancel to cancel your file selection and revert to the Confirm Deploy Built Project dialog box.
12	Click OK in the Reconfirm Deploy Built Project dialog box. Result: <ul style="list-style-type: none"> The selected files are deployed to the appropriate folders on the target computer. Completed is displayed in the notification panel and provides deployment information. The related date/time field is updated. NOTE: Click Cancel to revert to the Confirm Deploy Built Project dialog box.

Restoring the Supervision Project

After a first deployment, you must restore the deployed Supervision Participant project on the target computer.

Restoring the project creates a Supervision project folder on the target computer that has the same name as the deployed Supervision master project file (.ctz).

For information on restoring the deployed Supervision project, refer to topic describing how to restore Supervision projects (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).

Deploying the Built Supervision Participant Project Again

To perform a subsequent deployment of the complete Supervision project after you have modified and built it, follow the procedure describing how to [deploy the project files](#), [page 849](#).

The existing files in the *SoCoDeploy* and *SoCoOfsDeploy* shared folders (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*) are replaced on the target computer with the version that is contained in the Supervision project that you are deploying again.

Then, to apply the changes, the steps that you must complete on the target computer vary, depending on the changes that you have deployed (for example, updating the OFSOPC/OPC UA configuration data manually). Refer to the topic describing how to execute the deployed Supervision project (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).

NOTE: If you had already deployed [attached user files](#), [page 511](#) and you redeploy the same Supervision project after removing one or more of user files, these files remain in the *SoCoDeploy* shared folder where they had been deployed initially. You must remove them manually from the shared folder.

Deploying Supervision Participant Project Changes

Overview

By using the **Deploy Changes** command, you can deploy the changes that you have made to an already deployed Supervision Participant project given these changes are supported by the command.

You can deploy changes either from the executable or the workstation independently from the method that you have used to perform the previous deployment.

You can select to deploy only the Supervision master project file (.ctz) or also additional files.

Prerequisites

The following conditions must be satisfied to deploy changes by using the **Deploy Changes** command.

- You must have deployed the complete Supervision Participant project to the workstation and restored it by using the Supervision software.
- The Supervision project folder (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*) that was created by restoring the deployed Supervision Participant project must be present on the target computer.
- The changes that can be deployed incrementally are only those that can be applied by using the **Build** command:
 - A maximum of 5 changes involving Supervision pages.
 - A maximum of 30 changes involving data facets.
 - Modifications of parameters of the **Message** category of Supervision elements

Refer to the topic describing the build stage for details, [page 792](#).

NOTE: Using the **Build All** command disables the **Deploy Changes** command.

Deployed Files

The files that you can deploy are the same as for the deployment of the complete Supervision Participant project, [page 846](#).

You can change the selection during the deployment process in the **Deployment File Selection** dialog box, [page 848](#).

Aborting Deployment Tasks

After confirming the command by clicking **OK** in the **Reconfirm Deploy Changes** dialog box, the Abort icon is displayed in the [notification panel, page 85](#). Click the icon to cancel the task.

Files that have already been transferred to workstations at the moment you abort the task are not removed from the target locations. Open the shared folders on each workstation to confirm which files have been transferred.

Only the user who selected the command is allowed to cancel it.

Deploying Changes from the Executable

To deploy changes to the Supervision Participant project to the target computers from the executable, proceed as follows.

Step	Action
1	Ensure that the target computers are prepared, page 842.
2	In the Project Explorer , expand the Supervision project that you want to deploy from and open the Executables folder to show existing executables.
3	Right-click the executable that you want to deploy and select Deploy Changes . Result: The Deploy Changes dialog box opens, which gives you the possibility to modify the selection of files.
4	Click: <ul style="list-style-type: none"> • Yes to open the Deployment File Selection dialog box, page 848; proceed to step 5. • No to deploy only the master Supervision project file (.ctz) but no OFSOPC/OPC UA configuration file and no included project. • Cancel to cancel the deployment. Result: If you clicked No : <ul style="list-style-type: none"> • The master Supervision project file is deployed to the appropriate folders on each target computer. • Deployment information is displayed in the notification panel for each target computer, including the IP addresses that were used. • The Deploy Changes dialog box opens, which contains a summary of the deployment process per target computer.
5	In the Deployment File Selection dialog box, select the files that you want to deploy and click OK . Result: <ul style="list-style-type: none"> • The files are deployed to the appropriate folders on each target computer. • Deployment information is displayed in the notification panel for each target computer, including the IP addresses that were used. • The Deploy Changes dialog box opens, which contains a summary of the deployment process per target computer. NOTE: Click Cancel to the deployment.

Deploying Changes from the Workstation

To deploy changes to the Supervision Participant project from a workstation, proceed as follows.

Step	Action
1	In the Topology Explorer , identify the workstation that represents the computer you want to deploy to.
2	Ensure that the target computer is prepared, page 842.
3	Right-click the workstation and click Supervision > Deploy Changes . Result: The Confirm Deploy Changes dialog box opens. The information of the last deployed executable and project is indicated in the Executable and Project fields of the Project Information section.
4	Verify the status of the engine in the Status field.
5	Verify the NIC identification and IP address or select one in the IP Address field.
6	If you want to deploy other files in addition to the master Supervision project, select the Select OFSOPC/OPC UA configuration files and/or included projects check box, page 847 and click OK ; otherwise, to exclude configuration files and included projects from the deployed files, clear the check box, click OK , and proceed to step 8 Result: The Deployment File Selection dialog box opens. NOTE: Click Cancel to close the dialog box without deploying.

Step	Action
7	<p>In the Deployment File Selection dialog box, select the files you want to deploy to the workstation and click OK.</p> <p>Result: The Reconfirm Deploy Changes dialog box opens.</p> <p>NOTE: Click Cancel to cancel your file selection and revert to the Confirm Deploy Changes dialog box.</p>
8	<p>Click OK in the Reconfirm Deploy Changes dialog box.</p> <p>Result:</p> <ul style="list-style-type: none">• The selected files are deployed to the appropriate folders on the target computer.• Completed is displayed in the notification panel and provides deployment information.• The related date/time field is updated. <p>NOTE: Click Cancel to revert to the Confirm Deploy Changes dialog box.</p>

Execution Stage

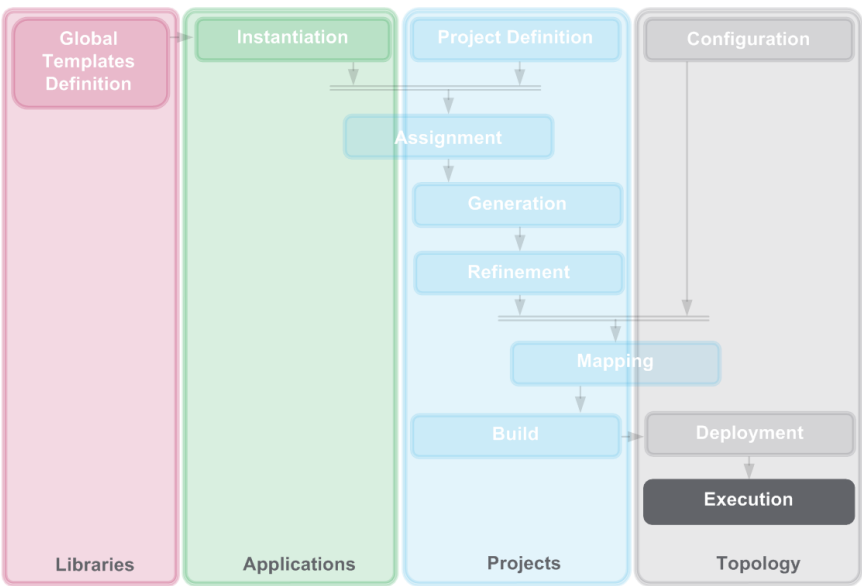
What's in This Chapter

Control Project Execution Stage	855
Supervision Project Execution Stage	907

Overview

This chapter describes how to execute and/or modify the Control and Supervision projects that you have deployed to engines of the system.

The following figure shows the position of the **Execution** stage within the system engineering life cycle.



Refer to the Execution stage, page 68 for a description of the purpose of this stage.

Control Project Execution Stage

Overview

This section describes how to use the **Topology Explorer** to change the state of controllers, modify deployed Control Participant projects, and how to transfer changes to the associated logical Participant project in EcoStruxure Process Expert.

NOTE: When one or more users work on components of the same system (for example, Participant projects, applications, project containers, topological entities), locking mechanisms may restrict certain concurrent actions. In this case, refer to the notification panel for details.

Control Participant Project Execution Process

Overview

You can execute a deployed Control Participant project, make changes to the deployed Participant project online, and update the logical Control Participant source project with these changes.

Following the deployment of a built Control Participant project, the following commands are enabled:

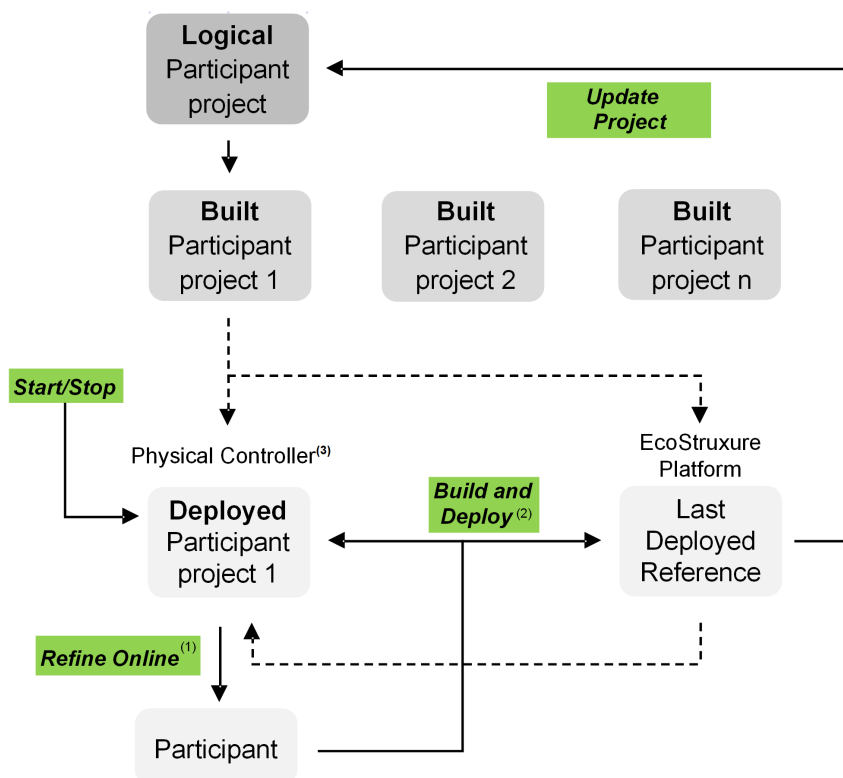
- **Start**
- **Stop**
- **Refine Online**
- **Sync (Primary→Standby)**
- **Update Project**

The commands are available for the following engines:

- Controllers.
- Workstations running a controller simulator.

Execution Flowchart

The following figure illustrates the different steps of the execution of the deployed Control Participant project and their associated commands.



- - - Represents commands pertaining to the deployment stage, page 795.

(1) Includes the **section import** functionality, page 885.

(2) The functionality is a sub-command of the **Refine Online** command in the Control Participant.

(3) Or controller simulator (Control service of a workstation).

Prerequisites

The following table describes the actions that are prerequisite to using a command.

Command	Prerequisites
Start / Stop	You deployed the built Control Participant project.
Refine Online	You deployed the built Control Participant project.
Sync (Primary→Standby)	<ul style="list-style-type: none"> You deployed the built Control Participant project but either of the following occurred: <ul style="list-style-type: none"> Deployment to or synchronization of the standby controller did not complete successfully after using the Deploy, Re-Deploy, Deploy Changes, or Refine Online command. Deployment was done to the primary controller because the <i>ControllerExist</i> parameter is set to <i>Primary</i>. You must set the parameter to <i>PrimaryAndStandby</i>, ensure that the standby controller is reachable, and that both controllers are connected by a high-speed cable link. Ensure that the status of the primary controller is Running Primary.
Update Project	<ul style="list-style-type: none"> You deployed the built Control Participant project. You modified the deployed Control Participant project by using the Refine Online command. <p>NOTE: Certain changes cannot be transferred, page 834 to the logical Control Participant project by using this command.</p>

For the other actions that are required, for example, mapping the Control executable to a controller, refer to the system engineering life cycle.

Control Project Signature Verification

Before executing the **Start/Stop**, **Sync (Primary→Standby)**, or **Refine Online** command, EcoStruxure Process Expert compares the Control Participant project that is deployed in the controller with the *last deployed reference*, page 856, which is the Control Participant project residing in the EcoStruxure Process Expert content repository.

If they are identical, the command is executed. Otherwise, a recovery process starts, which may take several minutes to complete.

If a notification about a signature mismatch is displayed, the command cannot be executed. Contact your local Schneider Electric service representative.

Client/Server Connection

After an unexpected interruption of the client/server connection, the notification panel of engineering clients that were connected to the system server may contain information about operations that did not complete successfully because of this interruption. It may also indicate corrective actions.

After an interruption of the client/server connection, the states of the engineering client and the system server may be inconsistent if a deployment operation was in progress.

NOTICE

DATA CORRUPTION

After an interruption of the client/server connection, ensure that the last operation executed by EcoStruxure Process Expert was completed successfully and, if necessary, repeat the last operation.

Failure to follow these instructions can result in equipment damage.

NOTE: For more information on client/server connection interruption, refer to the topic describing the [client/server connection](#), page 76.

Start and Stop Commands

EcoStruxure Process Expert can send a start or stop command to the controller.

⚠ WARNING

UNKNOWN OPERATIONAL STATE OF EQUIPMENT

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Execution With Redundant Controllers

Overview

This topic describes the aspects that are specific to the execution of Control Participant projects with redundant controllers.

The execution commands that are used on standalone configurations are also used for redundant controllers.

A description of these commands is given further in the chapter.

For a description of the sequence of actions that are performed on the controllers, refer to the topic describing [redundant controller scenarios](#), page 889.

NOTE: For M580 controllers and BMENOC communication modules, you can only use the main IP address, page 679.

ControllerExist Property

Ensure that the value of the **ControllerExist** property, page 689 of the redundant controller matches the physical configuration of the hardware.

If a mismatch is detected, you cannot use execution commands.

Controller Status for Redundant Controllers

Refer to the information that is provided in the topic describing the deployment to [redundant controllers](#), page 812.

Starting and Stopping a Redundant Controller

For redundant controllers, the **Start/Stop** commands act on the primary and the standby controllers provided that both are reachable.

When you select the **Start** or **Stop** command for a [redundant controller](#), page 892 composed of a primary and a standby controller to which IP addresses *IP*, and *IP* + 1 are assigned respectively, the **Confirm Start** or **Confirm Stop** dialog box opens respectively and provides information on both controllers.

NOTE: If you select the **Start** command and the status of the standby controller is **Stop Offline** (or **Stop** for controllers of the **M580** platform), the primary controller starts and synchronizes the standby controller so that both controllers start.

Redundant Controller Stop Sequence

When you stop a redundant controller and both controllers are running, they are stopped in the following order:

- The standby controller
- The primary controller

Refining Online With a Redundant Controller

To deploy changes made online by using the **Refine Online** command, the primary controller must be in **Running Primary** or **Stop Offline** (or **Stop** for controllers of the **M580** platform) state.

EcoStruxure Process Expert executes the command on the project that is deployed on the primary controller.

If you deploy online changes that do not require stopping the controller and the primary controller is in **Running Primary** state, the following actions are performed in sequence:

- The configuration mismatch feature of the controller is enabled so that the standby controller does not enter the offline operating mode when changes are deployed to the primary controller.
- Deployment is performed to the primary controller first and upon successful completion of the deployment, the standby controller is automatically synchronized.
- The configuration mismatch feature is disabled.

When you deploy changes made online to a redundant controller and deployment to the primary controller succeeds, the *synchronization of the standby controller*, page 892 may not complete. In this case, a message is displayed in the notification panel to inform you. If an interruption of the connection to the system server is the cause, the notification panel also contains additional information about steps that you must complete when the connection to the system server is re-established.

For example, if EcoStruxure Process Expert cannot disable the configuration mismatch feature and a switchover occurs after deployment to the primary controller, the standby controller operates with a program that is not synchronized with the changes that you deployed to the primary controller.

WARNING

UNINTENDED EQUIPMENT OPERATION

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: If the primary controller is not in **Running Primary** state, synchronization of the standby controller cannot complete successfully. In this case, *starting*, page 861 the primary controller synchronizes the standby one.

Redundant Controller Synchronization Timeout

After you select the **Build and Deploy** command (see *EcoStruxure Process Expert, Control Participant Services, User Guide*), upon successful deployment to the primary controller, the deployed Control Participant project is stored in the content repository as the last deployed reference, page 856.

It helps avoid a project mismatch, page 857 and allows you to synchronize the standby controller by using the **Sync (Primary→Standby)** command if the synchronization timeout is elapsed, page 866.

Starting and Stopping Controllers

Overview

Following the deployment of a built Control Participant project to a controller, you can change the state of the controller by using the following commands:

- **Start**
- **Stop**

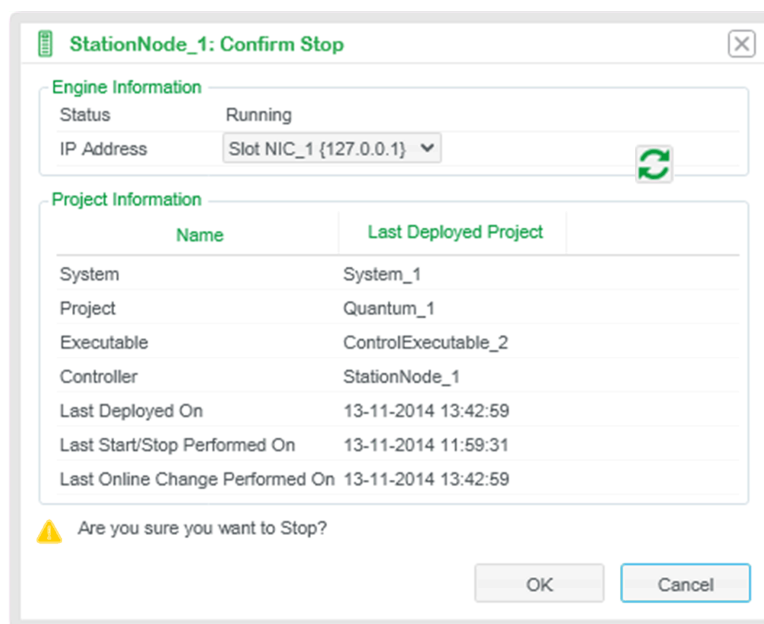
The commands act on the physical controller or on the controller simulator. The IP address of the controller or of the controller simulator must match with the address that is assigned to the corresponding entity.

Confirming Execution

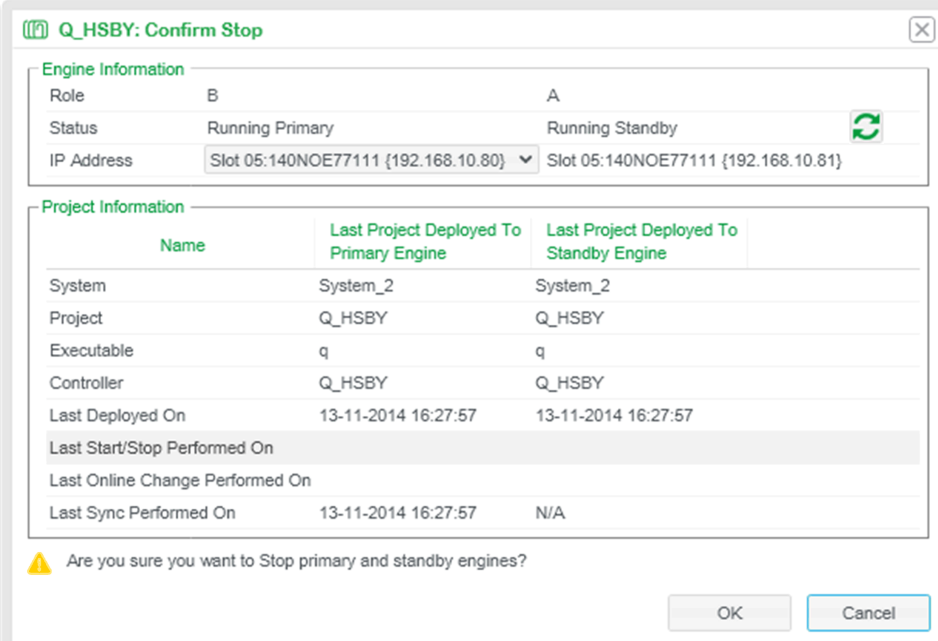
When you select the **Start** or **Stop** command, the **Confirm Start** or **Confirm Stop** dialog box opens respectively. It contains information about:

- The controller you want to start or stop.
- The Control Participant project that is deployed to the controller.

The following figure shows an example of the **Confirm Stop** dialog box that opens when you select the **Stop** command for a controller simulator.



The following figure shows an example of the **Confirm Stop** dialog box that opens when you select the **Stop** command for a redundant controller.



Q_HSBY: Confirm Stop

Engine Information

Role	B	A
Status	Running Primary	Running Standby
IP Address	Slot 05:140NOE77111 {192.168.10.80}	Slot 05:140NOE77111 {192.168.10.81}

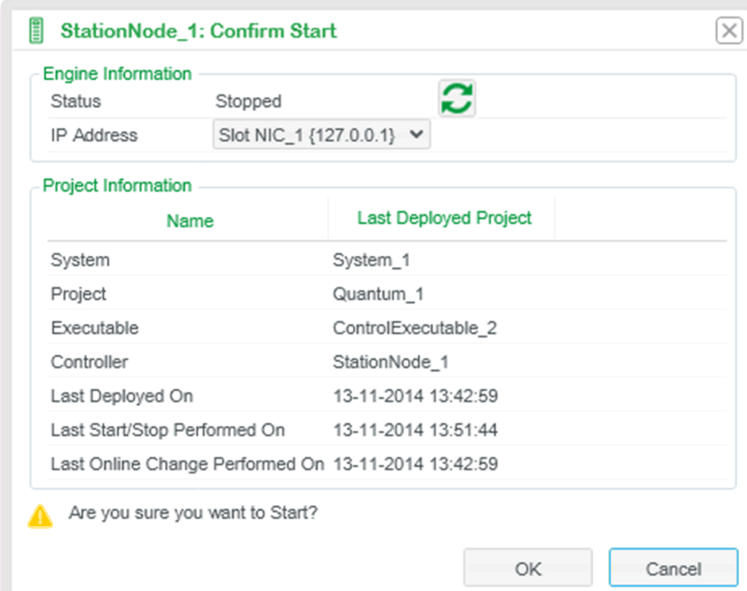
Project Information

Name	Last Project Deployed To Primary Engine	Last Project Deployed To Standby Engine
System	System_2	System_2
Project	Q_HSBY	Q_HSBY
Executable	q	q
Controller	Q_HSBY	Q_HSBY
Last Deployed On	13-11-2014 16:27:57	13-11-2014 16:27:57
Last Start/Stop Performed On		
Last Online Change Performed On		
Last Sync Performed On	13-11-2014 16:27:57	N/A

⚠ Are you sure you want to Stop primary and standby engines?

OK Cancel

The following figure shows an example of the **Confirm Start** dialog box that opens when you select the **Start** command for a controller simulator.



StationNode_1: Confirm Start

Engine Information

Status	Stopped
IP Address	Slot NIC_1 {127.0.0.1}

Project Information

Name	Last Deployed Project
System	System_1
Project	Quantum_1
Executable	ControlExecutable_2
Controller	StationNode_1
Last Deployed On	13-11-2014 13:42:59
Last Start/Stop Performed On	13-11-2014 13:51:44
Last Online Change Performed On	13-11-2014 13:42:59

⚠ Are you sure you want to Start?

OK Cancel

The following table describes the information that is displayed in the **Engine Information** section of the **Confirm Start** and **Confirm Stop** dialog boxes.

Label	Description
Role⁽¹⁾	Role of the controller: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the controller, the role is indicated as Not Detected .
Status	Status of the controller that is selected. If the status is Not Reachable you cannot proceed. In this case, verify the IP address, the connection to the controller, and the status of the controller.
IP Address	Name of the communication module through which the command is sent and the IP address that is configured in the Topology Explorer . If the topological entity has several communication modules, you can select one of them. NOTE: For M580 controllers and BMENOC**** communication modules, you can only use the main IP address, page 679. NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.
(1) Displayed only for a redundant controller.	

NOTE: To refresh the controller status and attempt to connect to it, click the **reconnect** button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Start** and **Confirm Stop** dialog boxes.

Name column	Last Deployed Project column ⁽²⁾
System	Identifier of the system that the controller from which you last deployed the Participant project belongs to.
Project	Identifier of the Control Participant project associated to the deployed Participant project that you last deployed.
Executable	Identifier of the Control executable that contains the built Participant project that you last deployed.
Controller	Name of the topological entity to which you last deployed the built Participant project.
Last Deployed On	Date and time when you last deployed the Participant project to the controller or you last completed the Deploy Changes / Undo Online Changes command, independently if changes were deployed or not.
Last Start/Stop Performed On	Date and time when you last started or stopped the controller.
Last Online Change Performed On	Date and time when you last changed the deployed Participant project by using the Refine Online command; otherwise, the field is empty.
Last Sync. Performed On⁽¹⁾	Date and time when you last synchronized the standby controller with the primary controller by using the Synchronize (Primary→Standby) command; otherwise, the field is empty.
(1) Displayed only for a redundant controller.	
(2) When you start or stop a redundant controller, the dialog box shows separate columns for the primary and the standby controller.	

Starting a Controller

To start the controller, proceed as follows.

Step	Action	Result
1	Ensure that the computer is connected, page 803 to the controller.	–
2	In the Topology Explorer , right-click the controller and select Start .	The Confirm Start dialog box opens.
3	Verify the information that is displayed.	–
4	Click OK .	The Reconfirm Start dialog box opens.
5	Click OK . NOTE: Click Cancel to revert to the Confirm Start dialog box without starting the controller.	<ul style="list-style-type: none"> The controller is started. Completed is displayed in the notification panel once the command is successfully executed. For a redundant controller, the information pane displays separate messages for the primary and the standby controllers by using their respective IP address to identify them. The Last Start/Stop Performed On field is updated.

WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that you are connected to the controller you intend to start.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For more information, refer to *Identifying Controllers*, page 796.

WARNING

UNKNOWN OPERATIONAL STATE OF EQUIPMENT

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

When starting a redundant controller, the synchronization of the standby controller may not complete successfully. In this case, a message is displayed in the notification panel to inform you. For more information, refer to *Redundant Controller Reservation and Connection Scenarios*, page 892.

WARNING

UNINTENDED EQUIPMENT OPERATION

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Stopping a Controller

To stop the controller, proceed as follows.

Step	Action	Result
1	Ensure that the computer is connected, page 803 to the controller.	–
2	In the Topology Explorer , right-click the controller that you want to stop and select Stop .	The Confirm Stop dialog box opens.
3	Verify the information that is displayed.	–
4	Click OK .	The Reconfirm Stop dialog box opens.
5	Click OK . NOTE: Click Cancel to revert to the Confirm Stop dialog box without stopping the controller.	<ul style="list-style-type: none"> The controller is stopped. Completed is displayed in the notification panel once the command is successfully executed. <p>For a redundant controller, the information pane displays separate messages for the primary and the standby controllers by using their respective IP address to identify them.</p> <ul style="list-style-type: none"> The Last Start/Stop Performed On field is updated.

WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that you are connected to the controller you intend to stop.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For more information, refer to [Identifying Controllers](#), page 796.

WARNING

UNKNOWN OPERATIONAL STATE OF EQUIPMENT

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Synchronizing Redundant Controllers

Overview

The **Sync (Primary→Standby)** command allows you to manually transfer the deployed Control Participant project from the primary to the secondary redundant controller if there is a program, configuration, or data mismatch.

Manual synchronization is only possible when the primary controller is running. The state of the standby controller is irrelevant.

EcoStruxure Process Expert verifies if the program or configuration of the primary and the secondary redundant controllers are identical and proceeds with the transfer only if it detects a difference.

It also verifies that there is no project mismatch, page 857.

The command is available only for entities of the topology that are configured as redundant controllers.

Also refer to Redundant Controller Deployment and Execution Scenarios, page 892.

High-Speed Cable Link

If the high-speed cable connection between the primary and standby controllers is interrupted, synchronization does not complete successfully and a message is displayed to inform you.

Synchronization Timeout

During synchronization of the standby controller, EcoStruxure Process Expert compares the signatures of the primary and the standby controllers. If they are identical, it considers that the synchronization completed successfully.

After a 3-minute timeout, if the signatures are not detected as identical, a dialog box opens to inform you that synchronization did not complete successfully and cancels the comparison process. However, the synchronization is not canceled and may complete after the timeout elapsed.

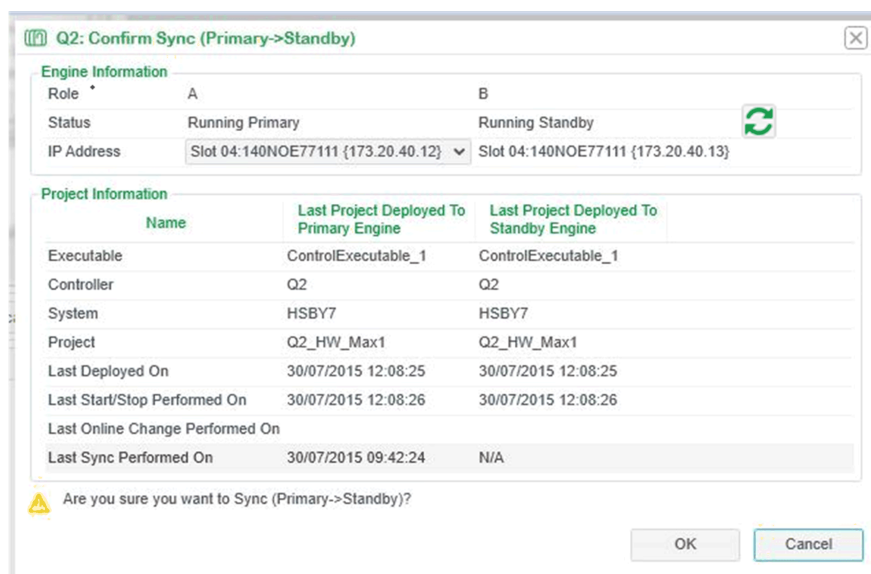
You can try synchronizing again by using the **Sync (Primary→Standby)** command.

Confirming the Synchronization

Before proceeding with the synchronization, the **Confirm Sync (Primary→Standby)** dialog box opens, which contains information about:

- The state of the primary and standby controllers.
- The deployed Control Participant project to transfer to the standby controller.

The following figure shows an example of the **Confirm Sync (Primary→Standby)** dialog box that opens when you select the **Sync (Primary→Standby)** command with the primary controller in **Running Primary** state.



The following table describes the information that is displayed in the **Engine Information** section of the **Confirm Sync (Primary→Standby)** dialog box.

Label	Description
Role	Role of the controller: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the controller, the role is indicated as Not Detected .
Status	Status of the controller that is selected. If the status is Not Reachable you cannot synchronize. In this case, verify the IP address, the connection to the controller, and the status of the controller.
IP Address	Name of the communication module through which the command is sent and the IP address that is configured in the Topology Explorer . If the topological entity has several communication modules, you can one of them. NOTE: For M580 controllers and BMENOC•••• communication modules, you can only use the main IP address, page 679.

NOTE: To refresh the controller status and attempt to connect to it, click the **reconnect** button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Sync (Primary→Standby)** dialog box.

Name column	Last Deployed Project To Primary Engine / Standby Engines columns
System	Identifier of the system that the controller from which you last deployed the Participant project belongs to.
Project	Identifier of the Control Participant project associated to the deployed Participant project that you last deployed.
Executable	Identifier of the Control executable that contains the built Participant project that you last deployed.
Controller	Name of the topological entity to which you last deployed the built Participant project.
Last Deployed On	Date and time when you last deployed the Participant project to the controller or you last completed the Deploy Changes / Undo Online Changes command, independently if changes were deployed or not.
Last Start/Stop Performed On	Date and time when you last started or stopped the controller.
Last Online Change Performed On	Date and time when you last changed the deployed Participant project by using the Refine Online command; otherwise, the field is empty.
Last Sync. Performed On	Date and time when you last synchronized the standby with the primary controller by using the Synchronize (Primary→Standby) command; otherwise, the field is empty.

NOTE: If EcoStruxure Process Expert detects that the controller is stopped, the **Confirm Sync (Primary→Standby)** dialog box opens to inform you that the synchronization cannot be completed.

Synchronizing the Standby Controller

To synchronize the standby controller, proceed as follows.

Step	Action
1	Ensure that the primary controller is connected to the computer, page 803 and its status is Running Primary .
2	In the Topology Explorer , right-click the redundant controller that you want to synchronize and select Sync (Primary→Standby) . Result: The Confirm Sync (Primary→Standby) dialog box opens.
3	Verify the information that is displayed.
4	Click OK .
5	The Reconfirm Sync (Primary→Standby) dialog box opens.
6	Click OK . NOTE: Click Cancel to revert to the Confirm Sync (Primary→Standby) dialog box without synchronizing.
7	Result: If a difference is detected in the program or the configuration between the primary and the standby controllers, it proceeds as follows: <ul style="list-style-type: none"> Transfers the deployed Control Participant project from the primary to the standby controller. Displays Completed in the notification panel once it has successfully executed the command. Updates the related date/time field.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

Ensure that you are connected to the controller you intend to start.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For more information, refer to *Identifying Controllers*, page 796.

⚠ WARNING**UNKNOWN OPERATIONAL STATE OF EQUIPMENT**

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Synchronization of the standby controller may not complete, for example, if the primary or the standby controller is reserved. In this case, a message is displayed in the notification panel to inform you. For more information, refer to *Redundant Controller Reservation and Connection Scenarios*, page 892.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

About Changes Made Online

Overview

This topic provides an overview of the changes that you can make online during the system engineering life cycle.

After refining the deployed project online, the logical and the deployed Control Participant projects are different.

If you refine (offline) the logical Control Participant project and deploy these changes, the operation may undo your online refinements. You can preserve online refinements by transferring them to the logical Control Participant project. However, some online refinements cannot be transferred or can only be transferred partially.

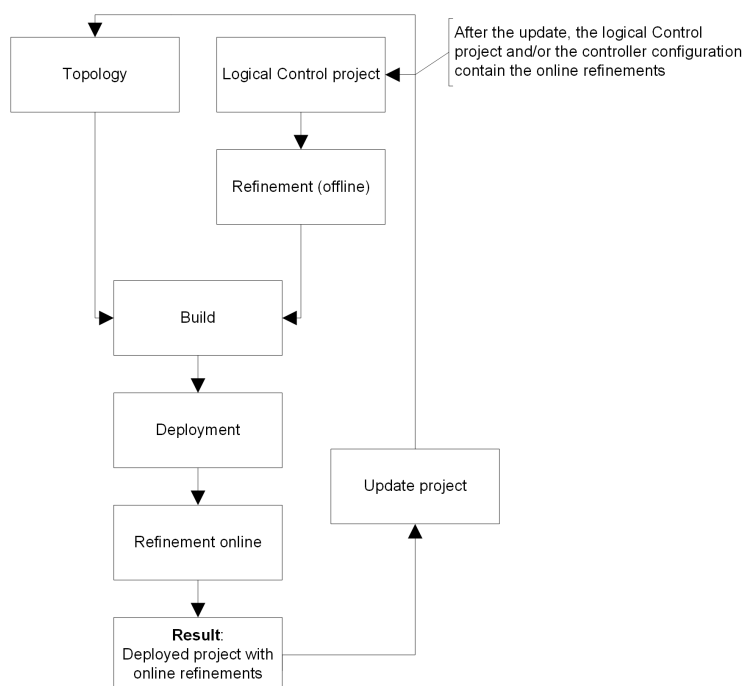
The following table indicates which workflow to refer to depending on your objective.

Objective	Refer to
Preserving online changes in the logical Control Participant project and the controller configuration by updating the logical project.	Workflow 1
Preserving online refinements that cannot be transferred to the logical Control Participant project	Workflow 2
Updating the logical Control Participant project with references to Device DDT variables coming from the topology.	Workflow 3

NOTE: For details on using the various commands that allow you to refine online, page 874, deploy changes, page 830, and update the logical Control project, page 897, refer to the corresponding topics in the deployment and execution sections.

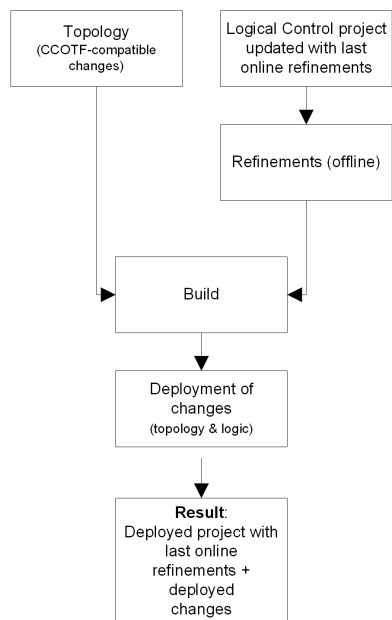
Workflow 1

The figure illustrates the typical workflow to preserve online refinements by transferring them to the logical Control Participant project.



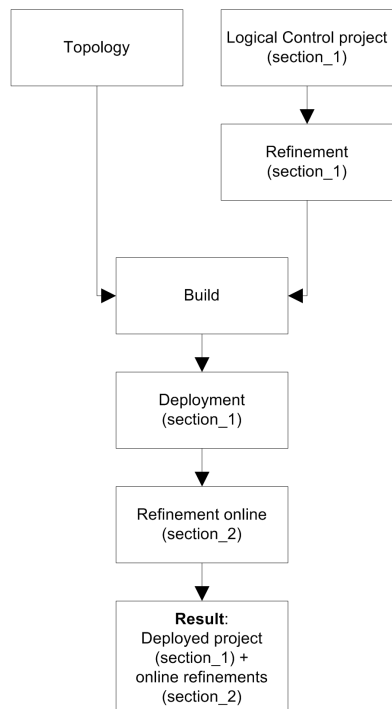
NOTE: Not updating the logical Control Participant project allows you to undo online refinements by deploying changes. EcoStruxure Process Expert detects the online refinements as change compared to the logical Control project. However, using the **Deploy Changes** command to **undo online changes**, page 831 is not possible if you made changes to the configuration online.

This allows you to deploy thereafter additional changes to the sections that you modified online while preserving online refinements. These changes can come from the topology, and/or the application, and/or from refinements of the logical Control Participant project.

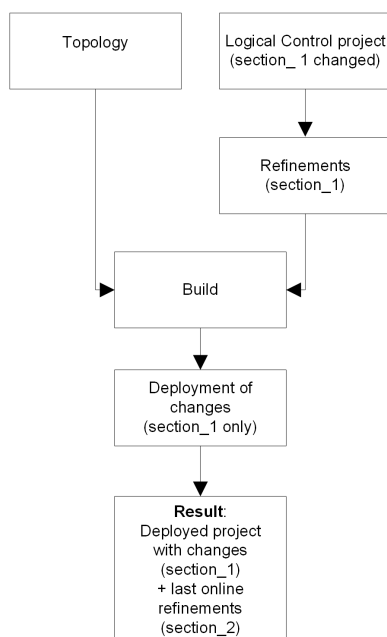


Workflow 2

When you make refinements online that you cannot or do not want to transfer to the Control Participant project, you can make these refinements in a new, separate section (section_2 in this example). You can unselect this section when deploying changes.



Making refinements online in a separate section (section_2 in this example) allows you to preserve them when you deploy changes (to topology, application, or in refinement (offline)) by unselecting the section that contains the online refinements (section_2).



Workflow 3

After you update the logical Control project with the following online refinements, the Control Participant detects an error during project analysis, page 401 because the Device DDT variable does not exist in the logical Control project:

- You add a variable that references a Device DDT variable that exists only in the topology, page 591.
- You add an instance of an existing DFB type that references a Device DDT variable, which exists only in the topology.

Proceed as follows to add a variable that references a Device DDT variable:

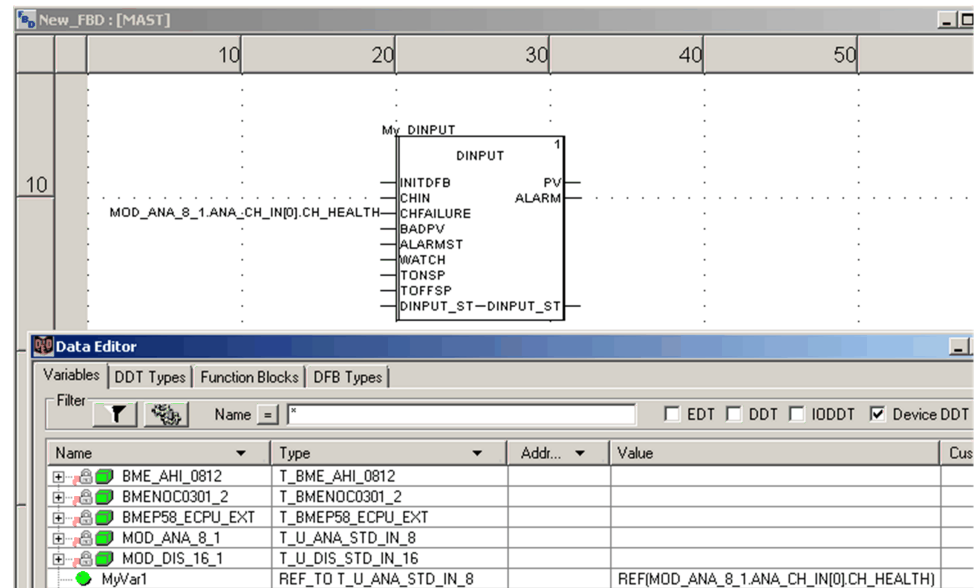
- Do not use the variable in code that is generated by assigned facets or modified during refinement (offline) so that a subsequent deployment of changes does not undo online refinement.
- Use the variable in a separate section, page 871, which you can unselect during deployment of changes.

To add an instance of an existing DFB type, which references a Device DDT variable, build the Control project. This integrates variables coming from the topology into the built Control Participant project.

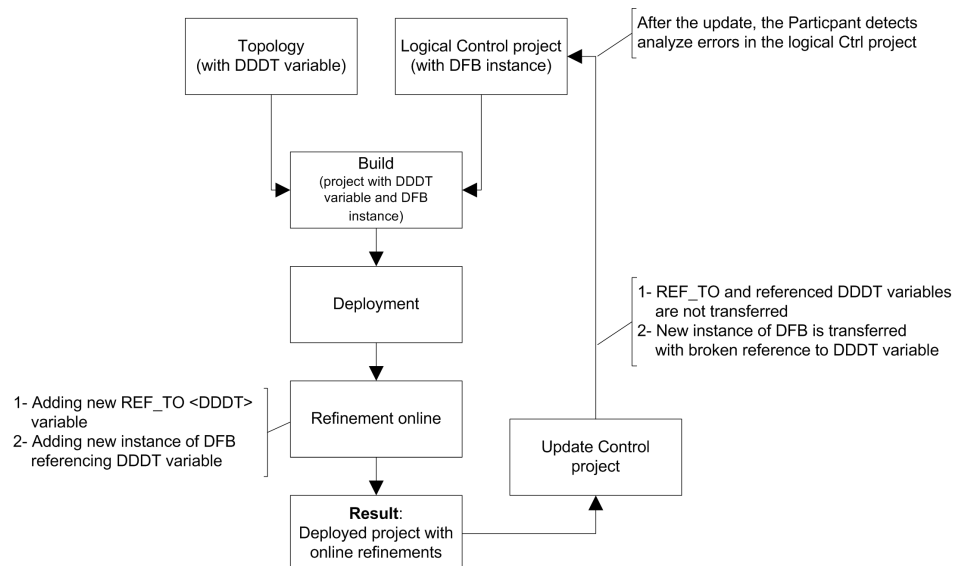
In this example, variable *MyVar1* is created during refinement online. It references the *MOD_ANA_8_1* DDDT variable of a HART analog input module, which comes from the topology. You cannot transfer *MyVar1* nor the *MOD_ANA_8_1* Device DDT variable to the logical Control project.

Variables DDT Types Function Blocks DFB Types						
Filter		Name		<input type="checkbox"/> EDT <input type="checkbox"/> DDT <input type="checkbox"/> IODDT <input checked="" type="checkbox"/> Device DDT		
Name	Type	Addr...	Value	Cus...	Constant	
BME_AHI_0812	T_BME_AHI_0812					
BMENOC0301_2	T_BMENOC0301_2					
BMEP58_ECPU_EXT	T_BMEP58_ECPU_EXT					
MOD_ANA_8_1	T_U_ANA_STD_IN_8					
MOD_DIS_16_1	T_U_DIS_STD_IN_16					
MyVar1	REF_TO T_U_ANA_STD_IN_8		REF(MOD_ANA_8_1.ANA_CH_IN[0].CH_HEALTH)			

In this example, the *My_DINPUT* DFB instance is created during refinement online from the DINPUT DFB type, which exists in the logical Control project. *My_DINPUT* references the *MOD_ANA_8_1* Device DDT variable of a HART analog input module, which comes from the topology. You can transfer the *My_DINPUT* DFB instance to the logical Control project but the reference to the Device DDT variable is broken.



The figure shows how online refinements that reference Device DDT variables coming from the topology affect the deployment and execution workflow.



NOTE: You can create the Device DDT variable in the logical Control project by using the wizard, page 455.

Making Changes Online

Overview

The **Refine Online** command opens the deployed Control Participant project in the Control Participant and allows you to modify the project when the controller is running or stopped.

You may be able to [import sections](#), [page 885](#) of the logical Control Participant project that were modified and generated since their last deployment.

Online refinement requires using the Build and Deploy sub-command, which is described in [Refining the Deployed Control Participant Project Online](#) (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

When you close the Control Participant after deploying changes online, you can update the associated logical Control Participant project with the changes you made online. Until you do so, an [online refinement icon](#), [page 529](#) is displayed on the corresponding topological entity. It indicates that the logical and the deployed Control Participant projects are different.

NOTE: Certain changes that you make online require [stopping the controller](#), [page 840](#) to be applied.

Refining Online With a Workstation

If you select the **Refine Online** command for a workstation (controller simulator) and the previously deployed project is not loaded in the controller simulator anymore, a message may be displayed prompting you to [deploy](#), [page 795](#) the built Control Participant project.

Preserving Changes Made Online

After selecting the **Refine Online** command, if deployment of the following changes does not complete successfully, these changes are not discarded:

- Deployment of changes that do not require stopping the controller.
- Deployment of changes that require stopping the controller.


In this case, select the **Refine Online** command again and continue your work.

NOTE: If the deployment of changes requires disconnecting the controller, the changes are discarded.

Updating the Project With Changes Made Online

After you successfully deployed changes made online and close the Control Participant, you are prompted to update the associated logical Control Participant project and the controller configuration with these changes. You are not prompted if you made changes that require stopping the controller.

If you proceed, the **Update Project** dialog box, page 897 opens. Follow the same process as you would after selecting the **Update Project** command.

If you do not proceed, the online refinement icon, page 529  is displayed on the topological entity. It indicates that changes made online can be applied to the associated logical Control Participant project and/or the controller configuration to make it consistent with the project that is deployed to the controller. The icon is displayed independently of the type of change that you made.

The icon is displayed and you are prompted to perform the update until you complete the update process.

Refer to the topic describing how to update logical Control Participant projects, page 897.

NOTE: If you changed the configuration online, you must perform the update to use the **Deploy changes / Undo Online Changes** command.

NOTE: The online refinement icon is also displayed if you perform the update but cancel the operation, update the project only partially, or if the update does not complete successfully.

Making Changes Online in Separate FBD Sections

If possible, make changes online in separate FBD sections that you create during online refinement because you cannot transfer to the logical Control Participant project all types of changes. Do this even if you do not update the logical Control Participant project after making changes online.

It helps preserve changes that you made online when you **deploy changes**, page 831 after building the logical Control Participant project.

Viewing the Application Structure When Making Changes Online

When you modify a Control Participant project online, the **Application Browser** displays FBD sections based on the location of instances whose Control facets are assigned to these sections.

It shows the relationship between the folder structure of the application and the sections of the Control Participant project.

The pane only shows FBD and non-FBD sections of the Control Participant project that are deployed at the moment you select the **Refine Online** command.

You can open sections from the pane.

For a general description of the contents of the pane, refer to the topic describing the application structure view, page 384.

For information on how to use the pane, refer to the topic describing the refinement of Control Participant projects online (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

Confirming the Command


When you select the **Refine Online** command, the **Confirm Refine Online** dialog box opens and displays information about:

- The controller you selected for online refinement.

- The Control Participant project that is deployed to the controller.

StationNode_1: Confirm Refine Online


Engine Information

Status: Running 

IP Address: Slot NIC_1 {127.0.0.1} ▼

Project Information

Name	Last Deployed Project
System	System_1
Project	Quantum_1
Executable	ControlExecutable_2
Controller	StationNode_1
Last Deployed On	13-11-2014 12:07:47
Last Online Change Performed On	

 Are you sure you want to continue refining online the last deployed project?


OK Cancel

The figure shows an example of the **Confirm Refine Online** dialog box for a redundant controller.

Q_HSBY: Confirm Refine Online


Engine Information

Role	B	A
Status	Running Primary	Running Standby
IP Address	Slot 05:140NOE77111 {192.168.10.80} ▼	Slot 05:140NOE77111 {192.168.10.81} ▼



Project Information

Name	Last Project Deployed To Primary Engine	Last Project Deployed To Standby Engine
System	System_2	System_2
Project	Q_HSBY	Q_HSBY
Executable	q	q
Controller	Q_HSBY	Q_HSBY
Last Deployed On	13-11-2014 16:27:57	13-11-2014 16:27:57
Last Online Change Performed On		
Last Sync Performed On	13-11-2014 16:27:57	N/A

 Are sure you want to continue refining online the last deployed project with primary engine ?

OK Cancel

The following table describes the information that is displayed in the **Engine Information** section of the **Confirm Refine Online** dialog box.

Label	Description
Role⁽¹⁾	Role of the controller: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the controller, the role is indicated as Not Detected .
Status	Status of the controller that is selected. If the status is Not Reachable or Stop Offline⁽²⁾ you cannot refine online. In this case, verify the IP address, the connection to the controller, or the status of the controller.
IP Address	Name of the communication module through which the command is sent and the IP address that is configured in the Topology Explorer . If the controller has several communication modules, you can select one of them. NOTE: For M580 controllers and BMENOC**** communication modules, you can only use the main IP address, page 679. NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.
(1) Displayed only for a redundant controller.	
(2) Stop for controllers of the M580 platform.	

NOTE: To refresh the controller status and attempt to connect to it, click the **reconnect** button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Refine Online** dialog box.

Name column	Last Deployed Project column ⁽²⁾
System	Identifier of the system that the controller from which you last deployed the Participant project belong to.
Project	Identifier of the Control Participant project associated to the deployed Participant project that you last deployed.
Executable	Identifier of the Control executable that contains the built Participant project that you last deployed.
Controller	Name of the topological entity to which you last deployed the built Participant project.
Last Deployed On	Date and time when you last deployed the Participant project to the controller or you last completed the Deploy Changes / Undo Online Changes command, independently if there changes were deployed or not.
Last Online Change Performed On	Date and time when you last changed the deployed Participant project by using the Refine Online command; otherwise the field is empty.
Last Sync. Performed On⁽¹⁾	Date and time when you last synchronized the standby controller with the primary controller by using the Synchronize (Primary→Standby) command; otherwise the field is empty.
(1) Displayed only for a redundant controller.	
(2) For a redundant controller, the dialog box shows separate columns for the primary and the standby controllers.	

Making Changes Online

To make changes to the deployed Control Participant project online, proceed as follows.

Step	Action	Result
1	Ensure that the engine is connected to the computer, page 803 and that it is running or stopped.	–
2	In the Topology Explorer , right-click the controller and select Refine Online .	The Confirm Refine Online dialog box opens.
3	Verify the information that is displayed.	–
4	Click OK .	The Reconfirm Refine Online dialog box opens.
5	Click OK . NOTE: Click Cancel to close revert to the Confirm Refine Online dialog box without changing the Participant project.	The Control Participant opens.
6	Proceed with the changes or the import of sections in the deployed Control project in the Control Participant (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>).	–
7	Close the Control Participant window.	If deploying these changes does not require stopping the controller, the Refine Online dialog box opens prompting you to update the associated logical Control Participant project.
8	Click Yes . NOTE: Click No to close the Refine Online dialog box without updating the logical Control Participant project and display an online refinement icon, page 529 on the associated topological entity.	The Update Project dialog box opens and allows you to select changes made online that you can transfer to the associated logical Control Participant project and controller configuration.
9	Follow the procedure to update logical Control Participant projects, page 897 and controller configurations.	–

NOTE: If you close the Control Participant window without making changes online, the **Refine Online** online dialog box opens prompting you to update the associated Control project if you previously made changes online without transferring them to the associated logical Control Participant project. The online refinement icon is also displayed on the associated topological entity to indicate it.

WARNING

UNINTENDED EQUIPMENT OPERATION

Before deploying to an engine, verify that you have selected the correct project files.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

Ensure that you are connected to the controller you intend to deploy the Control Participant project.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: For more information, refer to [Identifying Controllers](#), page 796.

⚠ WARNING**UNKNOWN OPERATIONAL STATE OF EQUIPMENT**

Always positively confirm that there is no mission-critical operation in progress before starting or stopping a controller, or any other command or action incompatible with the ongoing operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

When you deploy changes to a redundant controller, the synchronization of the standby controller may not complete. In this case, a message is displayed in the notification panel to inform you. For more information, refer to [Redundant Controller Reservation and Connection Scenarios](#), page 892.

⚠ WARNING**UNINTENDED EQUIPMENT OPERATION**

- Ensure that deployment to the primary controller completed successfully.
- Do not operate the primary controller if deployment to the standby controller does not complete successfully.
- Synchronize the standby controller manually by using the **Sync (Primary→Standby)** command and ensure that the synchronization completed successfully.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

When you make changes online, the associated logical and built Control Participant projects do not contain these changes. To retain these changes for the next deployment, integrate them into the Control Participant projects first by updating the logical Control Participant project, [page 897](#) and building it; otherwise, the changes are discarded.

⚠ WARNING**LOSS OF DATA**

Update the associated logical Control Participant project after building and deploying changes made by online refinement.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Backing Up Controller Data

Overview

The **Back Up Data** command lets you back up and save to file (.DTX) the following controller data:

- Located data of boolean type (%M).
- Located data of WORD type (%MW).
- Unlocated variables.
- Unlocated function block instances (DFB and EFB).

The functionality is available once you have deployed a built Control Participant project to a controller or workstation (controller simulator) of the system.

Data backup files have the following properties:

- They are associated to information, such as a time stamp that lets you select the appropriate file to deploy data again, page 824 and manage backup files, page 296.
- They retain your backup settings facilitating the reuse of the same settings for subsequent backup operations on the same deployed executable.
- They can be exported and imported, page 274 with the Control Participant project.

NOTE: For more information, refer to *Save/Restore Data Between a File and the PLC* in the help of the Control Participant, page 95.

Backing Up Data With Redundant Controllers

For redundant controllers, back up of data is possible only from the primary controller.

To back up data from the primary controller, the standby controller must be reachable at the configured IP address.

You can back up data from the primary controller while the standby controller is not reachable only if the **ControllerExist** property, page 592 is set to **Primary**.

NOTE: The status of the standby controller is not shown in the **Confirm Back Up Data** dialog box.

Backup File Identification

The following information is associated to a backup file:

- A date/time stamp.
- A description entered by the user.
- The user who is logged into the client.
- Names (identifiers) of the controller, the Control Participant project, and the executable.
- The type of data that is backed up.

This information is shown for existing backup files when you select the **Deploy Data** command.

NOTE: If you delete a Control Participant project or a controller, its data backup files are deleted.

NOTE: If you change the name of a controller or executable, or the role of a controller, the identification information of the corresponding backup file is not modified.

Controller and Controller Data Information

Before backing up controller data, the **Confirm Back Up Data** dialog box opens and informs you about:

- The executable whose data you want to back up.
- The associated Control Participant project.
- The controller where the executable is deployed.

This dialog box also lets you select the data that you want to back up.

The following figure shows an example of the **Confirm Back Up Data** dialog box for a standalone controller.

StationNode_3: Confirm Back Up Data (Control)

Engine Information

Status: Stopped

IP Address: Slot NIC_1 {127.0.0.1}

Project Information

Control Project: ControlProject_1

Executable: ControlExecutable_3

Description:

Controller Data

☒ **Located Data**

☒ %M From: 0 To: 512

☒ %MW From: 0 To: 2048

☒ **Unlocated Data**

☒ Variable instances

☒ Function block instances (DFB and EFB)

OK Cancel

The following figure shows an example of the **Confirm Back Up Data** dialog box for a redundant controller.

The following table describes the information that is displayed in the **Engine Information** section of the **Confirm Back Up Data** dialog box.

Label	Description
Controller⁽¹⁾	You can perform the backup only from the primary controller. Default value: Primary
Role⁽¹⁾	Role of the selected controller: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the controller, the role is indicated as Not Detected .
Status	Status of the controller that is selected. If the status is Not Reachable , you cannot back up data. In this case, verify the IP address, the connection to the controller, and the status of the controller. NOTE: If the status is Running , a notification is displayed in the dialog box to inform you about a possible desynchronization of the backed up data.
IP Address	Name of the communication module through which the data is backed up and the IP address that is configured in the Topology Explorer . If the topological entity has several communication modules, you can select one of them. Default value: IP address that was selected for the last deployment operation to this controller. NOTE: For M580 controllers and BMENOC**** communication modules, you can only use the main IP address, page 679. NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.
(1) Displayed only for a redundant controller.	

NOTE: To refresh the controller status and attempt to connect to it, click the reconnect button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Back Up Data** dialog box.

Label	Description
Control Project	Identifier of the Control Participant project associated to the executable that you last deployed.
Executable	Identifier of the Control executable that contains the built Participant project that you last deployed.
Description	You must enter a description for the backup file by using free-form text to proceed with the backup.

The following table describes the information that is displayed in the **Controller Data** section of the **Confirm Back Up Data** dialog box.

Section	Description
Located Data	<p>Select the check boxes depending on the type of located variables you want to back up to file:</p> <ul style="list-style-type: none"> • %M: When selected, backs up located boolean variables. You must provide the memory range to be backed up. Default range: From 0 to the maximum configured memory size -1. • %MW: When selected, backs up located variables of WORD data type. You must provide the memory range to be backed up. Default range: From 0 to the maximum configured memory size -1. <p>NOTE: The maximum configured memory size depends on the memory configuration of located data of the controller module to which the Control project is deployed.</p> <p>Selecting/clearing the Located Data check box also selects/clears the other check boxes in this section.</p> <p>Default value for the first data backup for an executable: TRUE (selected). If you already performed a data backup operation for the same executable, your last settings are retained. The settings are reset if you restore a system or import the topology.</p>
Unlocated Data	<p>Select the check boxes depending on the type of located variables that you want to back up to file:</p> <ul style="list-style-type: none"> • Variable instances: When selected, backs up unlocated variable instances. • Function block instances (DFB and EFB): When selected, backs up unlocated function block instances. <p>Selecting/clearing the Unlocated Data check box also selects/clears the other check boxes in this section.</p> <p>Default value for the first data backup for an executable: TRUE (selected). If you already performed a data backup operation for the same executable, your last settings are retained. The settings are reset if you restore a system or import the topology.</p>

Control Project Signature Verification

Before backing up data, EcoStruxure Process Expert compares the Control Participant project that is deployed in the controller with the **last deployed** reference, page 795, which is the Control Participant project residing in the EcoStruxure Process Expert content repository.

If they are identical, the command is executed. Otherwise, a recovery process starts, which may take several minutes to complete.

If a notification about a signature mismatch is displayed, the command cannot be executed. Contact your local Schneider Electric service representative.

Backing Up Controller Data

Backing up data when the controller is in **RUN** state may take several scan cycles and the backed up data may be desynchronized. This adversely impacts program execution when the backed up data is deployed to the controller.

WARNING

UNINTENDED EQUIPMENT OPERATION

Before backing up data from the controller, ensure that the controller state does not jeopardize data integrity when the data is backed up.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: Back up data while the controller is in **STOP** state.

To back up the controller data, proceed as follows.

Step	Action
1	Ensure that the computer is connected, page 803 to the controller.
2	In the Topology Explorer , right-click the entity whose data you want to back up and select the Back Up Data command. Result: The Confirm Back Up Data dialog box opens.
3	In the IP Address field, ensure that the communication module/controller simulator identification is correct.
4	Verify the status of the controller in the Status field. NOTE: Backup is possible only if the status of the controller is indicated as either RUN , STOP , or WAIT (OFFLINE for Hot Standby controllers of the Quantum platform) .
5	Verify the information that is displayed in the Project Information section.
6	Enter a description for the backup file.
7	Configure the data that you want to back up in the Controller Data section.
8	Click OK . Result: The Reconfirm Back Up Data dialog box opens.
9	Click OK . Result: The selected controller data is backed up to file and the status of the backup operation is shown.

Importing Sections from the Logical Control Participant Project

Overview

It is possible to import FBD sections directly from the logical Control Participant project to help speed up the implementation of changes during commissioning. This is achieved by skipping the build and deployment stages of the system engineering life cycle.

The import functionality allows you to import into the deployed Control project, FBD sections of the corresponding logical Control Participant project that you modified since they were last deployed. The modifications must be **generated**, page 429 to be imported.

The time stamp of the last generation of a section, page 349 is used as reference.

Changes made to a section by using the **Refine** command can be imported only if you generated the section afterwards.

You cannot import a section if you refined it by using the **Refine** command after generating it.

Variables and types that are not used anymore after an import are not removed.

NOTE: If the section that you import has the same name as a section in the deployed Control project, it replaces the existing section. As a result, changes you made in the section are overwritten.

NOTICE

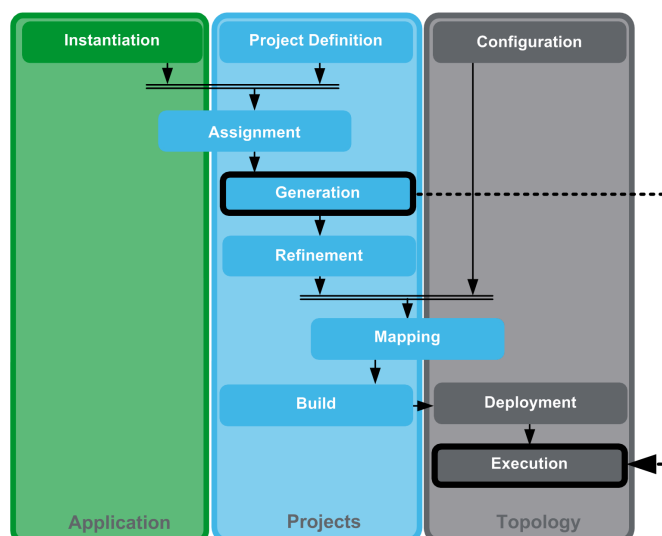
LOSS OF DATA

- Verify that the section that you import does not have the same name as a section in the deployed Control project.
- Modify the section name if you want to preserve the section of the same name in the controller.

Failure to follow these instructions can result in equipment damage.

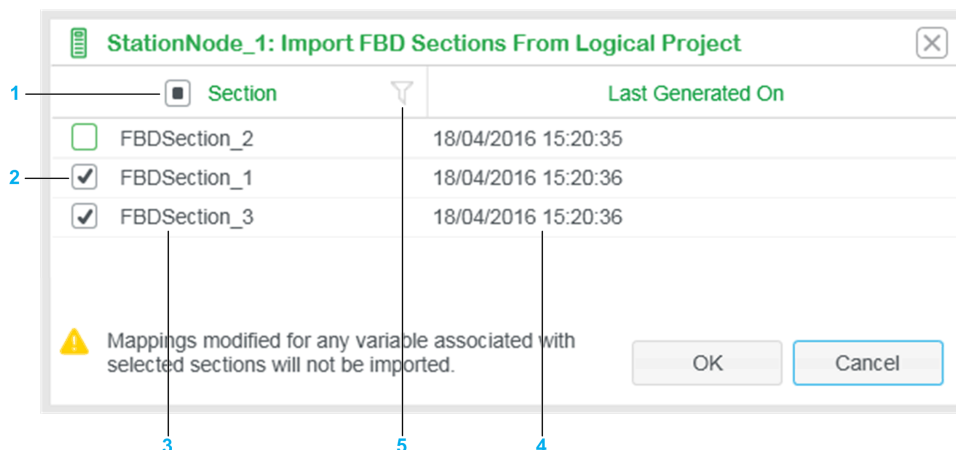
Section Import Functionality in the System Engineering Life Cycle

In the following figure, the dotted line shows the scope of the section import functionality in the context of the system engineering life cycle. Modifications that you make after generating a section and that pertain to the refinement, mapping, and build stages cannot be imported.



Section Import Window

The figure shows an example of the section import window, which opens when you click the section import button in the toolbar of the Control Participant window during online refinement.




Item	Description
1	Check box to select or clear the sections displayed in the import window. A black square inside the check box indicates that at least one section is not selected.
2	Check box to select or clear sections individually.
3	Identifier of sections of the corresponding logical Control project that were modified and generated since the build Control project was last deployed to the engine. A section appears only if the change is supported, page 887 by the section import functionality. Sections are listed in ascending Order (Containers pane of the Assignment Editor).
4	Value of the <i>Generated On</i> property of the section.
5	Filter, page 120 for the Section column.

Attributes Related to HAL and Network Variables

When you import a section that contains a variable generated by a Hardware Abstraction Layer (HAL) template (for example, a signal conditioning template) or peer to peer communication templates (for example, *\$DOUTBool_UL* and *\$DINBool_UL*, which generate a variable with *P2P* as custom attribute), and the section and variable already exist in the deployed Control project, the *Address* and *Value* attributes of the variable are not modified by the import operation.

Online Refinement Indicator

When you import a section, build, and deploy the change, the online refinement

icon, page 529  is displayed and you are prompted to update the associated logical Control project. However, if you did not make other changes apart from importing the section, the logical Control project is already up-to-date. Attempting to update the logical Control project displays a message informing you that no changes are detected.

Changes to Sections That Can Be Imported During Online Refinement

The following table indicates whether a section can be imported during online refinement depending on the change that you made. It is assumed that the section is generated after making the change, if required.

Change description	Allows importing the section	Comment
Assigning a facet to the section.	Yes	–
Unassigning a facet from the section.		–
Actions resulting in a change of the status of a facet.		Except unlinking and relinking a facet.
Regenerating the section (independently if changes were made in the section before regenerating).		–
Renaming the section.		The renamed section is imported as a new section. The section with the original name that existed remains.
Changing the order of sections.		–
Creating a new section without assigning facets.		–
Renaming a variable (for example, by changing the identifier of the instance referencing the facet that creates the variable). The variable and section already exist in the deployed Control project.		The renamed variable is created as a new variable. The variable with the original name that existed remains.
Creating a section and making a change by using the Refine command (for example, adding a function block). Then, making a change in the section that can be imported (for example, assigning a facet).		The section is imported and also contains the change you made through refinement.
Making a change in a section that can be imported (for example, assigning a facet) while a section with the same name already exists in the deployed Control project.	No	The section in the deployed Control project is replaced by the imported section.
Changing the order of facets inside the section.		–
Deleting a section to which facets are assigned and that is generated.		You cannot remove a section from the deployed Control project by using the section import functionality.
Making a change in the section that can be imported (for example, assigning a facet), then making a change in the section by using the Refine command (for example, adding a function block or modifying a variable used in this section).		–
Making changes in a section that can be imported and deploying these changes by using the Deploy Changes / Undo Online Changes command.		The changes are already applied to the deployed Control Participant project. If you made changes that can be imported in more than one section but have only selected some of these sections during deployment of changes, the sections that you have not selected appear as candidates for import.
<ol style="list-style-type: none"> 1. Deploying a section. 2. In the Project Explorer making a change in the section that can be imported (for example, assigning additional facets). 3. Making a change in this section in the deployed Control project by using the Refine Online command and applying it by clicking the Build and Deploy button (for example, removing a DFB). 		You deployed a change to a section that you want to import after generating this section.

Importing Sections from the Logical Control Participant Project

Refer to the topic describing how to refine the deployed Control project online (see *EcoStruxure Process Expert, Control Participant Services, User Guide*).

Redundant Controller Deployment and Executions Scenarios

Overview

This topic describes the sequence of actions that are performed when you execute commands from the context menu of a redundant controller.

The redundant controller is composed of a primary and a standby controller to which IP addresses *IP* and *IP + 1* are assigned respectively.

Other scenarios are described in the topic documenting [redundant controller reservation and connection scenarios](#), page 892.

Deployment and Re-Deployment Sequence of Actions

In this section, various deployment and re-deployment scenarios are described depending on the status of the controllers of the redundant controller before and after successful deployment.

It is assumed that deploying requires stopping the controller.

Scenario 1: Both controllers are stopped. No start after deployment.

	Actions
1	Deployment to <i>IP + 1</i> .
2	Deployment to <i>IP</i> .

Scenario 2: Both controllers are stopped. Controllers are started after deployment.

	Actions
1	Deployment to <i>IP</i> .
2	Start of <i>IP</i> .
3	Synchronization of <i>IP + 1</i> .
4	A successful synchronization starts <i>IP + 1</i> .

Scenario 3: Both controllers are running. No start after deployment.

	Actions
1	Stop of <i>IP + 1</i> .
2	Deployment to <i>IP + 1</i> .
3	Stop of <i>IP</i> .
4	Deployment to <i>IP</i> .

Scenario 4: Both controllers are running. Controllers are started after deployment.

	Actions
1	Stop of <i>IP + 1</i> .
2	Stop of <i>IP</i> .
3	Deployment to <i>IP</i> .
4	Start of <i>IP</i> .
5	Synchronization of <i>IP + 1</i> .
6	A successful synchronization starts <i>IP + 1</i> .

Deployment of Changes and Online Refinement Sequence of Actions

In this section, various scenarios for deployment of changes and online refinement are described depending on the following:

- The status of the redundant controller before and after successful deployment.
- The need to stop the controller.
- The need to disconnect the controller.
- The status of the redundant controller after successful deployment.

Scenario 5: *IP* is stopped. A disconnection is required. No start after deployment.

	Actions
1	Disconnection of <i>IP</i> .
2	Project is rebuilt.
3	Deployment as per deployment scenarios 1 or 2.

Scenario 6: *IP* is stopped. No disconnection is required. Controllers started after deployment.

	Actions
1	Deployment as per deployment scenario 2.

Scenario 7: *IP* is stopped. No disconnection is required. No start after deployment.

	Actions
1	Changes are built in <i>IP</i> .

Scenario 8: *IP* is stopped. No disconnection is required. No start after deployment.

	Actions
1	Changes are built in <i>IP</i> .
2	Deployment to <i>IP</i> + 1.

Scenario 9: *IP* is running. No controller stop is required. No disconnection is required.

	Actions
1	Changes are built in <i>IP</i> .
2	Synchronization of <i>IP</i> + 1.

Scenario 10: *IP* is running. Controller stop is required. No disconnection is required. Controllers are started after deployment.

	Actions
1	Deployment as per deployment scenario 4.

Scenario 11: *IP* is running. No disconnection is required. Controller stop is required. No start after deployment.

	Actions
1	Stop of <i>IP</i> + 1.
2	Stop of <i>IP</i> .
3	Changes are built in <i>IP</i> .

Scenario 12: *IP* is running. A disconnection is required. Controller stop is not required. No start after deployment.

	Actions
1	Disconnection of <i>IP</i> .
2	Project is rebuilt.
3	Deployment as per deployment scenarios 3 or 4.

Redundant Controller Reservation and Connection Scenarios

Overview

This topic describes the actions that are performed on the primary and standby controllers for each deployment and execution command depending on the following:

- The reservation state of the controller.
- The status of the connection to the controller.
- The status of the controller.

It is considered that the redundant controller is composed of a primary and a standby controller to which IP addresses *IP* and *IP + 1* are assigned respectively.

Deploy Built Project and Re-Deploy Last Project Commands

The following table describes the actions that are performed on the primary and standby controllers when you select the **Deploy Built Project** and **Re-Deploy Last Project** commands.

Controller external-ly reserved	Controller <i>IP</i> connected (Controller State)	Controller <i>IP + 1</i> connected (Controller State)	Action executed on controller		Comment
			Primary	Stand-by	
primary	true	indifferent	no action		Primary controller is reserved.
standby	true (stopped)	true	deployment	no action	Standby controller is reserved.
standby	true (running)	true	no action		Standby controller is reserved and cannot be stopped.
none	false	indifferent	no action		Unable to connect to controller <i>IP</i> .
none	true (running)	false	no action		Unable to connect to controller <i>IP + 1</i> .
none	true (stopped)	false	deployment	no action	Unable to connect to controller <i>IP + 1</i> .
none	true	true	deployment to both		<p>If the controllers must be stopped for the deployment, the primary controller is stopped even if the standby controller is already stopped, given that:</p> <ul style="list-style-type: none"> • None of the controllers is reserved • The connection status for both controllers is true

Deploy Changes/Undo Online Changes Command

The following table describes the actions that are performed on the primary and standby controllers when you select the **Deploy Changes/Undo Online Changes** command.

Control- ler external- ly reserved	Controller <i>IP</i> connected (Controller State)	Controller <i>IP</i> + 1 connected (Controller State)	Action executed on controller		Comment
			Primary	Stand- by	
primary	true	indifferent	no action		Primary controller is reserved.
none	false	indifferent	no action		Unable to connect to controller <i>IP</i> .
standby	true	indifferent	no action		Standby controller is reserved.
none	true (stopped)	indifferent	deployment of changes	no action	No synchronization of standby controller because primary controller is stopped.
none	true (running)	true	deployment of changes		If the controllers must be stopped for the deployment of changes, the primary controller is stopped even if the standby controller is already stopped, given that: <ul style="list-style-type: none"> • None of the controllers is reserved • The connection status for both controllers is true

Start Command Only

The following table describes the actions that are performed on the primary and standby controllers when you select the **Start** command.

Control- ler external- ly reserved	Controller <i>IP</i> connected (Controller State)	Controller <i>IP</i> + 1 connected (Controller State)	Action executed on controller		Comment
			Primary	Stand- by	
primary	true (running)	indifferent	no action		Primary controller is already running.
primary	true (stopped)	indifferent	no action		Primary controller is reserved.

Refer to For the Start and Stop Command, page 894 for additional scenarios.

Stop Command Only

The following table describes the actions that are performed on the primary and standby controllers when you select the **Stop** command.

Control- ler external- ly reserved	Controller <i>IP</i> connected (Controller State)	Controller <i>IP</i> + 1 connected (Controller State)	Action executed on controller		Comment
			Primary	Stand- by	
primary	true (stopped)	indifferent	no action		Primary controller is already stopped.
primary	true (running)	indifferent	no action		Primary controller is reserved.
none	true (running)	true (stopped)	Stop	no action	
standby	true (running)	true (running)	no action		Standby controller is reserved and cannot be stopped. Stopping primary controller causes switchover.
none	true (running)	false	no action		Unable to connect to controller <i>IP</i> + 1.
none	true (running)	true (running)	stop		Controllers are stopped in the following order: <ul style="list-style-type: none"> • The standby controller • The primary controller

Refer to For the Start and Stop Command, page 894 for additional scenarios.

Start and Stop Commands

The following table describes the actions that are performed on the primary and standby controllers when you select the **Start** or **Stop** command.

Control- ler external- ly reserved	Controller <i>IP</i> connected (Controller State)	Controller <i>IP</i> + 1 connected (Controller State)	Action executed on controller		Comment
			Primary	Stand- by	
standby	true	true	no action		Standby controller is reserved.
none	false	indifferent	no action		Unable to connect to controller <i>IP</i> .
none	true	false	no action		Unable to connect to controller <i>IP</i> + 1.

Refine Online Command

The following table describes the actions that are performed on the primary and standby controllers when you select the **Refine Online** command.

Control- ler external- ly reserved	Controller <i>IP</i> connected (Controller State)	Controller <i>IP</i> + 1 connected (Controller State)	Action executed on controller		Comment
			Primary	Stand- by	
primary	true	indifferent	no action		Primary controller is reserved.
standby	not required	true	no action		Standby controller is reserved.
Standby-Later ⁽¹⁾	true	indifferent	no action		Standby controller is reserved.
Standby-Later ⁽²⁾	true	indifferent	no action or deployment	no action	No synchronization of standby because reserved. Changes may have been deployed to primary controller.
none	false	indifferent	no action		Unable to connect to controller <i>IP</i> .
none	true	true	deployment of project changes		If the controllers must be stopped for the deployment of online changes, the primary controller is stopped even if the standby controller is already stopped, given that: <ul style="list-style-type: none"> None of the controllers is reserved The connection status for both controllers is true

(1) The controller is reserved after the Control Participant window opens in the **Topology Explorer**.

(2) The controller is reserved after the Build & Deploy, page 856 sub-command is executed.

Sync (Primary→Standby) Command

The following table describes the actions that performed on the primary and standby controllers when you select the **Sync (Primary→Standby)** command.

Control- ler external- ly reserved	Controller <i>IP</i> connected (Controller State)	Controller <i>IP</i> + 1 connected (Controller State)	Action executed on controller		Comment
			Primary	Standby	
primary	true	indifferent	no action		primary controller is reserved.
standby	true	true	no action		standby controller is reserved.
none	false	indifferent	no action		Unable to connect to controller <i>IP</i> .
none	true	false	no action	synchron- ize standby	–

Updating the Logical Control Participant Project and Controller Configuration

Overview

The **Update Project** command allows you to transfer to the logical Control Participant project and the controller configuration changes you made online, page 870 by using the **Refine Online**, page 874 command.

An online refinement icon, page 529 displayed on the associated topological entity indicates that online changes can be transferred.

You can update the logical Control Participant project with changes pertaining to the following categories:

- Animation tables
- Operator screens
- Sections
- Types
- Variables
- Hardware

Applicable restrictions are described in this topic.

In addition for Control projects of M580 safety controllers, page 908, you can deploy changes related to:

- Process (not safety-related) program interfaces
- The Safe program

When you select the **Deploy changes / Undo Online Changes** command, EcoStruxure Process Expert compares the Participant project that is deployed in the controller with the corresponding logical Participant project and controller configuration and identifies differences.

You can select which of the identified changes you want to apply.

You perform a consistency check before updating the logical Participant project.

If you changed the logical Participant project since the deployment, using the **Update Project** command may override some of these changes.

NOTE: You must build the Participant project, page 774 to which you have transferred changes by using the **Update Project** command to deploy these changes.

Confirming the Command

When you select the **Update Project** command, the **Confirm Update Project** dialog box opens and displays information about:

- The controller from which you are updating the Control Participant project.
- The Control Participant project that is deployed to the controller.

StationNode_1: Confirm Update Project (Control)

Engine Information

Status: Running

IP Address: Slot NIC_1 {127.0.0.1}

Project Information

Name	Selected Project
Executable	Sim
Controller	StationNode_1
System	CCOTF
Project	M5_14_1_CP1
Last Deployed On	25/11/2020 12:53:26
Last Online Change Performed On	25/11/2020 12:57:42
Last Control Project Modification On	25/11/2020 12:32:29

⚠ Are you sure you want to Update Project?

OK Cancel

The following figure shows an example of the **Confirm Update Project** dialog box for a redundant controller.

R_04: Confirm Update Project

Engine Information

Role: A B

Status: Running Primary Running Standby

IP Address: Slot 0:BMEH584040 {173.20.4.10} Slot 0:BMEH584040 {173.20.4.11}

Project Information

Name	Selected Project
Executable	R_04_1_EX
Controller	R_04
System	CCOTF
Project	R_04_1_CP
Last Deployed On	25/11/2020 14:10:02
Last Online Change Performed On	25/11/2020 14:11:14
Last Control Project Modification On	
Last Sync Performed On	N/A

⚠ Are you sure you want to Update Project?

OK Cancel

The following table describes the information that is displayed in the **Engine Information** section of the **Confirm Update Project** dialog box.

Label	Description
Role⁽¹⁾	Role of the controller: <i>A</i> or <i>B</i> . If EcoStruxure Process Expert cannot communicate with the controller, the role is indicated as Not Detected .
Status	Status of the controller that is selected. If the status is Not Reachable you cannot proceed with the update. In this case, verify the IP address, the connection to the controller, and the status of the controller.
IP Address	Name of the communication module through which the command is sent and the IP address that is configured in the Topology Explorer . If the controller has several communication modules, you can select one of them. NOTE: For a workstation, it shows the Network Interface Card (NIC) and the port of the Control service that are mapped to the executable.
(1) Displayed only for a redundant controller.	

NOTE: To refresh the controller status and attempt to connect to it, click the **reconnect** button, page 803.

The following table describes the information that is displayed in the **Project Information** section of the **Confirm Update Project** dialog box.

Name column	Selected Project column
System	Identifier of the system that the controller belongs to.
Project	Identifier of the Control Participant project associated to the deployed Participant project.
Executable	Identifier of the Control executable that contains the built Participant project.
Controller	Name of the topological entity from which you are updating the logical Control Participant project.
Last Deployed On	Date and time when you last deployed the Participant project to the controller or you last completed the Deploy Changes / Undo Online Changes command, independently if changes were deployed or not.
Last Online Change Performed On	Date and time when you last changed the deployed Participant project by using the Refine Online command; otherwise the field is empty.
Last Control Project Modified On	Date and time when you last modified the logical Control Participant project by using the Generate or Refine command.
Last Sync. Performed On⁽¹⁾	Date and time when you last synchronized the standby controller with the primary controller by using the Synchronize (Primary→Standby) command; otherwise the field is empty.
(1) Displayed only for a redundant controller.	

NOTE: The **Confirm Update Project** dialog box is not displayed if you choose to update the logical Control Participant project when prompted after completing an online refinement operation, page 875.

Updating the Logical Control Project With Changes Other Than FBD Language

The following table describes the effect of the **Update Project** command on items other than FBD language.

Item	Effect
Non-FBD sections	<p>Details of changes inside non-FBD sections that are detected are not shown. Only the name of the section is displayed.</p> <p>When selected, these sections are either updated, added, or removed from the logical Control Participant project.</p> <p>NOTE: If you update the logical Control Participant project with code in LL984 language added during online refinement and you did not select the Ladder Logic 984 (LL984) setting in the Control Participant before deploying, the validity icon, page 336 is displayed after the update.</p> <p>To build the logical Control Participant project, select the Ladder Logic 984 (LL984) setting in project containing the configuration of the controller (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>) that is mapped to the executable of this Control project.</p>
Animation tables	Changes to these items are not detected.
Operator screens	<p>Animation tables and/or operator screens that the deployed Participant project contains are shown, including modified and/or new ones. Animation tables and/or operator screens that you deleted from the deployed Participant project are not shown.</p> <p>You cannot select individual items but only the category to transfer it to the logical Participant project.</p> <p>When you select a category, animation tables or operator screens of the logical Control Participant project are overwritten with those contained in the deployed Participant Project.</p> <p>By default, both categories are unselected.</p> <p>NOTE: Overwriting an existing operator screen creates a copy of the screen in the deployed Participant project. Following deployment, use the Refine Online command to edit the categories and make their content consistent with the Control Participant project.</p> <p>NOTE: Animation tables and/or operator screens that you removed from the Participant project are not removed from the deployed Participant project by using the command. Use the Refine Online command to delete them.</p>
Hardware	<p>Details of configuration are not shown.</p> <p>The Hardware category is selected by default (if changes are detected) and is read-only.</p> <p>If the controller configuration is modified because of the update operation, mappings may be deleted.</p> <p>NOTE: If you changed the configuration online you must perform the update to use the Deploy changes / Undo Online Changes command.</p>

Updating the Logical Control Project With Changes to Variables

The following table describes the result when you update a logical Control Participant project depending on the changes that you made to variables contained in the deployed Control Participant project.

Variables in the Deployed Control Project	Result
Variables with HAL , page 784 or P2P , page 467 as Custom attribute.	No changes to their attributes are updated in the logical Control Participant project.
Variables without HAL , page 784 as Custom attribute.	Attributes of the variables that you add or modify during online refinement are updated in the logical Control Participant project, except for the Address or Value attributes if it corresponds to or references a topological address. NOTE: For more information on variables using the Value attribute for topological addressing, refer to the topic describing REF_TO type variables (see <i>EcoStruxure Process Expert, Control Participant Services, User Guide</i>).
Device DDT variables	These variables (and their attributes) coming from the configuration are not transferred to the logical Control Participant project. If you add code during refinement online, page 870 that uses this type of variable, you can update the logical Control Participant project with this code but not with the variable itself. This results in invalid code. NOTE: You can create the Device DDT variable in the logical Control project by using the wizard, page 455.

Updating the Control Logical Project With Changes to Sections Containing HAL DFBs

You cannot transfer to the logical Control Participant project changes that you made or code that you added to sections containing HAL DFB instances, page 784.

Changes That Cannot Be Transferred to the Logical Control Project

You cannot transfer to the logical Control Participant project changes pertaining to the following categories:

- Control project settings (refer to *EcoStruxure™ EcoStruxure Control Expert, Operating Modes*).
- SR sections.
- Properties of MAST and FAST tasks.
- Properties of MAST and FAST sections, except for changes in the **Comment** tab.
- Timer and I/O event tasks.
- *Alias of* attribute of variables for Modicon X80 I/O modules (reference to the Device DDT variable).

Identifying Project and Hardware Changes

Before proceeding with the update, the **Update Project** dialog box opens, which indicates the differences that it has identified between the logical and the deployed Participant project. The dialog box allows you to select the changes that you want to transfer to the logical Control Participant project.

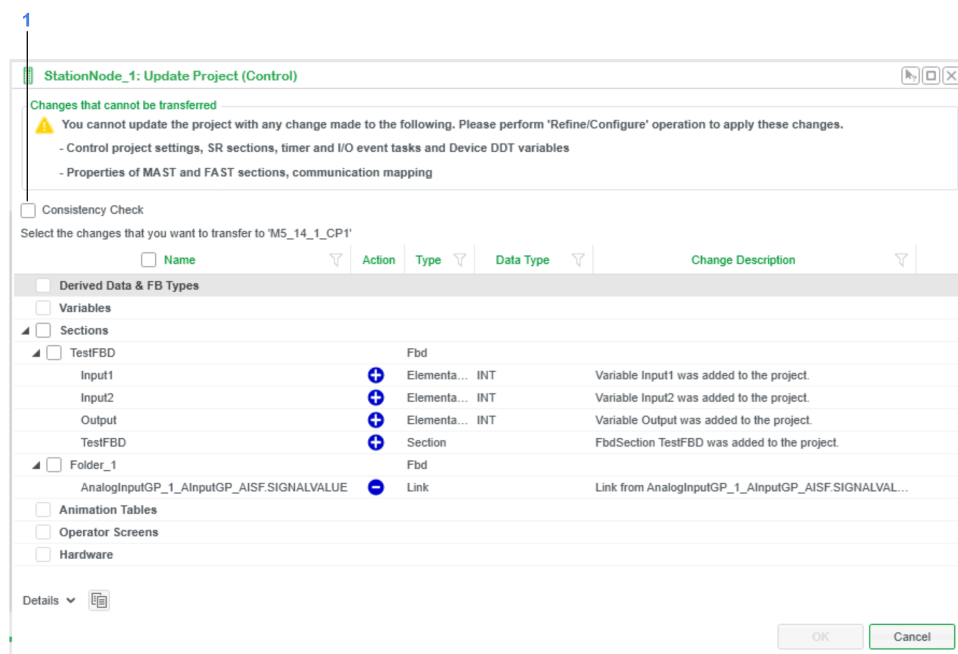
You can only transfer to the logical Participant project changes that pertain to a category displayed in the **Name** column of the **Update Project** dialog box. Changes that are not detected or that are detected but cannot be selected are not transferred.

If no change is detected, the **Update Project** dialog box opens and informs you that no changes were found.

NOTE: If you select a section in the logical Control Participant project in order to apply changes to this section, EcoStruxure Process Expert overrides the refinements that you made in this section. Refer also to the topic describing the impact of a Control project update, page 904 on sections that you created after deployment.

NOTE: Restrictions may apply when updating variables, page 901.

The following figure shows an example of the **Update Project** dialog box.



1 Check box to perform a consistency check, page 903 on the selected items before updating the logical Control Participant project. By default, the option is unselected.

Header	Description
Check boxes	<p>Select items that you want to update in the logical Control Participant project.</p> <p>A green square inside the check box in the column header indicates that, at least, 1 item is unselected.</p> <p>NOTE: You cannot select sections that contain HAL DFB instances, page 784.</p>
Name	<p>Identifier of the item that was detected as a difference.</p> <p>Items are grouped by categories of the Control Participant project.</p>
Action	<p>Indicates the action that are performed for the item:</p> <ul style="list-style-type: none"> Plus icon: The item is added to the logical Participant project/controller configuration. (Because the item was added during online refinement or removed from the logical Participant project after the last deployment.) Minus icon: The item is removed from the logical Participant project/controller configuration. (Because the item was removed during online refinement or added to the logical Participant project after the last deployment.) Pencil icon: The item is modified in the logical Participant project. <p>NOTE: When you select a section, changes that were identified for this section are applied. You cannot select/unselect changes individually inside a section.</p>
Type	Type of the item.
DataType	Data type of the item, if applicable.
Change Description	For FBD sections, variables and types, indicates whether the change that was identified is an addition, a modification, or a deletion compared to the logical Participant project.

Verifying the Consistency of the Control Project

Select the corresponding check box in the **Update Project** dialog box to perform a consistency check, page 460 before updating the logical Participant project.

In this case, the **Check Consistency** dialog box opens and shows the following for the changes that you have selected in the **Update Project** dialog box:

- The sections that contain constituents that are inconsistent.
- For each section, the facets that are assigned to the section and generate the corresponding constituents.
- For each facet, a description of the inconsistency.

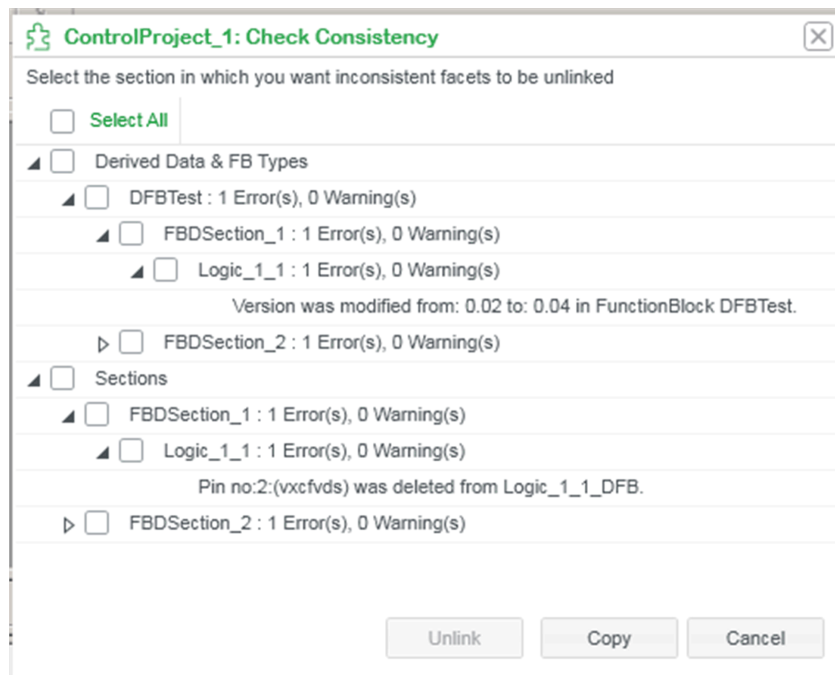
Select the check box next to a facet to **unlink**, page 424 it.

If the check box of a facet is cleared, the generation status of the facet remains **Inconsistent**, page 424.

During a subsequent generation, page 430 of the updated logical Participant project, EcoStruxure Process Expert proceeds as follows:

- It does not modify constituents of facets that are unlinked.
- It attempts to restore constituents of facets that are inconsistent.

By default, items are unselected.



NOTE: If no inconsistencies are detected, the **Check Consistency** dialog box does not open.

Updating After Creating Sections in the Logical Control Project Without Deploying Them

After updating the logical Control project, a section is moved to the end of sections in this project (highest order value) if the following conditions are fulfilled:

- The logical Control project is deployed.
- You create a section either in the **Project Explorer** or during refinement (offline) post deployment.
- You do not deploy this section.
- The section that you created occupies the same position as a section in the deployed Control project or you select the section in the **Update Project** dialog box (delete).

It may also have an impact on facets that are assigned to this section.

The following example shows how a section created in the logical Control project after deployment can have the same position as a section in the deployed Control project.

In this example, the logical Control project contains two sections and it is deployed.

Logical Control project (offline)	Deployed Control project (online)
Section_1 / order 0	Section_1
Section_2 / order 1	Section_2

Then, section_3 is created in between Section_1 and Section_2. It has the same position as Section_2 in the deployed Control project.

Logical Control project (offline)	Deployed Control project (online)
Section_1 / order 0	Section_1
Section_3 / order 1	Section_2
Section_2 / order 2	

The following table describes the possible scenarios where updating the logical Control project has an impact on the position of a section that you created in it post deployment and/or on the status of the facets assigned to this section.

Assignment and generation status	The position of the section conflicts with the position of a section in the deployed project	Proposed action during Control project update	User selection	Result in logical Control project after the update
Facets are assigned to the section but none are generated.	Yes	None. The section created in the logical project is not detected as a difference because it is not generated.	Not applicable.	The section that was created in the logical project is moved to the end of sections (highest order value) ⁽¹⁾⁽²⁾ .
Facets are assigned to the section and generated.	No	Delete.	The section is selected for deletion.	<ul style="list-style-type: none">The section that was created in the logical project is moved to the end of sections (highest order value)⁽²⁾.The generation status of facets is changed to Non Generated.
	Yes	Delete.	The section is selected for deletion.	<ul style="list-style-type: none">The section that was created in the logical project is moved to the end of sections (highest order value)⁽²⁾.The generation status of facets is changed to Non Generated.
		Delete.	The section is not selected (default).	<ul style="list-style-type: none">The section that was created in the logical project is moved to the end of sections (highest order value)⁽¹⁾.The generation status of facets is unchanged.
<p>(1) After the update is complete, you may be able to preserve the change that you made in the logical Control project by moving the section back to its original position and propagating the change in the system engineering life cycle.</p> <p>(2) To remove the section from the logical Control project, after the update is complete, right-click the section in the Containers pane of the Assignment Editor and select Delete.</p>				

NOTE: In the above scenarios, it is implied that other changes that you may have made during refinement online are applied to the logical Control project (for example, adding code to a section or changing the order of sections).

NOTE: The behavior is the same when you update the logical Control project if you create a section in the **Project Explorer**, generate, and deploy it and then, delete it during refinement online (given the section is not locked, page 452).


Aborting Project Update Tasks

After confirming the update by clicking **OK** in the **Update Project** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. The project is not updated.

Only the user who selected the command is allowed to cancel it.

Updating the Logical Control Project and the Controller Configuration

To update the logical Control Participant project and/or the controller configuration, proceed as follows.

Step	Action
1	Ensure that the computer is connected, page 803 to the controller.
2	<p>In the Topology Explorer, right-click the controller or workstation containing the Participant project that you refined online and select Update Project.</p> <p>Result: The Confirm Update Project dialog box opens.</p> <p>NOTE: For a workstation, you can select a Control project from a list of projects containing online changes (if applicable) in the Update Project dialog box.</p>
3	<p>Verify the information that is displayed and click OK.</p> <p>Result: The Update Project dialog box opens.</p>
4	Select the changes that you want to transfer to the logical Control Participant project.
5	Select the Check Consistency check box if you want to perform a consistency check; otherwise proceed to step 8.
6	<p>Click OK.</p> <p>Result: The Check Consistency dialog box opens and inconsistencies are identified.</p>
7	<p>In the Check Consistency dialog box, select facets that you want to unlink.</p> <p>NOTE: Click Cancel to revert to the Update Project dialog box without unlinking facets.</p>
8	<p>Click Unlink or OK.</p> <p>Result:</p> <ul style="list-style-type: none"> • The changes that you selected are applied. • The status of the associated executable changes to Out Of Date. • Completed is displayed in the information pane once the update process is successfully executed. • The related date/time field is updated. • The online refinement icon, page 529  is removed from the associated topological entity only if you selected to update the Control Participant project with all the detected changes of the following categories; otherwise the indicator remains. <ul style="list-style-type: none"> ◦ Derived Data & FB Types ◦ Variables & FB Instances ◦ Sections

Supervision Project Execution Stage

Executing the Deployed Supervision Participant Project

Overview

You need to execute the deployed Supervision Participant project locally, on each computer to which you have deployed the Supervision project by using the appropriate Supervision runtime.

For information on the required software, refer to the topic describing software requirements (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

Executing a Deployed Supervision Project

For information on executing the deployed Supervision Participant project on a computer, refer to Executing the Deployed Supervision Project (see *EcoStruxure Process Expert, Supervision Participant Services, User Guide*).

Engineering M580 Safety Systems

What's in This Chapter

Using Modicon M580 Safety Controllers.....	909
M580 Safety Systems in the Instantiation Stage.....	911
M580 Safety Systems in the Project Definition Stage	912
M580 Safety Systems in the Assignment Stage	919
M580 Safety Systems in the Generation Stage	920
M580 Safety Systems in the Refinement Stage	921
M580 Safety Systems in the Configuration Stage.....	923
M580 Safety Systems in the Mapping Stage.....	929
M580 Safety Systems in the Build Stage	930
M580 Safety Systems in the Deployment Stage.....	931
M580 Systems in the Execution Stage	932
Using Supervision with M580 Safety Systems	934

This chapter describes the aspects that are specific to using an M580 safety controller and its associated Control project (the *M580 safety system*) when engineering a system with EcoStruxure Process Expert. The information is grouped by stages of the system engineering life cycle.

Using Modicon M580 Safety Controllers

Certifications

The M580 safety controller is certified by TÜV Rheinland Group for use in applications up to and including safety integrity level 3 (SIL3).

For additional information, refer to the topic describing certifications in *Modicon M580 Safety Manual*, which you can find in the help of the Control Participant, page 95.

Control Participant of EcoStruxure Process Expert

EcoStruxure Process Expert 2023 uses Control Expert XL Safety 15.3 as Control Participant.

The safety-related configuration and safety-related program of an M580 safety system are created and modified exclusively by the Control Participant.

Only the process (not safety-related) areas and the not safety-related modules of the M580 safety system are managed by EcoStruxure Process Expert.

Engineering Systems Containing M580 Safety Controllers

When you engineer an EcoStruxure Process Expert system that contains an M580 safety controller, in addition to following the system engineering life cycle, perform the following activities:

- Ensure that the reference and version of the Control Participant is certified by the TÜV.
You can look up the TÜV-certified software tools and versions in the *Revision List*, which is in the appendices of the *M580 Functional Safety Controller System* certificate at www.certipedia.com or www.fs-products.com.
- Use and monitor the Safe Signature and the safety application ID (SAId).
For details, refer to the topic describing building an M580 safety-related project in *Modicon M580 Safety Manual*.
- Follow the Safe application life cycle and apply the safety rules as described in *Modicon M580 Safety Manual*, which you can find in the help of the Control Participant, page 95.

Information related to the following topics that are described in *Modicon M580 Safety Manual* does not apply or applies only partially when working with the Control Participant.

Topic	Difference when using EcoStruxure Process Expert
Usage of Process Namespace project setting (Refer to <i>Data Separation in an M580 Safety Project</i>)	The project setting, page 921 is not supported.
Access rights (Refer to <i>Workstation Security Management</i>)	User profiles and access rights for the Control Participant are managed by EcoStruxure Process Expert and cannot be modified. Users who can log into the engineering client have full access to the functionality of the Control Participant.
Maintenance mode (Refer to <i>M580 Safety PAC Operating Modes</i>)	You can switch the M580 safety controller to maintenance mode by using the button in the toolbar of the Participant window when you use the Refine Online command.
Building Control projects of M580 safety PACs	Use a build command in the context menu of the executable of the Control project in the Project Explorer .

Topic	Difference when using EcoStruxure Process Expert
(Refer to <i>Operation of an M580 Safety System</i>)	
<i>Safe signature</i> (Refer to <i>Building an M580 Safety Project</i>)	Using a build command, page 930 in the context menu of the executable of the Control project of an M580 safety controller generates a new <i>SourceSafeSignature</i> if you have made an applicable configuration or value change.
<i>Safe Area Password Protection</i>	The password protection, page 925 is managed in the Topology Explorer instead by using the Safety password.
Transferring M580 safety-related projects (Refer to <i>modifications to Control Expert for the M580 safety system</i>)	The command to transfer the project from the controller to the Control Participant is disabled. To transfer the project to the controller, use only the deployment commands in the Topology Explorer .
<i>CCOTF for an M580 Safety PAC</i> (Refer to <i>modifications to Control Expert for the M580 safety system</i>)	CCOTF operations can be performed also when you configure, page 585 an M580 safety controller (offline).
<i>Saving & Restoring Data Between a File and the PAC</i> (Refer to <i>modifications to Control Expert for the M580 safety system</i>)	The commands are disabled in the Control Participant. Use the Back Up Data and Deploy Data commands in the Topology Explorer instead.

Additional Information

For more information on M580 safety systems, refer to *Modicon M580 Safety System Planning Guide*, which you can find in the help of the Control Participant, page 95.

M580 Safety Systems in the Instantiation Stage

This topic describes the aspects of the instantiation that are specific to using M580 safety systems. For general information, refer to the topic describing the instantiation stage.

Systems Explorer

You can use M580 safety controllers and their projects in any EcoStruxure Process Expert system.

A system can contain both safety and not safety-related controllers and the corresponding Control projects.

Instantiation Stage

No safety-related templates are provided to create application instances in a system that uses an M580 safety controller. The safety-related data and logic can be created only by refining the Control project of an M580 safety controller.

Instances that you create in the **Application Browser** can only be used to generate the following:

- The process data and logic in the Control project of an M580 safety controller.
- Supervision project data.

M580 Safety Systems in the Project Definition Stage

This topic describes the aspects of the Control project definition that are specific to using M580 safety systems. For general information, refer to the topic describing the Control project definition stage.

Creating Control Projects of M580 Safety Controllers

Control Projects of M580 Safety Controllers

In a system, the Control project of an M580 safety controller can coexist with Control projects of other controller families.

Content Control Projects of M580 Safety Controllers

The Control project of an M580 safety controller can contain the following:

- A safety-related program (**SAFE**)
- A not safety-related program (**PROCESS**)
- Global data

The logic and data that is part of each program is isolated from the other.

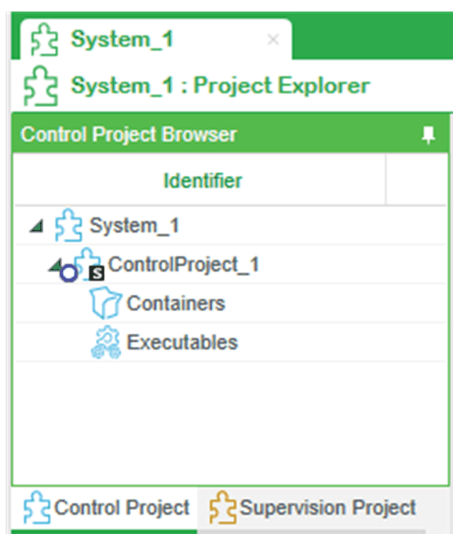
Creating the Control Project of an M580 Safety Controller

To create the Control project of an M580 safety controller, right-click the system root folder in the **Project Browser** and select **M580 Safety**.

Creating this type of project also creates an M580 safety controller with a default configuration in the **Topology Explorer**, which is associated to the project.

Conversely, if you create an M580 safety controller, a blank Control project of an M580 safety controller is created in the **Project Explorer**.

The following figure shows the Control project of an M580 safety controller as it appears in the **Project Browser**.



Working With the Control Project of an M580 Safety Controller

After you create a safety-related Control project in the **Project Explorer**, you must open the configuration of the associated M580 safety controller once by using the

Configure command in the **Topology Explorer** before you can perform actions on the Control project, such as generating or refining it or managing its passwords.

This also applies if the safety-related Control project was created because you created an M580 safety controller in the topology.

Creating Sections, Logic, and Data

The following table indicates how to create the various types of program components of the Control project of an M580 safety controller and where you can view them.

Type of logic and data	Ways to create	Viewing logic and data
Safety-related sections, logic, and data (includes animation tables)	By refining the Control Participant project only.	Visible only in the refinement window (offline and online) of the Control Participant.
Process sections, logic, and data	By following the normal system engineering life cycle. (Non-FBD sections and logic requires refining the Control Participant project.)	FBD and non-FBD sections as well as facets assigned to FBD sections are visible in the Assignment Editor .
Global data	By refining the Control Participant project only.	Visible only in the refinement window (offline and online) of the Control Participant.

Deleting the Control Project of an M580 Safety Controller

Deleting the Control project of an M580 safety controller also deletes the associated M580 safety controller in the **Topology Explorer**, and vice versa.

Managing Network Variables

The **Manage Network Variables** dialog box, page 471 does not let you manage variables of the safety-related program.

Supervision Projects

The Supervision projects that you create are independent of the M580 safety system.

You can create instances to be assigned to Supervision projects to monitor safety-related Control data, page 934.

Exporting and Importing the Control Project of an M580 Safety Controller

This topic describes the aspects of the export and import that are specific to the Control project of an M580 safety controller. For general information about project export and import operations, refer to the topic describing how to [manage Control projects](#), page 274 of a system.

Exporting the Control Project of an M580 Safety Controller

The **Export** command of the **Project Explorer** lets you export one or more safety-related Control projects and create a non-editable export file (.sbk).

You can export Control projects of the other controller families and Supervision projects at the same time.

In the **Export** window, the **Safe Program** section of safety-related Control projects is selected by default and read-only. The safety-related program is exported as a whole.

The associated safety-related configuration is not exported.

Importing the Control Project of an M580 Safety Controller

The **Import** command of the **Project Explorer** lets you import the M580 safety-related Control projects that are contained in a project export file (.sbk).

The following table describes the result when you import an M580 safety-related Control project in different scenarios.

Target system	Result after import in Project Explorer	Impact on the topology after import
The Control project of an M580 safety controller with the same identifier does not exist.	The safety-related Control project is imported. If you select to import the safety-related program, it is imported as a whole.	An M580 safety controller entity is created with the default configuration ⁽¹⁾ . NOTE: If you want to import a safety-related Control project that is associated to an existing M580 safety controller in the Topology Explorer , before importing the safety-related project, rename the existing safety-related Control project, page 917 to have the same identifier as the one you are importing.
The Control project of an M580 safety controller with the same identifier already exists.	If you select to import the safety-related program, it is imported as a whole. The import rules, page 280 that are applied with regard to safety-related sections are the same as those for process sections.	The existing M580 safety controller is not impacted and no additional one is created.
A Control project with the same identifier but of the M580 (not safety-related) or another controller family already exists.	You cannot import the M580 safety-related Control project. In the Import window, it is tagged with a red dot ⁽²⁾ .	None.
<p>(1) Ensure that the Global Templates Library of the target system contains the necessary topological safety-related templates (Foundation Library) before importing. Otherwise, you cannot save your changes when you configure the associated safety controller in the Topology Explorer.</p> <p>(2) The same also applies when you import other Control projects and an M580 safety-related Control project with the same identifier already exists in the target system.</p>		

NOTE: During the import of the safety-related Control project, the associated M580 safety controller is locked.

Importing a Control Expert Project

The **Import Control Project** command of the **Project Explorer** lets you create a new M580 safety-related Control project by using the data of a compatible Control Expert project export file (.stu or .sbk).

The following are created:

- An M580 safety-related Control project in the **Project Explorer**.
- An associated M580 safety controller with the same configuration and project settings as in the export file in the **Topology Explorer**.

The name of the imported file becomes the identifier of the safety-related Control project that is created. The associated M580 safety controller is created with the default identifier.

Functional Safety Password Management When Importing

When you import one or more M580 safety-related Control projects, the **Verify Safety Password** window may open prompting you to enter a functional safety password for each one.

The import operation starts only after the required passwords were entered.

NOTE: The functional safety password that is referred to is the one that has been set in the **Topology Explorer** for the associated M580 safety controller, page 925.

The following table describes which functional safety password you must enter when prompted during the import of an M580 safety-related Control project. This applies even if you select not to import the safety-related program.

Target system	Functional safety password required	Impact on functional safety password after import
An M580 safety-related Control project with the same identifier does not exist.	The functional safety password of the safety-related Control project that you are importing (if set).	The functional safety password settings are the same as that of the imported safety-related Control project.
An M580 safety-related Control project with the same identifier already exists.	The functional safety password of the existing safety-related Control project in the target system (if set).	The functional safety password and password settings are the same as that of the existing safety-related Control project in the target system. For example, if the existing safety-related Control project has no functional safety password but the safety-related Control project that you are importing has a functional safety password, after the import, no functional safety password is set.

The following table describes which functional safety password you must enter when prompted during the import of a Control Expert M580 safety-related Control project.

Target system	Functional safety password required	Impact on functional safety password after import
Irrelevant	The functional safety password of the safety Control project that you are importing (if set).	The functional safety password settings are the same as that of the imported safety-related Control project. You can manage the password from the controller entity in the Topology Explorer .

Importing a Safety-Related Control Project Associated to an Existing M580 Safety Controller

By using an example, this topic illustrates the required workflow to import an M580 safety-related Control project that is associated to an existing M580 safety controller or one that you will be importing. The workflow for M580 safety systems differs from other controller platforms because the import of a safety-related controller/project creates the counterpart safety-related project/controller.

Starting Point

Initially, you have exported the following to two separate export files (.sbk):

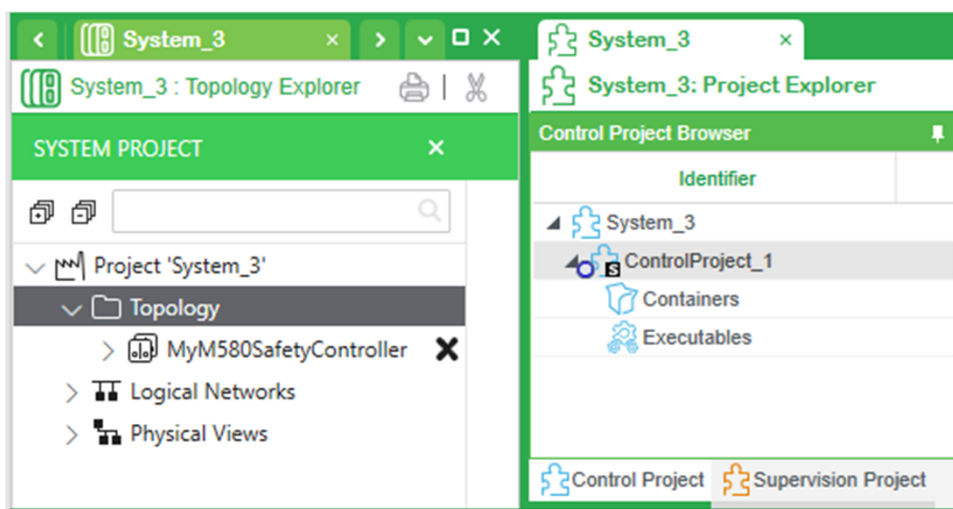
- The configuration of an M580 safety controller *MyM580SafetyPAC* from the **Topology Explorer**
- The associated M580 safety-related Control project *MyM580SafetyProject* from the **Project Explorer**

The executable of *MyM580SafetyProject* is mapped to *MyM580SafetyPAC* in the service mapping and both are exported.

In this example, the objective is to import into a different system the M580 safety controller and its project so that they continue to be associated and the executable of the project remains mapped to the controller.

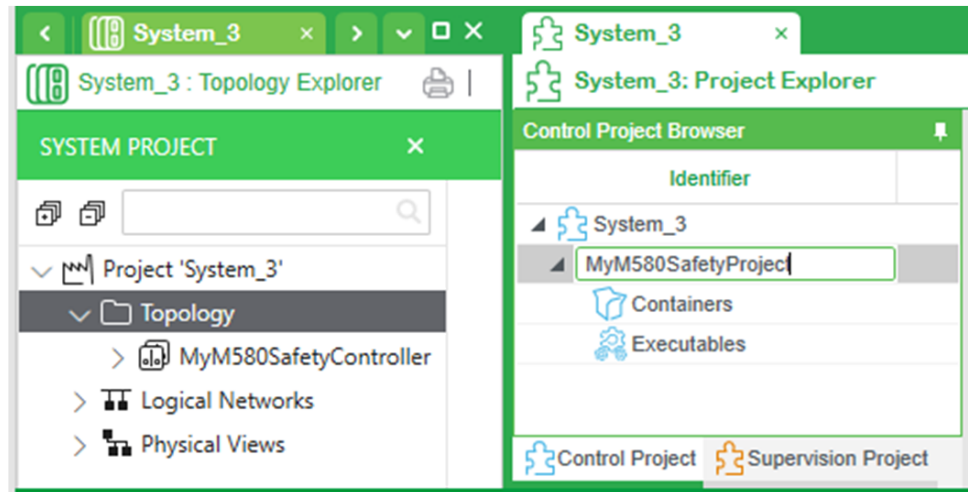
Importing the M580 Safety Controller

After the import of *MyM580SafetyPAC*, a blank safety-related Control project is created. The project is associated to the controller but has the default identifier.



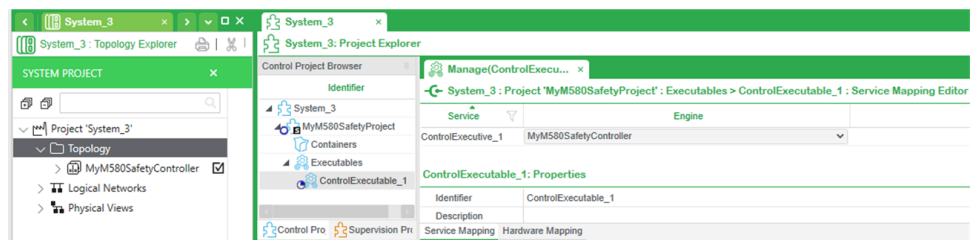
Renaming the Associated M580 Safety-Related Control Project

Rename the associated safety-related Control project so that it has the same identifier as the project that you are going to import.



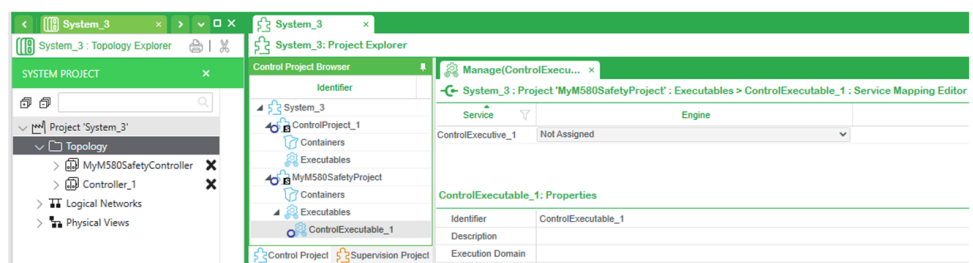
Importing the Associated M580 Safety-Related Control Project

When you import *MyM580SafetyProject* and select its executable and service mapping, it is merged with the safety-related Control project that you have renamed. No new associated safety controller is created. Instead, the service mapping with *MyM580SafetyPAC* is recreated.



Importing the Associated Project Without Renaming

The following figure shows the result after importing *MyM580SafetyProject* if you do not rename *ControlProject_1*. Two pairs of controller/Control project are created and *MyM580SafetyPAC* is not associated nor mapped to *MyM580SafetyProject*.



M580 Safety Systems in the Assignment Stage

This topic describes the aspects of the Control project assignment that are specific to using M580 safety systems. For general information, refer to the topic describing the assignment stage.

Creating and Viewing Sections

You can only create FBD sections of the process program by using the **Assignment Editor**.

None of the sections of the safety-related program that you create in refinement appear in the **Assignment Editor**.

Assigning Facets of Instances

You can only assign facets of instances to FBD sections of the process program by using the **Assignment Editor**.

M580 Safety Systems in the Generation Stage

This topic describes the aspects of the Control project generation that are specific to using M580 safety systems. For general information, refer to the topic describing the generation stage.

Generating M580 Safety-Related Control Projects

The generation process only generates logic and data in the process program of an M580 safety-related Control project in the same way as it is done for other Control projects.

NOTE: Ensure that the associated M580 safety controller is configured first, page 912.

NOTE: While the safety-related Control project is being generated, it is locked. This also locks the associated safety controller.

Regenerating M580 Safety-Related Control Projects

Regenerating a safety-related Control project does not discard changes that you performed in the safety-related program.

The process program is impacted in the same way as the logic and data of not safety-related Control projects.

Impact of Project Generation on the Associated M580 Safety Controller

When you generate a safety-related Control project and you have not yet configured the associated M580 safety controller, the safety controller changes to a configured state in the **Topology Explorer**. As a result, the controller properties that you configured in the **Topology Explorer** are applied to the configuration of the Control Participant.

M580 Safety Systems in the Refinement Stage

This topic describes the aspects of the offline Control project refinement that are specific to using M580 safety systems. For general information, refer to the topic describing the refinement stage.

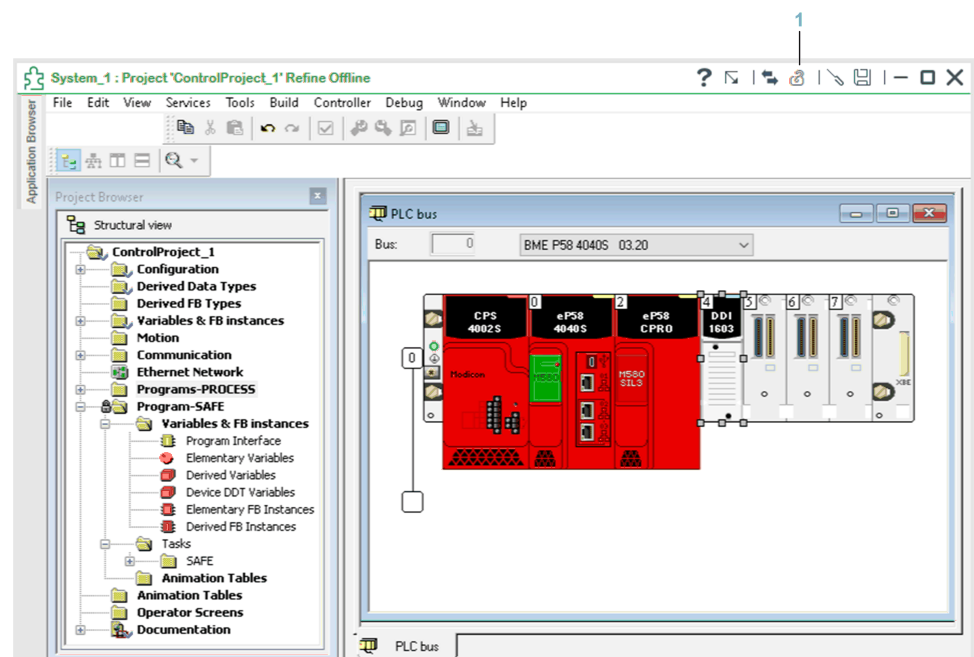
Refining M580 Safety-Related Control Projects Offline

Use the **Refine** command in the **Project Explorer** to open the project in the Control Participant. This lets you refine both the safety-related and the process program.

NOTE: Ensure that the associated M580 safety controller is configured first, page 912.

If a functional safety password, page 925 is set for the controller, you must disable the functional safety (safe area) password protection in the toolbar of the Participant window and enter the functional safety password before you can modify the safety-related program. The functional safety password protection is activated automatically when you close the Participant window.

The following figure shows an example of the Control Participant window that opens when you refine the safety-related Control project while functional safety password protection is enabled.



1 Button to disable/enable the functional safety password protection.

NOTE: When you refine a safety-related Control project, you can view and access the configuration of the associated controller. However, to modify the controller configuration, follow the system engineering life cycle and use the **Configure** command in the **Topology Explorer**.

NOTE: While the Control Participant refinement window is open for the M580 safety-related project, the associated safety controller is locked.

Impact of Refinement on the Associated M580 Safety Controller

When you refine a safety-related Control project and you have not yet configured the associated M580 safety controller, the safety controller changes to a configured state in the **Topology Explorer**. As a result, the controller properties that you configured in the **Topology Explorer** are applied to the configuration of the Control Participant.

Control Participant Project Settings

The *Usage of Process Namespace* project setting of the Control Participant is not supported and it must remain disabled. Otherwise, you cannot save changes that you made during the Control project refinement.

This has an impact on how the operator screen can reach variables of the Process and Global namespaces because you cannot use the *PROCESS* prefix anymore if two variables are declared with the same name in these namespaces.

For details, refer to the topic describing how to display data on operator screens (*Animation Tables and Operator Screens*) in *Modicon M580 Safety Manual*.

Performing Consistency Checks

When you perform a consistency check, page 460, only the logic of the process program is verified.

Device DDT Variable Wizard

In the toolbar of the Control Participant window, the button that opens the *device DDT variable wizard*, page 455 is not available because device DDT variables that are created for the hardware that is configured in the associated safety controller are already present and can be used.

Managing Network Variables

The **Manage Network Variables** dialog box, page 471 does not let you manage variables of the safety-related program.

M580 Safety Systems in the Configuration Stage

This topic describes the aspects of the configuration that are specific to using M580 safety systems. For general information, refer to the topic describing the configuration stage, page 562.

Creating and Configuring M580 Safety Controllers

Creating M580 Safety Controllers

To create an M580 safety controller, drag, page 565 an M580 safety controller entry from the **TOOLBOX** of the **Topology Explorer** to the **TOPOLOGY** node of the **SYSTEM PROJECT** pane or to a physical view.

Creating a safety controller also creates a blank M580 safety-related Control project in the **Project Explorer**, which is associated to the controller.

Conversely, if you create an M580 safety-related Control project, a default standalone safety controller is created in the topology.

Deleting M580 Safety Controllers

Deleting a safety controller also deletes the associated M580 safety-related Control project in the **Project Explorer**, and vice versa.

Configuring an M580 Safety Controller

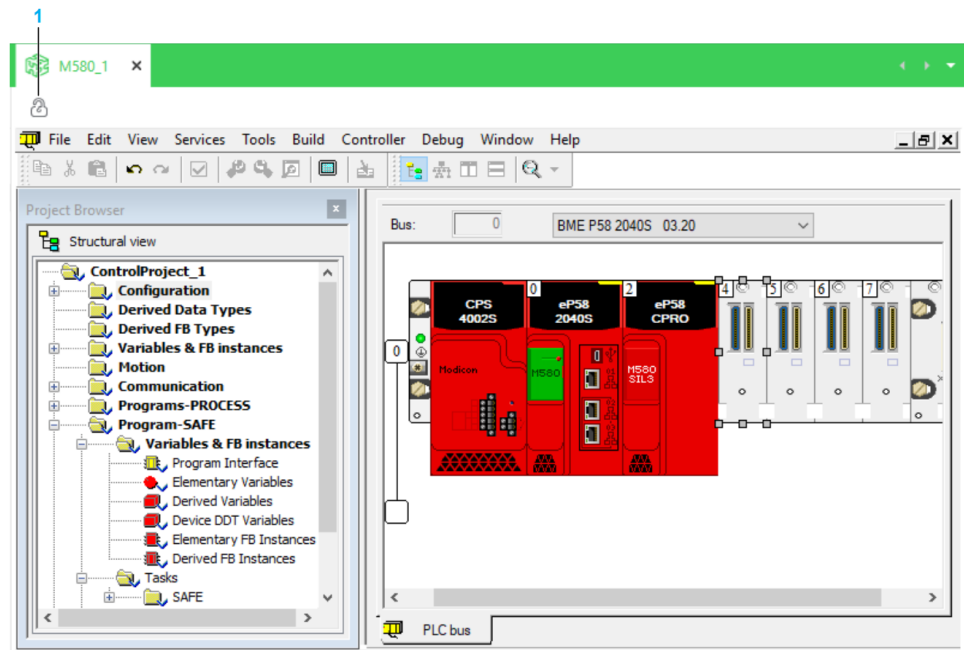
Use the **Configure** command, page 591 to open the Control Participant. This lets you add, change, and configure the hardware of the safety controller:

- The safety controller module and coprocessor module (including changing from a standalone to a redundant configuration)
- Safety power supply modules
- Safety I/O modules
- Other modules
- Racks

To configure an M580 safety controller that is created automatically when you create M580 safety-related Control project, you may need to set a **Web Diagnostic/Data Storage password**, page 588 and configure the properties of the controller module. Follow the indications that are displayed on the screen.

If the functional safety password, page 925 is enabled for the controller, you must disable the functional safety password protection in the toolbar of the Participant window and enter the functional safety password before you can configure safety modules. Functional safety password protection is enabled automatically when you close the Participant window.

The following figure shows an example of the Control Participant window that opens when you configure an M580 safety controller while functional safety password protection is enabled.



1 Button to disable/enable functional safety password protection.

NOTE: When you configure an M580 safety controller, you can view and access both the safety-related and the process programs of the associated safety-related Control project. However, to create or modify project data and logic, follow the system engineering lifecycle and use the **Project Explorer**.

NOTE: While the Control Participant configuration window is open for the M580 safety controller, the associated safety-related Control project is locked.

Analyzing Configuration Changes

By default, when you save the configuration of an M580 safety controller, the Control Participant does not perform an analysis. To analyze the configuration, enable *Analyze Project* in the settings, page 269 of the M580 safety-related Control project. This also analyzes the program areas of the Control project.

Managing the Functional Safety Password Protection Password

Overview

The functional safety password protection of the Control Participant for M580 safety controllers is managed from the **Topology Explorer** by using the **Safety** password function instead. The corresponding property in the Control Participant is disabled.

The **Safety** password is not available for a workstation running the controller simulator.

For more information about the functional safety password, refer to the topic describing *Application Security Management in Modicon M580, Safety Manual*.

NOTE: Ensure that the associated M580 safety controller is configured first, page 912.

Scope of the Functional Safety Password

The following table summarizes for which actions you must enter the functional safety password when the functional safety password protection is enabled.

Action	Purpose
Configuration of the M580 safety controller.	Disabling functional safety password protection to configure safety modules.
Offline refinement of the safety-relatedControl project of the M580 safety controller.	Disabling functional safety password protection to modify safety-related data and logic.
Online refinement of the safety-relatedControl project of the M580 safety controller.	Switching the controller to maintenance mode to modify safety-related data and logic online. This applies also to the controller simulator.
All deployment operations on the M580 safety controller that include the safety-related program.	Entering the Safety password is required to be able to proceed with the deployment only if the safety-related program has been modified. This applies also to the controller simulator. NOTE: A first deployment may require entering the Safety password even when the safety-related program has not been modified.

Functional Safety Password Requirements

Your password must contain from 8 to 16 alphanumeric characters to be accepted.

It can contain lower and upper case as well as special characters.

NOTE: Password strength is increased when the password contains a upper and lower case, alphabetical, numerical, and special characters.

Enabling and Disabling the Safety Password Property

The **Safety** password is enabled by default.

To disable it, use the **Safety** property, page 692 of the M580 safety controller.

Setting the Functional Safety Password

To set the functional safety password, use the **Manage Password** command, page 925 in the context menu of the M580 safety controller.

This action sets the build state of the executable of the associated safety-related Control project to **Out Of Date** if it was built.

Changing the Functional Safety Password

To change the functional safety password, use the **Manage Password** command, page 925 in the context menu of the M580 safety controller.

This action sets the build state of the executable of the associated safety-related Control project to **Out Of Date** if it was built.

Clearing the Functional Safety Password

To clear the functional safety password, use the **Clear Password** command, page 925 in the context menu of the M580 safety controller.

This action sets the build state of the executable of the associated safety-related Control project to **Out Of Date** if it was built.

Resetting a Forgotten Functional Safety Password

If you also forgot the **Controller** password, reset, page 804 it before resetting the functional safety password.

To reset the functional safety password of an M580 safety controller, proceed as follows.

Step	Action
1	Right-click the topological entity and select Forgot Password > Safety . Result: The Forgot Password dialog box opens and an authentication code is displayed.
2	Contact your local Schneider Electric service representative, provide the authentication code, and specify that you want to reset the functional safety password. Result: You receive a temporary password. NOTE: The temporary password is valid as long as you do not modify the configuration or the associated safety-related Control project after requesting it.
3	Right-click the safety controller and select Manage Password > Safety . Result: The Manage Password dialog box opens.
4	Enter the temporary password in the Current Password text box, the new password, and click OK . Result: The new functional safety password is set.
5	Build the executable of the associated safety-related Control project if its build state is Out Of Date .
6	If you already deployed the project to the safety controller, apply the new functional safety password to the deployed safety-related Control project by using the Deploy Changes/Undo Online Changes command.

Exporting and Importing M580 Safety Controllers

This topic describes the aspects of the topology export and import that are specific to M580 safety controllers. For general information about topology export and import operations, refer to the topic describing how to manage the topology, page 622 of a system.

Exporting M580 Safety Controllers

The **Export > Topology** command of the **Topology Explorer** lets you export one or more M580 safety controllers and create a non-editable export file (.sbk).

You can also export at the same time other topological entities of the system.

The associated safety-related Control project is not exported.

Importing M580 Safety Controllers

The **Import > Topology** command of the **Topology Explorer** lets you import the M580 safety controllers that are contained in a topology export file (.sbk).

The Global Templates Library of the target system must contain the necessary topological safety-related templates (Foundation Library (see *EcoStruxure Process Expert, Installation and Configuration Guide*)) before importing. These templates may be missing if you migrated a database of an earlier version. In this case, you cannot save your changes when you configure the safety controller.

The following table describes the result when you import an M580 safety controller in different scenarios.

Target system	Action selected in the Import window to solve the conflict	Result after import in Topology Explorer	Impact on projects after import
An M580 safety controller with the same identifier does not exist.	N/A	The safety controller is created with the same identifier and configuration. The exported project settings are also retained.	A blank M580 safety-related Control project is created, which is associated to the safety controller.
An M580 (not safety-related) controller with the same identifier already exists.	Create NOTE: Update is not available.	The safety controller is created with a new identifier but the same configuration. The exported project settings are also retained.	A blank M580 safety-related Control project is created, which is associated to the safety controller.
An M580 safety controller with the same identifier already exists.	Create	The safety controller is created with a new identifier but the same configuration. The exported project settings are also retained.	A blank M580 safety-related Control project is created, which is associated to the safety controller.
	Update	The configuration of the existing safety controller is replaced with that of the imported safety controller. The project settings are also replaced.	No impact.

NOTE: During import of a safety controller, the associated safety-related M580 Control project is locked.

Functional Safety Password Management When Importing

When you import one or more M580 safety controllers, the **Verify Password** window may open prompting you to enter a functional safety password for each one.

The import operation starts only after all required passwords were entered.

The following table describes which functional safety password you must enter when prompted during the import of an M580 safety controller.

Target system	Functional safety password required	Impact on functional safety password after import
An M580 safety controller with the same identifier does not exist.	The functional safety password of the safety controller that you are importing (if set).	The functional safety password settings are the same as that of the imported safety controller.
An M580 safety controller with the same identifier already exists.	The functional safety password of the existing safety controller in the target system (if set).	<p>The functional safety password and password settings are the same as that of the existing safety controller in the target system.</p> <p>For example, if the existing safety controller has no functional safety password set but the safety controller that you are importing does, after the import, no functional safety password is set.</p>

M580 Safety Systems in the Mapping Stage

This topic describes the aspects of the Control project mapping that are specific to using M580 safety systems. For general information, refer to the topic describing the mapping stage.

Control Executable Service Mapping

You can create a service mapping only with the M580 safety controller that is associated to the safety-related Control project.

Only one executable of a safety-related Control project can be mapped to this controller. If you have created additional executables, you can only map them to workstations on which a Control Expert service exists.

Communication Mapping

The table describes the type of peer to peer communication that you can implement by using the **Communication Mapping Editor**.

Controller 1	Controller 2	Peer to peer communication
M580 safety controller	M580, M340, or Quantum platform	Possible by using process variables only.
M580 safety controller	M580 safety controller	Possible by using process variables only. To use variables of the safety-related program, refine the safety-related Control projects.

In addition, communication between an M580 safety controller module and CIP safety devices cannot be implemented by using the **Communication Mapping Editor**. You must refine the safety-related Control project.

Hardware Mapping

The I/O safety modules of the M580 safety controller do not appear in the **Hardware Mapping Editor** and can, therefore, not be mapped to interfaces of Control facets.

Mapping of communication interfaces of the controller module and other not safety-related communication modules of the safety controller can be performed as you would do with controllers of the other platforms.

M580 Safety Systems in the Build Stage

This topic describes the aspects of building executables of Control projects in the **Project Explorer** that are specific to M580 safety systems. For general information, refer to the topic describing the build stage.

Building M580 Safety-Related Control Project Executables

Using a build command in the context menu of the executable of a safety-related Control project locks the associated M580 safety controller.

NOTE: Ensure that the associated M580 safety controller is configured first, page 912.

NOTE: After you build the executable of a safety-related Control project by using the **Build All** command, you cannot modify the service mapping anymore nor delete the executable if it is mapped to the M580 safety controller.

Safe Signature

If you changed the safety-related configuration, safety-related program, or certain project settings of an M580 safety-related Control project, each time you build its executable, a *SourceSafeSignature* is generated.

For a list of actions that generate a *SourceSafeSignature* change, refer to *Modicon M580 Safety Manual*.

The *SourceSafeSignature* is used to inform you of a change when you use a deployment command.

However, the Safe Signature cannot be viewed at the platform level. To copy it, open the safety-related Control project by using the **Open Built Project** command in the context menu of its Control executable.

NOTE: The **Safety Signature management** setting of the Control Participant is set to **Automatic** when you build an M580 safety-related Control project (**Tools > Project Settings > General > Build Settings**).

M580 Safety Systems in the Deployment Stage

This topic describes the aspects of the Control project deployment that are specific to using M580 safety systems. For general information, refer to the topic describing the deployment stage.

Deploying to M580 Safety Controllers

Use the **Deploy Built Project** and **Re-Deploy Last Project** commands in the **Topology Explorer** to deploy the safety-related Control project to the M580 safety controller. This lets you deploy both the safety-related and the process program.

Deploying Changes to M580 Safety Controllers

Use the **Deploy Changes / Undo Online Changes** command in the **Topology Explorer** to [deploy changes, page 830](#) to the safety-related and/or the process program of the M580 safety controller.

The changes that appear in the **Safe Program** section can only be deployed all at once or not at all.

When you deploy changes to the process program that impact the safety-related program (for example, a process variable that is used as program interface to exchange data with the safety-related program), a dialog box opens, which allows you to confirm the change or modify the selection.

Safe Signature Changes

When you deploy, re-deploy, or deploy changes that include the safety-related program, if EcoStruxure Process Expert detects that the *SourceSafeSignature* of the program you are deploying is different from that of the program in the controller, a dialog box opens to inform you.

You can acknowledge the message and proceed with the deployment or cancel the operation.

The *SourceSafeSignature* is not displayed and cannot be copied.

Deploying With Functional Safety Password Protection Enabled

If the functional safety password, [page 925](#) is enabled for the controller and the safety-related program was modified, you must enter the functional safety password when prompted before you can deploy, re-deploy, or deploy changes to the safety-related program.

M580 Systems in the Execution Stage

This topic describes the aspects of the project online refinement and update that are specific to using M580 safety systems. For general information, refer to the topic describing the execution stage.

Refining M580 Safety-Related Control Projects Online

Use the **Refine Online** command in the **Topology Explorer** to make online modifications to the process program and the associated configuration, and the safety-related program and the associated configuration of the deployed Control project.

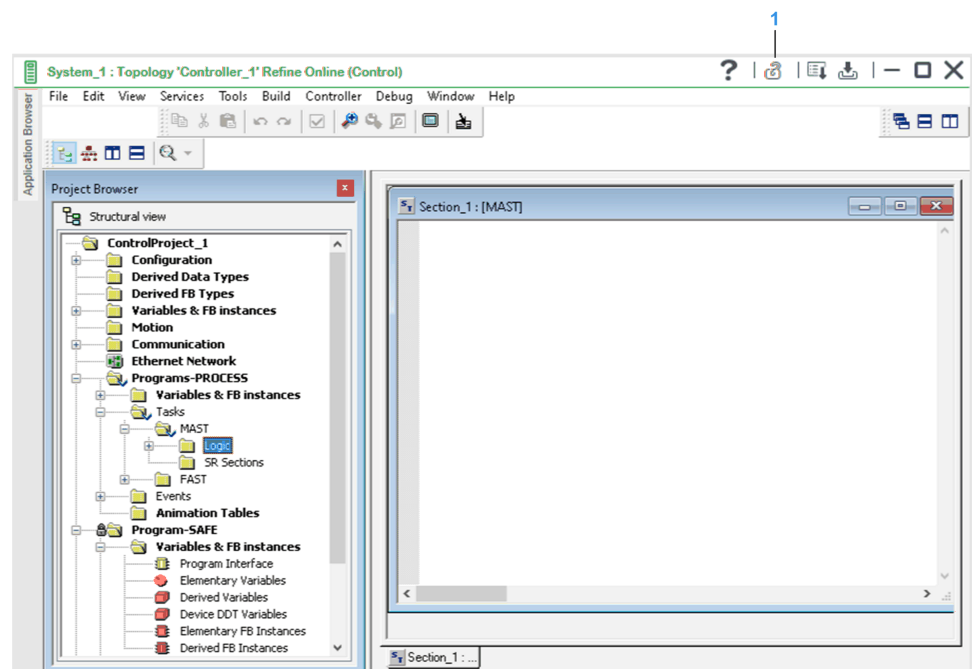
The controller opens in safety mode.

If a functional safety password, page 925 is set for the controller, you must switch to maintenance mode by using the button in the toolbar of the Participant window before you can modify the safety-related program or configuration online. This requires entering the functional safety password. If the functional safety password is disabled, switching to maintenance mode does not require a password.

Safety mode is enabled automatically when you close the Participant window.

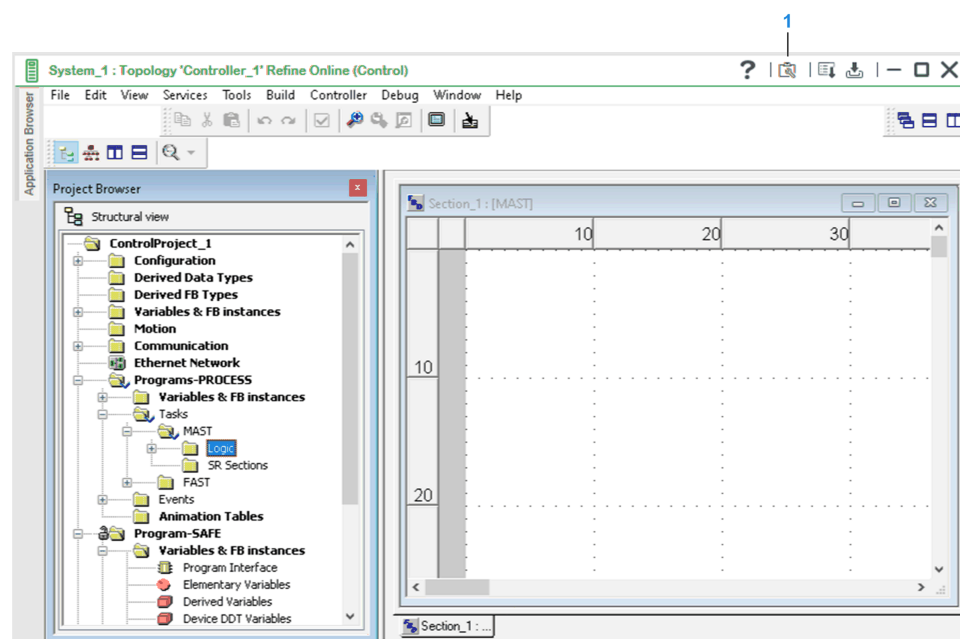
The same applies also when you refine online by using the controller simulator.

The following figure shows an example of the Control Participant window that opens when you refine a safety-related Control project online while functional safety password protection is enabled.



1 Button to switch between safety mode and maintenance mode.

The following figure shows an example of the Control Participant window that opens when functional safety password protection is disabled.



1 Button to switch between safety mode and maintenance mode without entering the password.

Updating Logical M580 Safety-Related Control Projects

Use the **Update Project** command in the **Topology Explorer** to update the logical M580 safety Control project and the configuration with changes that you have made online. This lets you update, page 897 both the process and the safety areas.

However, the changes that appear in the **Safe Program** section can only be updated all at once or not at all.

If the functional safety password protection, page 925 is enabled for the controller, before you can update the safety program, you need to enter the safety password.

Using Supervision with M580 Safety Systems

Overview

You need to create your own Supervision instances to monitor safety-related Control data because instances of the application cannot be used to create data and logic of the safety-related program of M580 safety-related Control projects.

For general information on working with Supervision instances, refer to the various Supervision topics in the description of the system engineering life cycle.

Using Supervision Templates

To use Supervision facets, proceed as follows.

Step	Description
1	<p>You can instantiate the following Supervision data facet templates of the Foundation Library for each variable of an M580 safety-related Control project that you want to monitor from the Supervision runtime.</p> <ul style="list-style-type: none"> • <i>\$VariableTag_CD</i> • <i>\$AdvancedAlarm_CD</i> • <i>\$DigitalAlarm_CD</i> • <i>\$TrendTag_CD</i> <p>NOTE: Select Facets as filter in the Browser pane.</p>
2	<p>Configure instances of <i>\$VariableTag_CD</i> as follows:</p> <ul style="list-style-type: none"> • <i>Addr</i> parameter (<i>Configuration</i> category): Name of the safety-related variable in the Control project preceded by the <i>safe.</i> prefix. For example, if the variable name is <i>MyVar1</i>, enter <i>safe.MyVar1</i>. • <i>Type</i> parameter (<i>Configuration</i> category): Data type of the variable. For example, <i>UINT</i>.
3	<p>If you use trend tags, open the Asset Workspace Editor and link the trend tag instance to the variable tag instance so that the instance status becomes valid.</p>
4	<p>Follow the system engineering life cycle to complete the other stages for the Supervision project.</p>

Managing Communication

What's in This Chapter

Communication Through Modbus TCP Ethernet Implicit Messaging (I/O Scanning)	936
Communication by Using Modbus TCP Ethernet Explicit Messaging	944
Communication by Using EtherNet/IP Explicit Messaging	949
Communication Through Modbus Explicit Messaging	952
Monitoring Modbus Data From Supervision	984
Communication by Using a PROFIBUS Primary Module	988

Overview

This chapter contains information describing how to manage communication between devices and controllers depending on the communication protocol that you want to use. It contains examples that use templates of the General Purpose Library Classic.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

NOTE: EcoStruxure Process Expert does not verify the status of the communication between controllers and I/O devices.

Communication Through Modbus TCP Ethernet Implicit Messaging (I/O Scanning)

Overview

This section contains specific procedures to help you implement communication through Modbus TCP Ethernet implicit messaging by using the I/O scanner function. It applies to communication between a controller and devices of the **EPE Managed Devices** product range that are configured in the topology of the system, and between several controllers (peer to peer communication). It contains examples that use templates of the General Purpose Library Classic.

NOTE: For the engineering of a system, it is not relevant how Modbus TCP devices are physically connected to the logical network. As long as they are connected to the same logical logical network with a valid IP address in EcoStruxure Process Expert, you can proceed with the implementation of communication. Thereafter, in your physical architecture, you must ensure that connections are made properly so that devices can communicate.

For example, in an M580 architecture, whether an STB island is connected directly to the service port of the controller module, to the port of a BMENOC0301.2 module, or to the service port of a BMXCRA31210 module does not matter. In EcoStruxure Process Expert, you configure only the DIO scanner service of the controller or BMENOC•••• module to exchange data with the STB island on the same logical network.

NOTE: To configure communication for devices of the other product ranges of the **Automation Devices** product type, configure their properties in the **PROPERTIES** pane and use the commands of their context menu.

Communicating with Devices Through Modbus TCP Implicit Messaging (I/O Scanning)

Overview

You can implement communication through Modbus TCP Ethernet implicit messaging between devices of the **EPE Managed Devices** product range and supported controller platforms by using the I/O scanner function.

In addition to configuring the controller as a client and enabling its I/O scanner, you must instantiate the topological template of the Modbus TCP device.

The topological device instance allows you to assign an IP address to the device and connect it to the logical network to which the controller is connected.

The communication channel between the controller and the device is managed through the [communication mapping](#), page 725.

The build process creates the communication channel in the built Control Participant project.

For a list of available device templates, refer to the Device/Communication Matrix (see *EcoStruxure Process Expert*, *General Purpose Library Classic*, *Communication Control Services*, *Reference Manual*).

NOTE: Refer to the TVDA document *Hot Standby Quantum_EIO* for guidelines when using a 140NOC•••• communication module to implement communication between Modbus TCP devices and Hot Standby controllers of the Quantum platform.

Using Device Application Templates

The device-specific application template of a Modbus TCP device (for example, *\$TsysTE*) allows you to edit the Control and Supervision parameters of the device and proceed with the hardware mapping to the controller I/O modules.

It is not required for managing communication with the device using I/O scanning.

Generic Device Templates

If no specific topological template is available for the device that you are using, use one of the [generic topological device templates](#), page 603 and its application counterpart.

The generic application template lets you create an input and an output variable of required length and data type.

The table indicates which device template to use depending on the module that is used to communication with the device.

Module used to communicate with device	Topological device template to use	Corresponding application device template to use
M580 controller module	<i>\$EGenericDeviceHW</i>	<i>\$GenNOCDevice</i>
BMENOC•••• communication module		Some manual configuration is required (see <i>EcoStruxure™ Process Expert</i> , <i>General Purpose Library Classic Device Templates Reference Manual</i>).
Quantum controller module	<i>\$EGenericDeviceHW</i>	<i>\$GenericDevice</i>
140NOE••••• communication module		

Implementing Communication

To implement communication between a controller and a device through Modbus TCP by using I/O scanning, complete these steps.

Step	Description	Stage
1	Create a Control project in the system.	Project Definition
2	Create a Control executable in the project.	
3	Create and configure a controller, page 591 with the I/O scanner enabled and its communication module. Configure the client memory start and length parameters, page 595 of the communication module.	Configuration
4	Create and configure, page 603 the topological device entity (configure IP address and other applicable properties). NOTE: If the device communicates with a communication module that validates IP addresses based on the subnet mask, verify that the IP addresses of the device and the module belong to the same subnet.	
5	Create a logical network, page 576.	
6	Connect the communication module of the controller to the logical network.	
7	Connect the I/O device, page 679 to the same logical network.	Instantiation
8	Instantiate the application template that corresponds to the topological device entity and configure its properties.	
9	Map the Control executable of the project to the controller entity, page 723.	Mapping (service)
10	Define the communication channel, page 725 between the controller and the device.	Mapping (communication)

To complete the other steps to implement the system that contains the controller and the device, follow the system engineering life cycle described in this document.

NOTE: Refer to the topic describing device communication technologies (see *EcoStruxure Process Expert*, *General Purpose Library Classic*, *Device Control Services*, *Reference Manual*) for configuration and setup details.

Implementing Communication by Using Generic Device Templates

To implement communication through Modbus TCP by using I/O scanning, between a controller and a device for which no specific topological template exists, complete these steps.

Step	Description	Stage
1	Create a Control project in the system.	Project Definition
2	Create a Control executable in the project.	
3	Create and configure a controller, page 591 with the I/O scanner enabled and its communication module. Configure the client memory start and length parameters, page 595 of the communication module.	Configuration
4	Create and configure the generic topological device entity, page 603, assign an IP address to it, and enter values for the read/write parameters, page 604. NOTE: If the device communicates with a communication module that validates IP addresses based on the subnet mask, verify that the IP addresses of the device and the module belong to the same subnet.	
5	Create a logical network, page 576.	
6	Connect the communication module of the controller to the logical network.	
7	Connect the I/O device, page 679 to the same logical network.	

Step	Description	Stage
8	Instantiate the generic application device, page 937 template and configure its properties.	Instantiation
9	Map the Control executable of the project to the controller entity, page 723.	Mapping (service)
10	Define the communication channel, page 725 between the controller and the device.	Mapping (communication)

To complete the other steps to implement the system that contains the controller and the device, follow the system engineering life cycle described in this document.

Communicating with STB Islands Through Modbus TCP Implicit Messaging

Overview

You can implement communication through Modbus TCP Ethernet implicit messaging between STB islands of the **EPE Managed Devices** product range and supported controller platforms by using the I/O scanner function.

The controller acts as client, and its I/O scanner is enabled.

The STB island acts as server, and both are connected to the same logical network, on which they are identified with their respective IP address.

The communication channel between the controller and the STB island is defined through the [communication mapping](#), [page 725](#).

The build process creates the communication channel in the built Control Participant project.

Implementing Communication

To implement communication between a controller and an STB island through Modbus TCP (I/O scanning), complete these steps.

Step	Description	Stage
1	Create a Control project in the system.	Project Definition
2	Create a Control executable in the project.	
3	Create and configure a controller, page 591 with the I/O scanner enabled and its communication module. Configure the client memory start and length parameters, page 595 of the communication module.	Configuration
4	Create and configure an STB island, page 599 entity and assign an IP address to its NIM. NOTE: If the NIM communicates with a communication module that validates IP addresses based on the subnet mask, verify that the IP addresses of the NIM and the module belong to the same subnet.	
5	Create a logical network, page 576 .	
6	Connect the communication module of the controller, page 617 to the logical network.	
7	Connect the communication module of the STB island, page 679 to the same logical network.	
8	Map the Control executable of the project to the controller entity, page 723 .	Mapping (service)
9	Define the communication channel, page 725 between the controller and the STB island.	Mapping (communication)

To complete the other steps to implement the system that contains the controller and the STB island, follow the system engineering life cycle described in this document.

Communicating with PROFIBUS Remote Masters Through Modbus TCP Implicit Messaging

Overview

You can implement communication through Modbus TCP implicit messaging between PROFIBUS remote masters (PRMs) of the **EPE Managed Devices** product range and supported controller platforms by using the I/O scanner function.

The topological PRM entity allows you to configure decentralized devices (DPs) and assign an IP address to the PRM to connect it to the logical network to which the controller is connected.

In addition to configuring the controller as a client with the I/O scanner enabled, you must instantiate:

- A PRM management application template.
- A device application template that corresponds to the DP that is configured in the PRM entity.

The communication channel between the controller and the PRM is managed through the [communication mapping](#), page 725.

The build process creates the communication channel in the built Control Participant project.

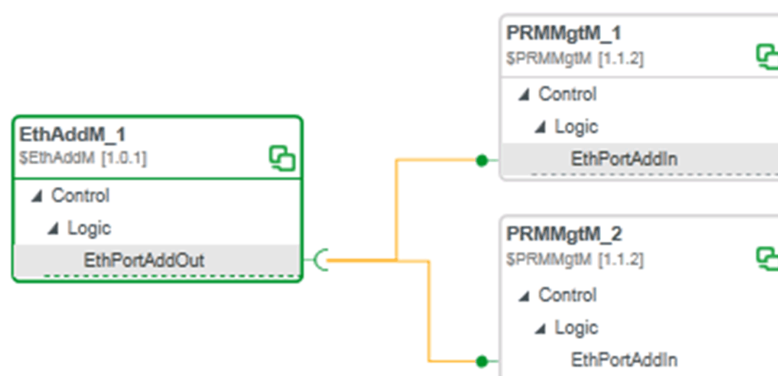
NOTE: The device application templates of the configured DPs allow you to edit the Control and Supervision parameters of the devices and proceed with the hardware mapping to the controller I/O modules. They are not required to manage communication with the DPs by using I/O scanning.

Using several Communication Port Instances

To use one topological communication port with several PRM management instances (application), you must instantiate the following intermediate templates (fan-out type application templates). Each instance of such template accepts links to n PRM management instances.

PRM management template	Intermediate template
<i>\$PRMMgtM</i>	<i>\$EthAddM</i>
<i>\$PRMMgtQ</i>	<i>\$EthAddQ</i>

The following figure shows an example of how to link two instances of the PRM management template to the instance of an intermediate template. During hardware mapping, the interface of the intermediate template instance is mapped to the interface of the Ethernet port of an M340 or M580 controller.



Implementing Communication

To implement communication between a controller, a PRM and its DP through Modbus TCP by using I/O scanning, complete these steps.

Step	Description	Stage
1	Create and configure a controller, page 591 with the I/O scanner enabled and its communication module. Configure the client memory start and length parameters, page 595 of the communication module.	Configuration
2	Create and configure, page 601 the PRM entity and its DPs (configure IP address and other applicable properties). NOTE: If the PRM communicates with a communication module that validates IP addresses based on the subnet mask, verify that the IP addresses of the PRM and the module belong to the same subnet.	
3	Create a logical network, page 576.	
4	Connect the communication module of the controller, page 615 to the logical network.	
5	Connect the PRM entity, page 618 to the same logical network.	
6	In the Application Explorer , instantiate the following templates: <ul style="list-style-type: none"> The PRM management template. One for each PRM entity that is configured: <ul style="list-style-type: none"> <code>\$PRMMgtQ</code> for a controller of the Quantum platform. <code>\$PRMMgtM</code> for a controller of the M340 or M580 platform. One application template for each DP that you have configured (for example, <code>\$ATV71PB</code> for an Altivar 71 DP). NOTE: If no specific template is available for a given DP, you can instantiate the generic application template <code>\$GenericDevice</code> .	Instantiation
7	If you are using the <code>\$PRMMgtQ</code> PRM management template, in the Instance Editor , configure the StartMemoryHeap parameter. Refer to the EcoStruxure Process Expert help on the template for details.	
8	If you are using several PRM management instances, instantiate the corresponding intermediate template, page 941 and link its instance to each PRM management instance by using the Asset Workspace Editor . NOTE: A link between the DP instance and the PRM management instance is not required.	
9	Create a Control Participant project in the system.	Project Definition
10	Assign the necessary facets to the Control Participant project.	
11	Create a Control executable in the project.	
12	Map the Control executable of the project to the controller entity, page 723.	Mapping (service)
13	Define the communication channel, page 725 between the controller and the device.	Mapping (communication)
14	Map to their respective topological counterpart (HO) by dragging from right to left: <ul style="list-style-type: none"> The EMPortChM or EMPortChQ interface. The PRMCTRLData interface. The PRMEthernetAddress interface. The interface of the device application facet to the interface of the topological DP device. 	Mapping (hardware)

To complete the other steps to engineer the system that contains the controller and the device, follow the system engineering life cycle described in this document.

NOTE: Refer to the EcoStruxure Process Expert help on the various templates for configuration and setup details.

Peer to Peer Communication Through Modbus TCP Ethernet Implicit Messaging

Overview

Peer to peer communication allows you to share data between two or more Control Participant projects by using the I/O scanner function for any pair of controllers that are configured in the topology of the system.

Network variables are used to exchange data between the Participant projects to which the controllers are mapped.

Client/server communication channels and network variable mappings are managed through [communication mapping](#), [page 725](#).

The build process creates the communication channels and allocates the mapped variables in the respective built Control Participant projects.

For more information about ways to create variables to share data, refer to the topic documenting the [creation of network variables](#), [page 463](#).

Implementing Peer to Peer Communication

To implement peer to peer communication through Modbus TCP implicit messaging by using I/O scanning, follow these steps.

Step	Description	Stage
1	Create two Control projects in the same system.	Project definition
2	Create a Control executable in each project.	
3	Create the network variables that will be exchanged, page 463 between the projects.	Refinement
4	Create and configure two controllers, page 591 with the necessary communication and I/O modules. In the controller acting as client, enable the I/O scanner function.	Configuration
5	In the controller acting as client, configure the client memory start and length parameters, page 595 of the communication module.	
6	In the controller acting as server, configure the server memory start and length parameters, page 595 of the communication module. NOTE: If the server communicates with a communication module that validates IP addresses based on the subnet mask client-side, verify that the IP addresses of the client/server modules belong to the same subnet.	
7	Create an logical network, page 576 .	
8	Connect the communication module, page 617 of each controller to this logical network.	Mapping (service)
9	Map the Control executable of the owner project to the controller, page 723 acting as server.	
10	Map the Control executable of the consumer project to the controller, page 723 acting as client.	Mapping (communication)
11	Define the peer to peer communication channels, page 725 between the controllers.	
12	Map the network variables, page 738 to the peer to peer communication channels.	

To complete the other steps to implement the system, follow the system engineering life cycle described in this document.

Communication by Using Modbus TCP Ethernet Explicit Messaging

Communication with Devices by Using Modbus TCP Ethernet Explicit Messaging

Overview

This topic contains examples that use templates of the General Purpose Library Classic.

Supported Architectures

You can implement communication by using Modbus TCP Ethernet explicit messaging between devices of the **EPE Managed Devices** product range and supported controller platforms.

The process requires to instantiate the topological template of the Modbus TCP device, its application counterpart, and the appropriate application template of the communication port.

The communication port instance lets you configure the number of simultaneous requests that are sent per cycle. The configuration depends on the capacity of the communication module. For example, the various M580 controller modules can send from 16 to 80 requests as client.

The topological device instance lets you assign an IP address to the device and connect it to the logical network to which the controller is connected.

The IP address is exchanged with the application template of the device through the hardware mapping.

For a list of supported communication architectures, refer to the Device/Communication Matrix (see *EcoStruxure Process Expert, General Purpose Library Classic, Communication Control Services, Reference Manual*).

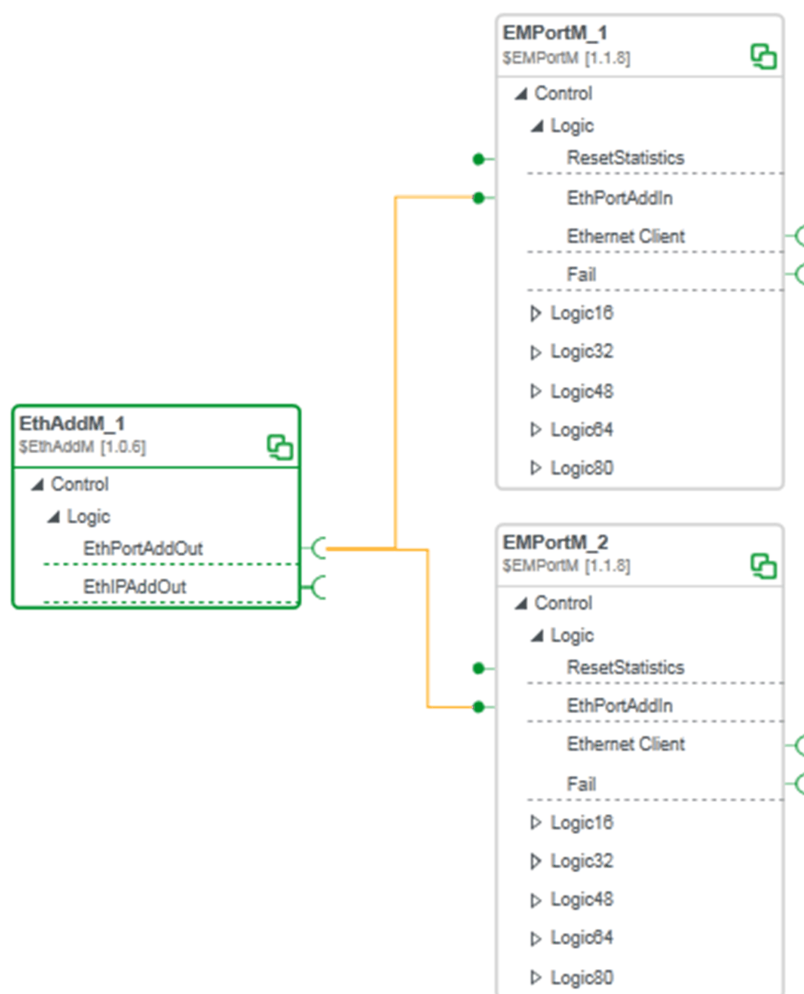
NOTE: If no device-specific topological template is available for the device that you are using, use a generic topological device template, page 603 and its application counterpart *\$EMClient*.

Using several Communication Port Instances

To use one topological communication port with several communication port instances (application), you must instantiate the following intermediate templates (fan-out type application templates). Each instance of such template accepts links to n communication port instances.

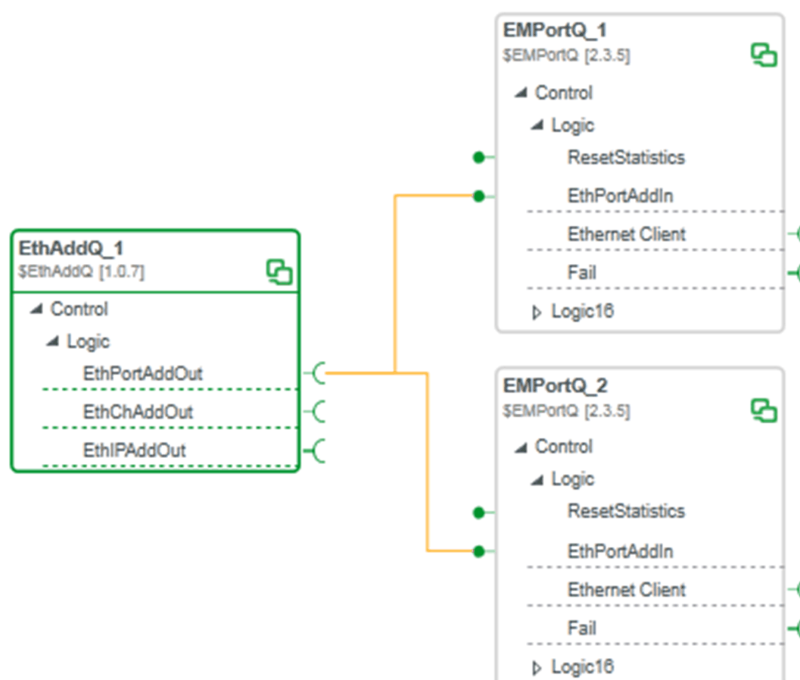
Communication port template	Intermediate template
<i>\$EMPortM</i>	<i>\$EthAddM</i>
<i>\$EMPortQ</i>	<i>\$EthAddQ</i>

The following figure shows an example of how to link two instances of the communication port template to the instance of an intermediate template. In the hardware mapping, the interface of the intermediate template instance is mapped to the interface of the Ethernet port of an M340 or M580 controller.



NOTE: The **EthIPAddOut** interface is not used for Modbus TCP explicit messaging.

The following figure shows an example of how to link two instances of the communication port template to the instance of an intermediate template. In the hardware mapping, the interface of the intermediate template instance is mapped to the interface of the Ethernet port of a Quantum controller.



NOTE: The **EthChAddOut** and **EthIPAddOut** interfaces are not used for Modbus TCP explicit messaging.

Implementing Communication

To implement communication between a controller and a device by using Modbus TCP Ethernet explicit messaging, follow these steps.

Step	Description	Stage
1	Instantiate the process template that corresponds to the piece of equipment that you want the device to communicate with (for example, \$Motor).	Instantiation
2	Instantiate the Modbus TCP explicit messaging (EM) application template that corresponds to device that you want to use (for example, \$TesySTEM). NOTE: If no device-specific application template exists, use the \$EMClient template. Create as many instances of it as you have requests. For example, for one read and one write request you need two instances.	
3	Instantiate the following communication port template: <ul style="list-style-type: none"> \$EMPortQ for a controller of the Quantum platform. \$EMPortM for a controller of the M340 or M580 platform. 	
4	Configure the communication port instance: <ul style="list-style-type: none"> The Control element lets you select a profile, which corresponds to the maximum number of simultaneous requests that the communication module supports. The corresponding LogicX service lets you optimize communication by reducing the number of simultaneous requests that are sent depending on your needs. X corresponds to the last two digits of the profile you select under Control. For example, to EMPortM16 corresponds Loigc16. NOTE: If you select a profile that exceeds the capacity of the communication module, you cannot assign the communication port instance to the Control Participant project. Even if you reduce the number of requests under LogicX . For example, if you select EMPortM32 for a BMENOC communication module, which can send a maximum of 16 requests simultaneously.	

Step	Description	Stage
5	If you are using one topological communication port with several communication port instances, instantiate the corresponding intermediate template, page 944.	
6	Open the Asset Workspace Editor from the device instance and link to it: <ul style="list-style-type: none"> The port instance The process instance NOTE: In case of a generic device, the device instance is the <i>\$EMClient</i> template instance. NOTE: If you use an intermediate template instance, link it to the port instance.	
7	Configure a controller and its communication module.	
8	Create, page 603 the corresponding topological device entity and configure its IP address and other applicable properties (for example, <i>\$ETesysTHW</i>). NOTE: If no device-specific topological template exists, use a generic device template.	Configuration
9	Create a logical network.	
10	Connect the communication module of the controller and the topological device entity to the same logical network.	
11	Assign the Control facets of the process, device, and communication port instances to the Control Participant project so that the execution order, page 394 is as follows: <ol style="list-style-type: none"> The facet of the Modbus TCP explicit messaging device instance (for example, <i>TesysTEM_1</i>). The facet of the communication port instance (for example, <i>EMPortQ_1</i>). 	Assignment
12	Generate the project.	Generation
13	Map the executable of the Control Participant project to the engine of the controller, page 723.	Mapping (service)
14	Map the EMPortChM or EMPortChQ interface of the communication port instance (or of the intermediate template instance) to its topological counterpart.	Mapping (hardware)
15	Map the interface of the application instance of the device to the interface of the topological device instance.	

To complete the other steps to engineer the system that contains the controller and the device, follow the system engineering life cycle described in this document.

NOTE: Refer to the following topics for configuration and setup details of:

- Devices (see *EcoStruxure Process Expert, General Purpose Library Classic, Device Control Services, Reference Manual*)
- Ports (see *EcoStruxure Process Expert, General Purpose Library Classic, Communication Control Services, Reference Manual*)

Examples

The following examples illustrate the configuration of Modbus TCP explicit messaging to connect to 20 devices from clients of two different platforms with different communication capacities.

Example 1	
Number of devices	20
Controller platform	Modicon Quantum
Communication module	140NOC77101
Communication port template	<i>\$EMPortQ</i>
Communication port instance configuration (<i>PortSelection</i>)	<i>EMPortQ16</i>

Example 1

Requests sent simultaneously (<i>SimultaneousSends</i>)	10 in <i>Logic16</i>
Cycles to connect once to the number of devices	2 (2 x 10)

Example 2

Number of devices	20
Controller platform	Modicon M580
Communication module	BMEP582020
Communication port template	<i>\$EMPortM</i>
Communication port instance configuration (<i>PortSelection</i>)	<i>EMPortM32</i>
Requests sent simultaneously (<i>SimultaneousSends</i>)	20 in <i>Logic32</i>
Cycles to connect once to the number of devices	1

Managing IP Address Changes

The topological templates of devices communicating by using Modbus TCP Ethernet explicit messaging use a variable to define the IP address of the device.

During deployment, the templates set up the initial value of this variable, which is connected to the pin of the DFB implementing the device.

When you change the IP address of a device by changing the hardware mapping of the facet modeling the device and deploy the change by using the **Deploy Changes** command, only the initial value, in this case the IP address, is changed online. The existing value remains the same.

The IP address change becomes effective, either through a cold start of the controller or by replacing the existing value of the variable in the animation table by the new initial value.

Communication by Using EtherNet/IP Explicit Messaging

Communication with Devices by Using EtherNet/IP Explicit Messaging

Overview

This section contains a procedure specifically aimed at managing communication between devices of the **EPE Managed Devices** product range and M580 controllers by using EtherNet/IP explicit messaging. It contains examples that use templates of the General Purpose Library Classic.

Supported Architectures

You can implement communication by using EtherNet/IP explicit messaging between devices and controller modules of M580 controllers or BMENOC... communication modules.

The process requires to instantiate the topological template of the EtherNet/IP device, its application counterpart, and the appropriate application template of the communication port.

The communication port instance lets you configure the number of simultaneous requests that are sent per cycle. The configuration depends on the capacity of the communication module. For example, the various M580 controller modules can send from 16 to 80 requests as client.

The topological device instance lets you assign an IP address to the device and connect it to the logical network to which the controller is connected.

The IP address is exchanged with the application template of the device through the hardware mapping.

For a list of supported communication architectures, refer to the EcoStruxure Process Expert help.

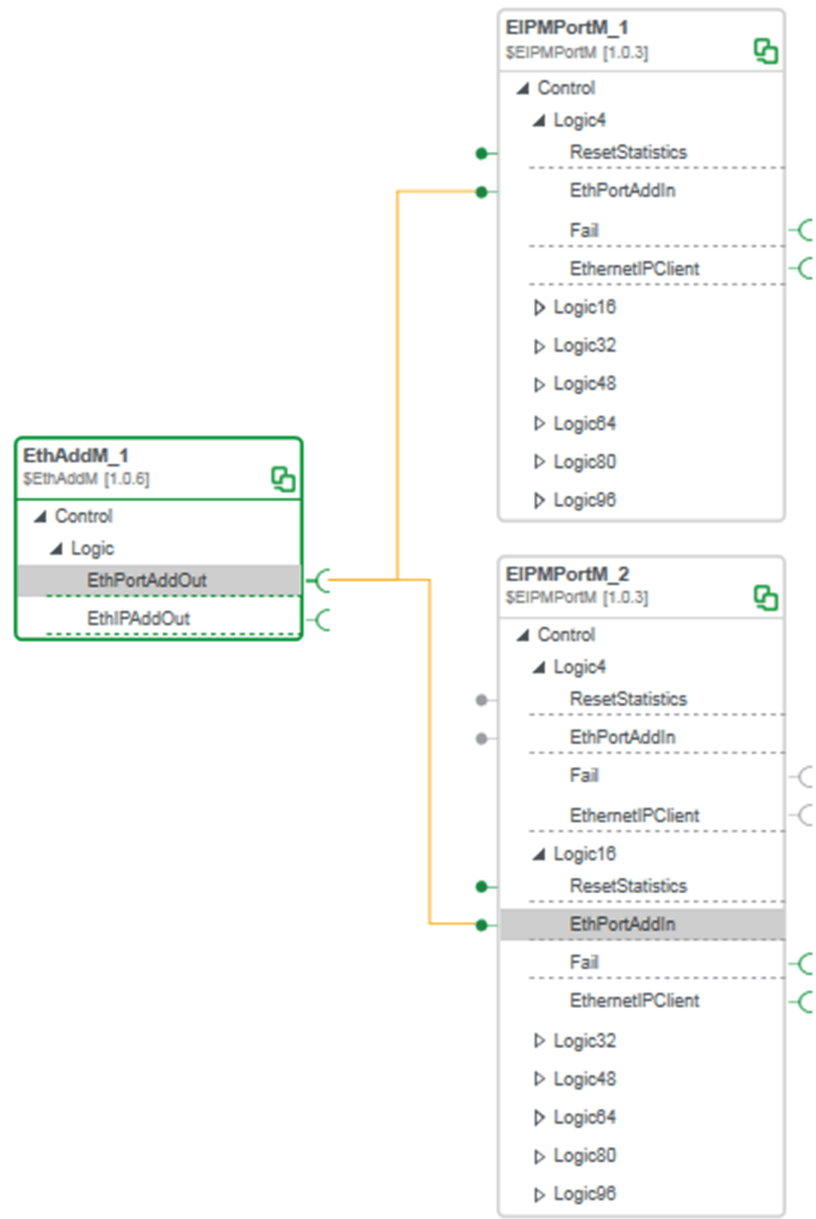
NOTE: If no device-specific topological template is available for the device that you are using, use a generic topological device template, page 603 and its application counterpart *\$EIPMClient*.

Using several Communication Port Instances

To use one topological communication port with several communication port instances (application), you must instantiate the following intermediate templates (fan-out type application templates). Each instance of such template accepts links to n communication port instances.

Communication port template	Intermediate template
<i>\$EIPMPortM</i>	<i>\$EthAddM</i>

The following figure shows an example of how to link two instances of the communication port template to the instance of an intermediate template. In the hardware mapping, the interface of the intermediate template instance is mapped to the interface of the Ethernet port of an M580 controller.



NOTE: The **EthIPAddOut** interface is not used.

Implementing Communication

To implement communication between a controller and a device by using EtherNet/IP explicit messaging, follow these steps.

Step	Description	Stage
1	Instantiate the EtherNet/IP messaging (EIPM) application template that corresponds to device that you want to use. NOTE: If no device-specific application template exists, use the <i>\$EIPMClient</i> template. Create as many instances of it as you have requests. For example, for one read and one write request you need two instances.	Instantiation
2	Instantiate the <i>\$EIPMPortM</i> EtherNet/IP communication port template and configure its parameters, such as:	

Step	Description	Stage
	<ul style="list-style-type: none"> Port Selection of the Control element: Enables the corresponding LogicX service where X represents the numeric part of the selected port function. It needs to correspond to the maximum number of simultaneous requests that the communication module supports. For example, to EIPMPortM16 corresponds Loigc16. Simultaneous Sends of the LogicX service to optimize communication by reducing the number of simultaneous requests that are sent. 	
3	If you are using one topological communication port with several communication port instances, instantiate the corresponding <i>intermediate template</i> , page 949.	
4	Open the Asset Workspace Editor and link the communication port instance to: <ul style="list-style-type: none"> The device instance. The intermediate template instance (if used). NOTE: In case of a generic device, the device instance is the <i>\$EIPMClient</i> template instance.	
5	Configure an M580 controller and its communication module.	Configuration
6	Create, page 603 the corresponding topological device entity and configure its IP address and other applicable properties (for example, <i>\$EIPMPMESWTHW</i>). NOTE: If no device-specific topological template exists, use the <i>\$EIPGenericDeviceHW</i> generic device template.	
7	Create an logical network.	
8	Connect the communication module of the controller and the topological device entity to the same logical network.	
9	Assign the Control facets of the device, the communication port instances, and the intermediate template instance to the Control Participant project.	Assignment
10	Generate the project.	Generation
11	Map the executable of the Control Participant project to the engine of the controller, page 723.	Mapping (service)
12	Map the EMPortChM interface of the communication port instance (or of the intermediate template instance) to its topological counterpart.	Mapping (hardware)
13	Map the interface of the application instance of the device to the interface of the topological device instance.	

To complete the other steps to engineer the system that contains the controller and the device, follow the system engineering life cycle described in this document.

NOTE: For configuration and setup details of devices and communication ports, refer to the EcoStruxure Process Expert help.

Managing IP Address Changes

The topological templates of devices communicating by using Modbus TCP Ethernet and EtherNet/IP explicit messaging use a variable to define the IP address of the device.

During deployment, the templates set up the initial value of this variable, which is connected to the pin of the DFB implementing the device.

When you change the IP address of a device by changing the hardware mapping of the facet modeling the device and deploy the change by using the **Deploy Changes** command, only the initial value, in this case the IP address, is changed online. The existing value remains the same.

The IP address change becomes effective, either through a cold start of the controller or by replacing the existing value of the variable in the animation table by the new initial value.

Communication Through Modbus Explicit Messaging

Overview

This section contains a procedure describing how to manage communication between devices of the **EPE Managed Devices** product range and M340 or Quantum controllers through Modbus explicit messaging.

It also contains a set of four examples illustrating how to use Modbus serial communication templates and do the required hardware mapping. Each example uses a different architecture based on the M580 platform. The examples use templates of the General Purpose Library Classic.

Refer to the example overview and select the one that applies to your situation. From there, a link allows you to jump directly to the topic describing it.

Communication with Devices Through Modbus Explicit Messaging

About Communication Port Templates for Modbus Explicit Messaging

You can implement communication through Modbus explicit messaging between devices of the **EPE Managed Devices** product range and various controller platforms by using specific communication templates.

Controller platform	Communication module	Communication port template
Modicon M580	BMXNOM**** communication module in the local rack.	<i>\$MBPortM</i>
	BMXNOM**** communication module in a drop.	<i>\$MBPortM58X80</i>
Modicon M340	BMXNOM**** communication module in the local rack.	<i>\$MBPortM</i>
Modicon Quantum	140NOM***** communication module in an X80 drop.	<i>\$MBPortQX80</i>

Communication through an Ethernet port of the controller is also supported.

For a list of supported communication architectures, refer to the EcoStruxure Process Expert help.

NOTE: If no device-specific application template is available for the device that you are using, use the *\$MBClient* template.

About Intermediate Templates

Intermediate templates are used when more than one communication port instance is used according to the following rules:

- *\$MBAddM*: When a physical communication module (BMENOC****, BMXNOE****, or BMXNOM****) needs to pass its port address to more than one communication port instance. Use one template per physical communication port.

This can be case when you are linking more devices to a communication port instance than this instance can handle requests per cycle. For example, The DFB of the *\$MBPortM58X80* communication port template can send a maximum of 4 simultaneous requests per cycle. If you want to send 10 requests per cycle, you must use 3 instances of the *\$MBPortM58X80* template. These 3 instances are linked to an instance of the *\$MBAddM* template.

- *\$EthAddM*: When either of these conditions is fulfilled:
 - A communication module (controller, BMENOC****, or BMXNOE****) needs to pass its Ethernet port address to more than one communication port instance.
 - A BMECRA**** module needs to pass on its IP address to more than one communication port instance.

The intermediate templates allow grouping the interfaces coming from several communication port instances to connect them to the corresponding interfaces of the topology. This is because a 1 to N mapping is not possible for topological mapping interfaces.

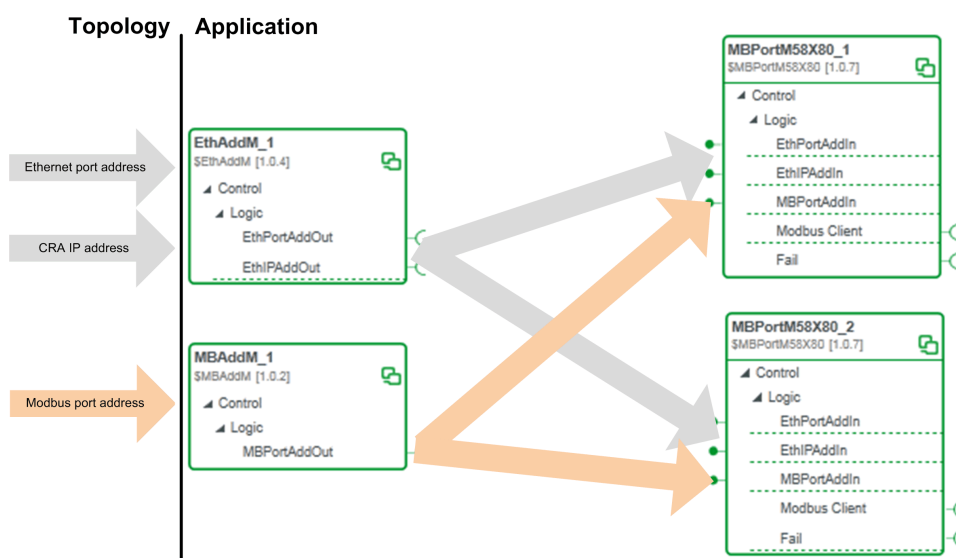
Together, the *\$MBAddM* and *\$EthAddM* templates expose the same interfaces as one *\$MBPortM58X80* template.

The table indicates which intermediate templates to instantiate depending on the communication port template that is used and also indicates the data that is transferred to the communication port instance.

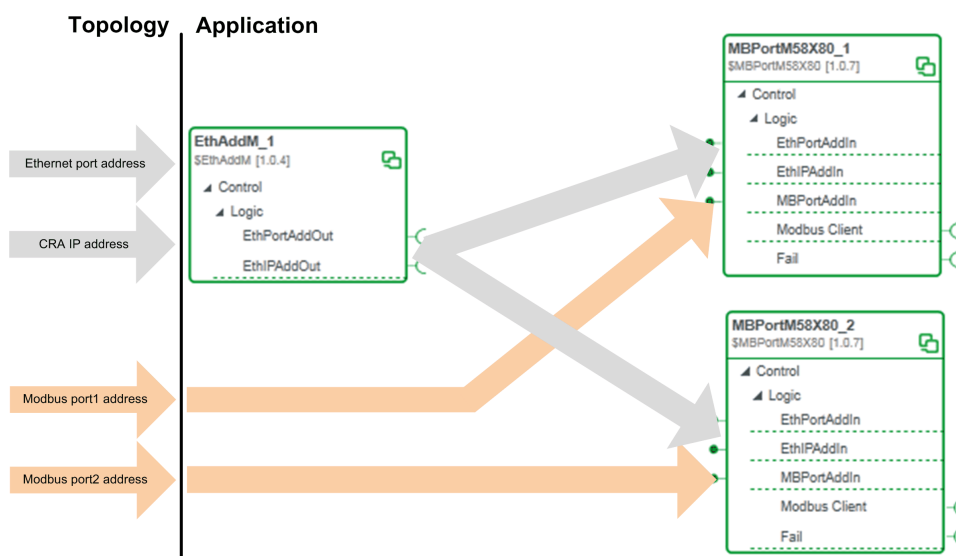
Communication port template	Intermediate template	Transferred address
<i>\$MBPortM</i>	<i>\$MBAddM</i>	Port address of BMXNOM•••• communication module.
<i>\$MBPortM58X80</i>	<i>\$MBAddM</i>	Port address of BMXNOM•••• communication module.
	<i>\$EthAddM</i>	Port address of M580 controller or BMENOC•••• communication module. IP address of the BMECRA•••• module.
<i>\$MBPortQX80</i>	<i>\$MBAddM</i>	Port address of BMXNOM•••• communication module.
	<i>\$EthAddQ</i>	Port address of Quantum controller or 140NOC••••• communication module. IP address of the 140CRA•••• module.

The following figures illustrate the transfer of data between the topology and the application in various scenarios.

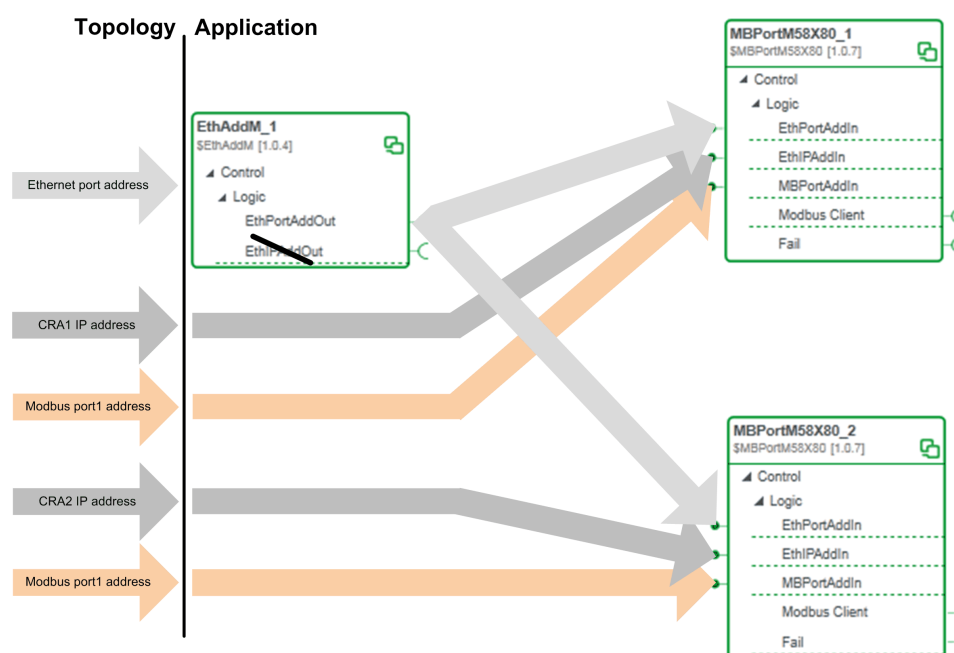
Multiple communication port instances per physical BMXNOM•••• port. In this case, only one BMXNOM•••• port is used and located in a drop.



One communication port instance per physical BMXNOM•••• port. In this case, the two BMXNOM•••• ports are located in the same drop.



One communication port instance per physical BMXNOM.... port. In this case, the two BMXNOM.... ports are located in separate drops.



Implementing Communication with Controllers Through Modbus Port

To implement communication between an M580, M340, or Quantum controller with Modbus port and devices through Modbus explicit messaging, follow these steps.

Step	Description	Stage
1	Instantiate the process template that corresponds to the piece of equipment that you want the device to communicate with (for example, \$Motor).	Instantiation
2	Instantiate a Modbus explicit messaging application template that corresponds to the device that you want to use (for example, \$TesySTMB), and configure its Modbus address. NOTE: If no device-specific application template exists, use the \$MBClient template. Create as many instances of it as you have requests. For example, for one read and one write request you need two instances.	
3	Instantiate the corresponding communication port template, page 953.	
4	If you are using one topological communication port with several communication port instances, instantiate the corresponding intermediate template, page 953.	
5	Open the Asset Workspace Editor and link the device instance to the communication port and the process instance. NOTE: In case of a generic device, the device instance is the \$MBClient template instance.	
6	Configure the controller with a Modbus port as master.	Configuration
7	Assign the Control facets of the process, device, and communication port instances to the Control Participant project so that the execution order, page 394 is as follows: 1. The facet of the Modbus explicit messaging device instance (for example, \$TesySTMB_1). 2. The facet of the communication port instance (for example, MBPort58X80_1).	Assignment
8	Generate the project.	Generation

Step	Description	Stage
9	Map the executable of the Control Participant project to the engine of the controller.	Mapping (service)
10	<p>If the communication module is in the local rack, map by dragging from right to left:</p> <ul style="list-style-type: none"> The EMPortChM or EMPortChQ interface to its topological counterpart. The MBPortChM or MBPortChQ interface to the interface of the BMXNOM**** module of the drop. <p>If the communication module is in a drop, in addition, map by dragging from right to left:</p> <ul style="list-style-type: none"> The CRAIPAddress interface to its topological counterpart. 	Mapping (hardware)

To complete the other steps to engineer the system, follow the system engineering life cycle described in this document.

NOTE: For details on Modbus technology, refer to the EcoStruxure Process Expert help.

Implementing Communication with Controllers Through Ethernet Port

To implement communication between an M340, M580, or Quantum controller with Ethernet port and devices through Modbus explicit messaging, follow these steps.

Step	Description	Stage
1	Instantiate the process template that corresponds to the piece of equipment that you want the device to communicate with (for example, \$Motor).	Instantiation
2	<p>Instantiate a Modbus explicit messaging application template that corresponds to the device that you want to use (for example, \$TsysTMB), and configure the Modbus address.</p> <p>NOTE: If no device-specific application template exists, use the <i>\$MBClient</i> template. Create as many instances of it as you have requests. For example, for one read and one write request you need two instances.</p>	
3	<p>Instantiate the following communication port templates:</p> <ul style="list-style-type: none"> Gateway template \$EGtwMB. \$EMPortM for a controller of the M340 or M580 platform. \$EMPortQ for a controller of the Quantum platform. 	
4	<p>Open the Asset Workspace Editor and link the device instance to the gateway and the process instances.</p> <p>NOTE: In case of a generic device, the device instance is the <i>\$MBClient</i> template instance.</p>	
5	In the Asset Workspace Editor , link the gateway instance to the port instance.	
6	Create a logical network.	Configuration
7	Configure Ethernet communication of the controller.	
8	Connect the communication module of the controller to the logical network.	
9	<p>Assign the Control facets of the process, device, and communication port instances to the Control Participant project so that the execution order, page 394 is as follows:</p> <ol style="list-style-type: none"> The facet of the Modbus explicit messaging device instance (for example, <i>TsysTMB_1</i>). The facet of the gateway instance (for example, <i>EGtwMB_1</i>). The facet of the communication port instance (for example, <i>EMPortM_1</i>). 	Assignment
10	Generate the project.	Generation

Step	Description	Stage
11	Map the executable of the Control Participant project to the engine of the controller.	Mapping (service)
12	Map the EMPortChM or EMPortChQ interface of the communication port instance (CO) to its topological counterpart (HO).	Mapping (hardware)

To complete the other steps to implement the system that contains the controller and devices, follow the system engineering life cycle described in this document.

NOTE: For configuration and setup details of gateways, refer to the EcoStruxure Process Expert help.

Example Overview

Supported Architectures

These examples are based on a controller of the M580 platform configured with one BMENOC0301.2 communication module.

They can be used also for controllers of the Quantum platform by using the corresponding templates with the Q variant (for example, *\$EthAddQ* or *\$MBPortQX80*).

NOTE: The examples do not represent a functional system. Only the elements that are relevant for the examples are configured and illustrated. Default identifiers have been edited to be more comprehensible.

About Communication Port Templates for Modbus Explicit Messaging

Various Modbus serial communication port templates, page 953 exist depending on the architecture, which contains the BMXNOM0200 communication module and its location.

When the communication module is located in a drop, the *\$MBPortM58X80* template is used. Its instance exposes three mapping interfaces (**HW Interface** column) to transfer the required data to the process objects:

- **MBPortChM:** The port address of the BMXNOM•••• communication module.
- **EMPortChM:** The port address of the controller or BMENOC•••• communication module.
- **CRAIPAddress:** The IP address of the BMECRA•••• module.

Available for mapping			
AppInstance	AppFacet	AppTemplate	AppMappingInterface
MBPortM58X80	MBPortM58X80_MBPM58X80	\$MBPortM58X80	MBPortChM
MBPortM58X80	MBPortM58X80_MBPM58X80	\$MBPortM58X80	EMPortChM
MBPortM58X80	MBPortM58X80_MBPM58X80	\$MBPortM58X80	CRAIPAddress

Instantiate one *\$MBPortM58X80* template for each physical communication port used. When you need several communication port templates, use also intermediate templates, page 953.

The *\$MBPortM58X80* template allows mapping with the topology, which exposes the following interfaces in the **Hardware Mapping Editor** when the BMXNOM•••• communication module is located in a drop.

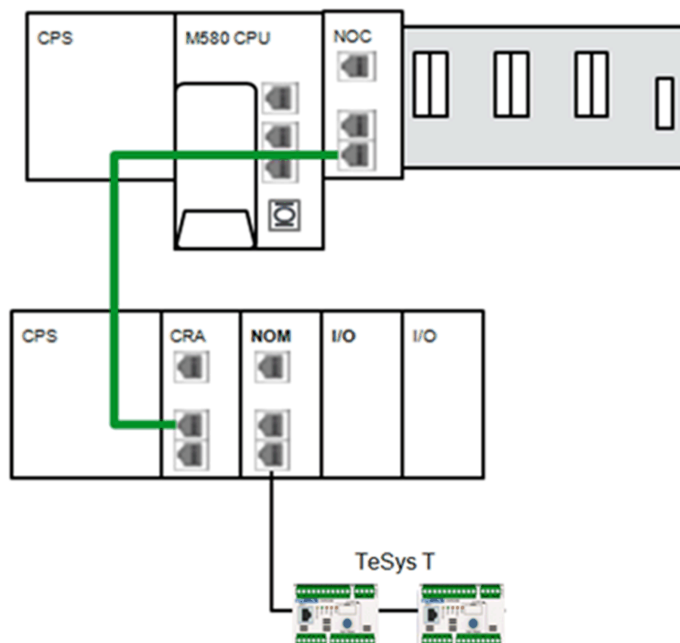
Available for mapping			
HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType
0:Prilocal 0:D 0:R 0:BMEP584040	\$M580CPUethCRP	MBTCPETHM580.EMPortChM	\$MEthernetPortCH/HO
0:Prilocal 0:D 0:R 02:BMENOC0301.2	\$MNOCIOSClient	MBTCPETH.EMPortChM	\$MEthernetPortCH/HO — 1
2:EthRIO 1:D 0:R 0:BMECRA31210.2	\$CRAEth	CRAIPAddress	\$EthernetAddress/HO — 2
2:EthRIO 1:D 0:R 01:BMXNOM0200.3	\$HWNOMMB	MB_M340_M580_Master_CH0.MModbusPortCH	\$MModbusPortCH/HO — 3
2:EthRIO 1:D 0:R 01:BMXNOM0200.3	\$HWNOMMB	MBControllerSlave_CH1.ModbusAddress	\$ModbusAddress/HO

Item	Description
1	Port address of the BMENOC•••• communication module in the local rack.
2	IP address of the BMECRA•••• module.
3	Port address of the BMXNOM•••• communication module when the BMXNOM•••• port is configured for Modbus link.

Example A

The table indicates the specifics of this example in terms of remote drop configuration and application objects used.

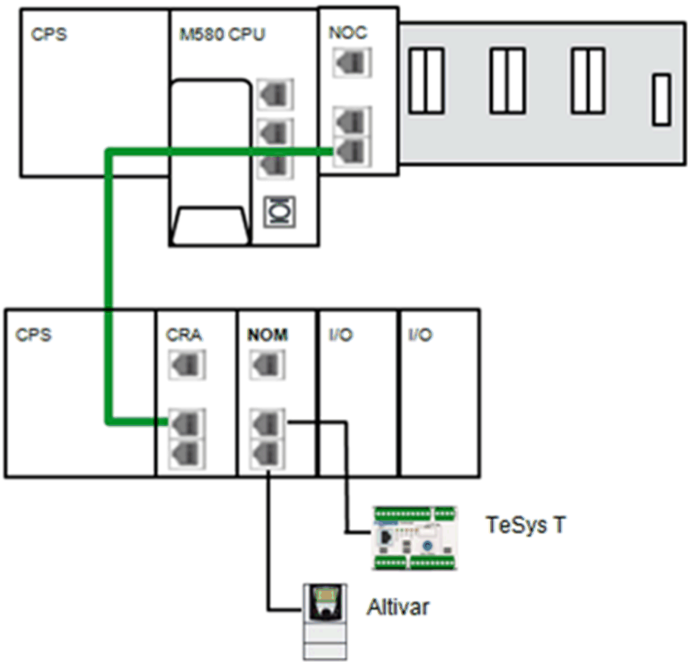
M580 remote drop configuration	Application objects	Example description
1 BMXNOM0200 communication module.	2 devices of the EPE Managed Devices product range communicating through 1 communication port.	A, page 962



Example B

The table indicates the specifics of this example in terms of drop configuration and application objects used.

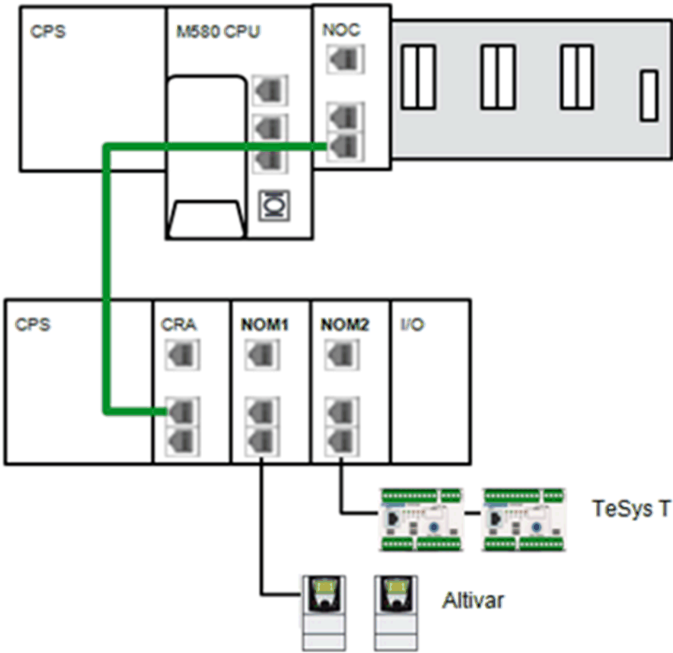
M580 remote drop configuration	Application objects	Example description
1 BMXNOM0200 communication module.	2 devices of the EPE Managed Devices product range, each communicating through 1 communication port (2 ports total).	B, page 966



Example C

The table indicates the specifics of this example in terms of drop configuration and application objects used.

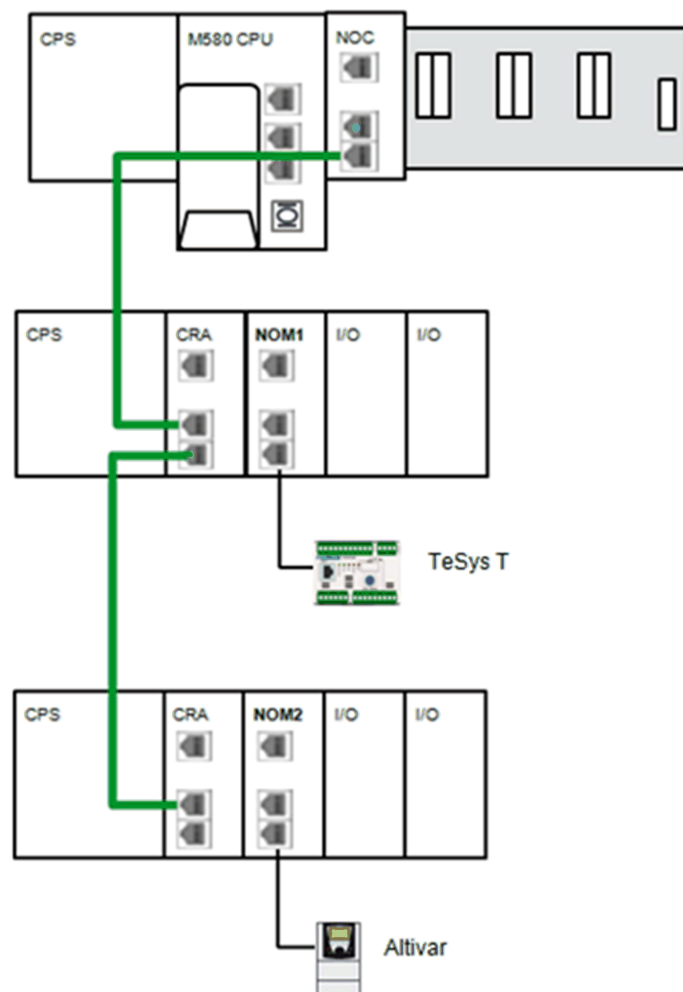
M580 remote drop configuration	Application objects	Example description
2 BMXNOM0200 communication modules	For each BMXNOM••••, 2 devices of the EPE Managed Devices product range communicating through 1 communication port.	C, page 972



Example D

The table indicates the specifics of this example in terms of drop configuration and application objects used.

M580 remote drop configuration	Application objects	Example description
2 drops with 1 BMXNOM0200 communication module in each.	For each drop, 1 device of the EPE Managed Devices product range communicating through 1 communication port.	D, page 979

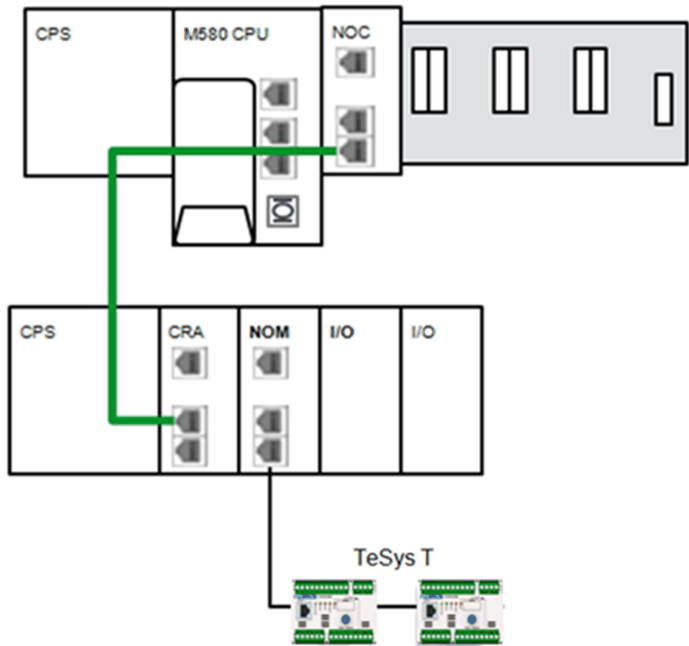


Example A

Example Description

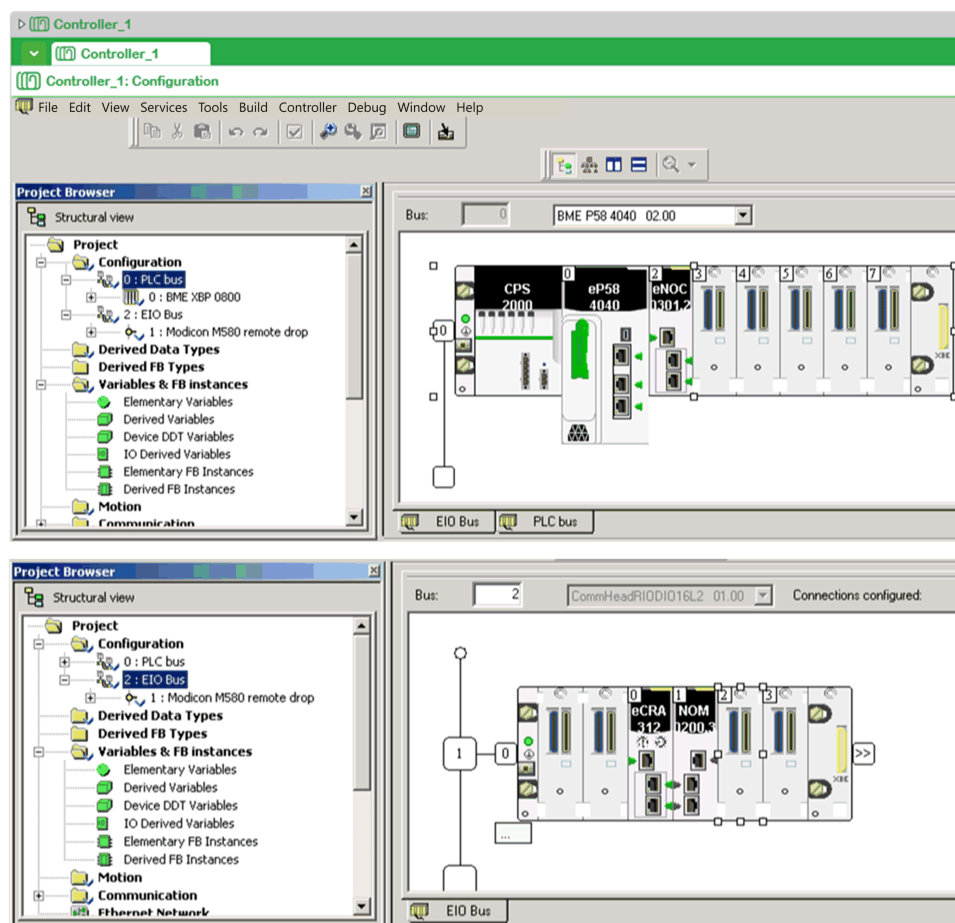
The table indicates the specifics of this example in terms of remote drop configuration and application objects used.

M580 remote drop configuration	Application objects
1 BMXNOM0200 communication module.	2 devices of the EPE Managed Devices product range communicating through 1 communication port.



Configuration Stage

The following figure shows the controller configuration with the drop and the communication module.



Instantiation Stage

The following Instances are required to implement communication at the application level with two devices (TeSys T motor controllers) by using Modbus serial.

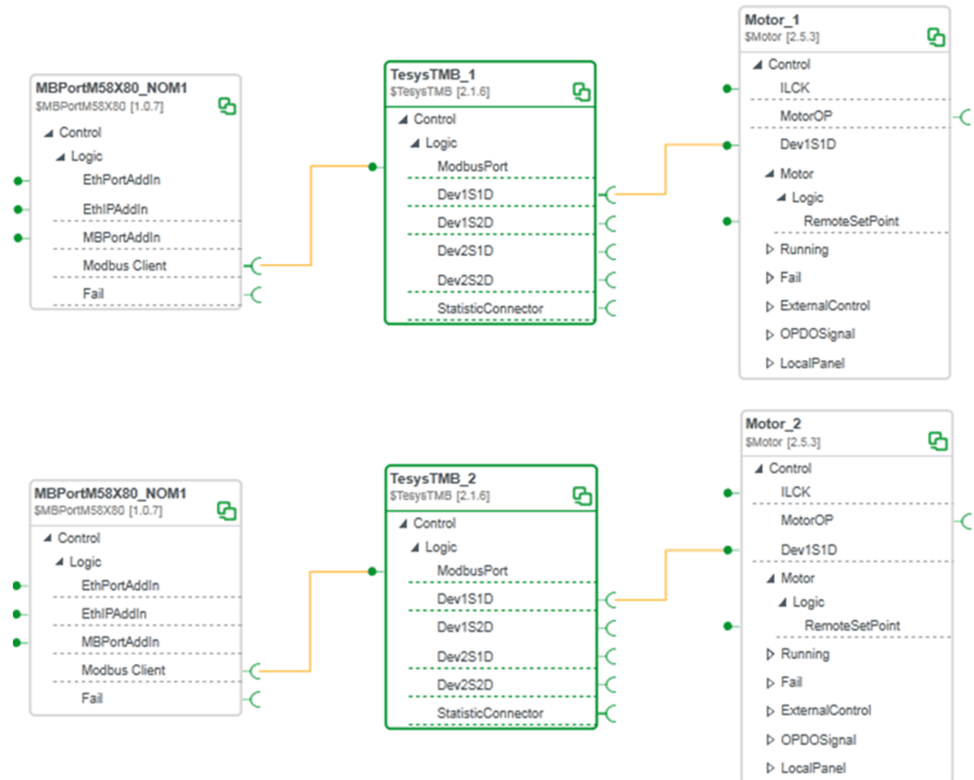
The *MBPortM58X80_NOM1* Modbus serial communication port instance is to be used when the BMXNOM.... communication module is located in the remote drop of an M580 controller. In this example, one instance is created because one physical communication port is used.

Identifier	Template	Version	Data	Link	Assigned State
MBPortM58X80_NOM1	SMBPortM58X80	1.0.7	Valid	Valid	Unassigned
Motor_1	\$Motor	2.5.3	Valid	Valid	Unassigned
Motor_2	\$Motor	2.5.3	Valid	Valid	Unassigned
TesysTMB_1	\$TesysTMB	2.1.6	Valid	Invalid	Unassigned
TesysTMB_2	\$TesysTMB	2.1.6	Valid	Invalid	Unassigned

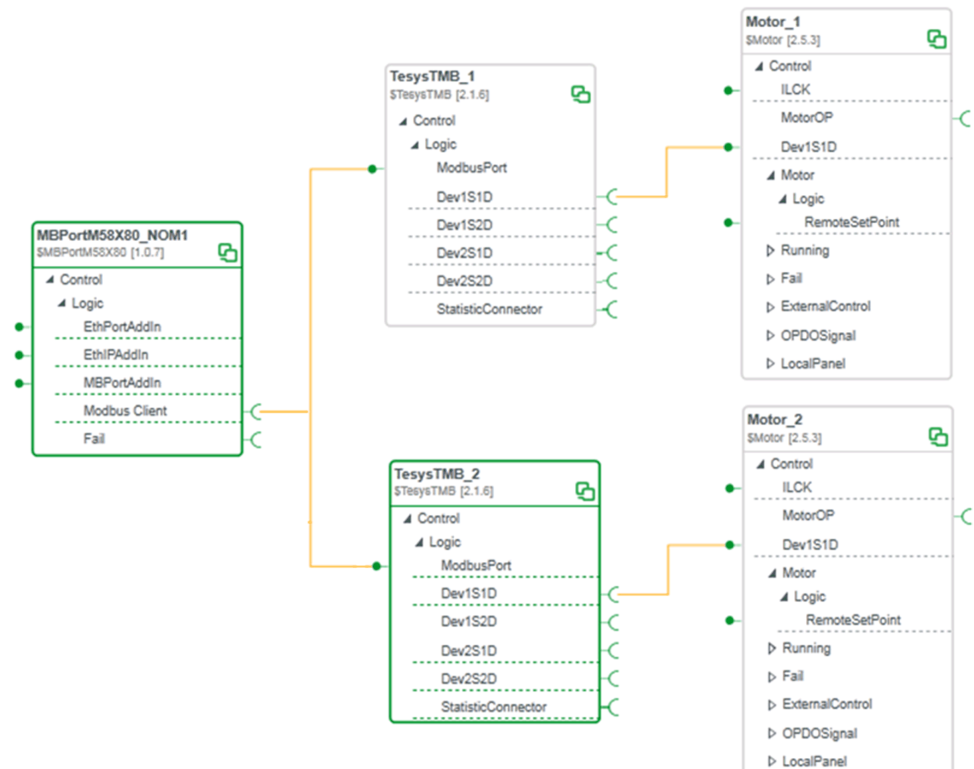
Application Interface Links

The following links are created in the **Asset Workspace Editor** for the TeSys T devices.

The process instance (*Motor_n*), the device instance (*TesysTMB_n*), and the communication port instance (*MBPortM58X80_{NOM1}*) are linked.



These steps result in the following links.

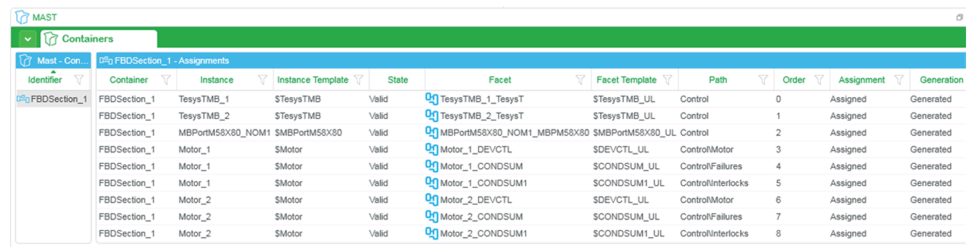


The communication port instance (*MBPortM58X80_{NOM1}*) passes the following data to the process instances:

- Port address of the BMENOC•••• communication module
- IP address of the BMECRA•••• module of the drop
- Port address of the BMXNOM•••• communication module

Control Facet Assignment

Facets of the instances are assigned to a Control Participant project in an FBD section.



Identifier	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
FBDSection_1	FBDSection_1	TesysTMB_1	\$TesysTMB	Valid	0q TesysTMB_1_TesysT	\$TesysTMB_UL	Control	0	Assigned	Generated
FBDSection_1	FBDSection_1	TesysTMB_2	\$TesysTMB	Valid	0q TesysTMB_2_TesysT	\$TesysTMB_UL	Control	1	Assigned	Generated
FBDSection_1	FBDSection_1	MBPortM58X80_NOM1	\$MBPortM58X80	Valid	0q MBPortM58X80_NOM1_MBPMS58X80	\$MBPortM58X80_UL	Control	2	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_1	\$Motor	Valid	0q Motor_1_DEVCTL	\$DEVCTL_UL	ControlMotor	3	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_1	\$Motor	Valid	0q Motor_1_CONDSUM	\$CONDSUM_UL	ControlFailures	4	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_1	\$Motor	Valid	0q Motor_1_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	5	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_2	\$Motor	Valid	0q Motor_2_DEVCTL	\$DEVCTL_UL	ControlMotor	6	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_2	\$Motor	Valid	0q Motor_2_CONDSUM	\$CONDSUM_UL	ControlFailures	7	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_2	\$Motor	Valid	0q Motor_2_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	8	Assigned	Generated

Generation and Service Mapping

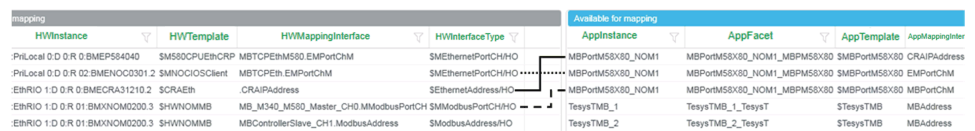
The Control Participant project is generated.

An executable is created in the Control Participant and mapped to the M580 controller.

Hardware Mapping

The **Hardware Mapping Editor** shows the following interfaces.

(Lines have been drawn to show the mappings that you must do.)



HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	Available for mapping
PrLLocal 0.D 0.R 0.BMEP584040	\$M580CPUeHCRP	MBTCPeHmM580.EMPorChM	\$MEthernetPortCHHO	MBPortM58X80_NOM1
PrLLocal 0.D 0.R 02.BMENOC0301.2	\$MNOCIOClient	MBTCPeHm.EMPorChM	\$MEthernetPortCHHO	MBPortM58X80_NOM1
EBRIO 1.D 0.R 0.BMECRA31210.2	\$CRAIEth	.CRAIPAddress	\$SEthernetAddressHO	MBPortM58X80_NOM1
EBRIO 1.D 0.R 01.BMEXNOM0200.3	\$HWNOMMB	MB_M340_M580_Master_CH0.MModbusPortCH	\$MModbusPortCHHO	TesysTMB_1
EBRIO 1.D 0.R 01.BMXXNOM0200.3	\$HWNOMMB	MBControllerSlave_CH1.ModbusAddress	\$ModbusAddressHO	TesysTMB_2

NOTE: You do not need to map **MBAddress** interfaces (for example, of facet *TesysTMB_1_TesysT*).

Completing the System Engineering Life Cycle

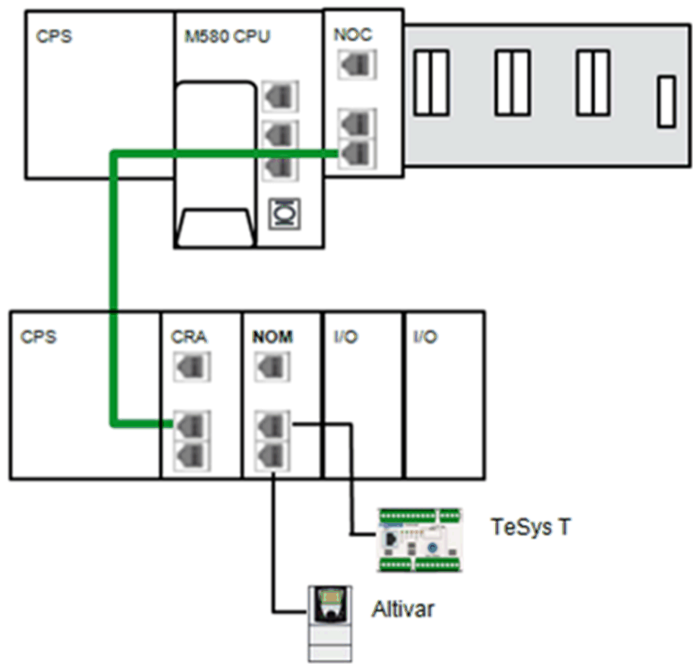
Complete the creation of the system by performing the other steps of the system engineering life cycle.

Example B

Example Description

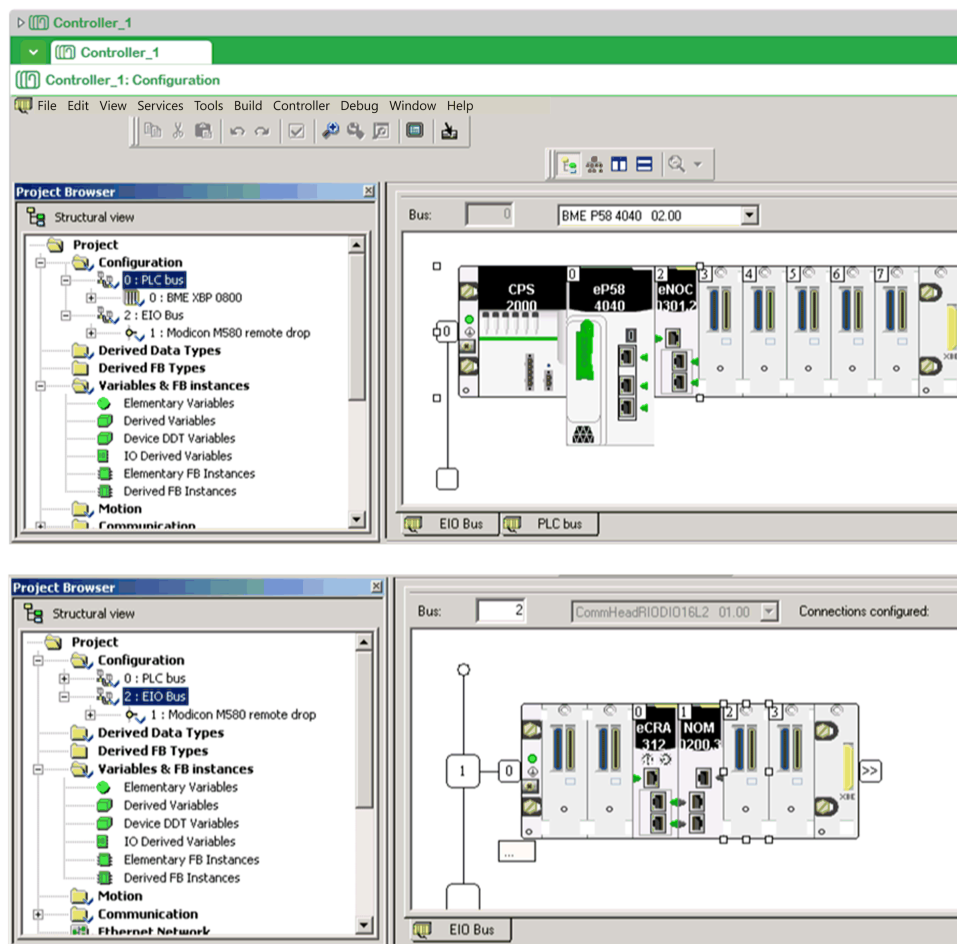
The table indicates the specifics of this example in terms of drop configuration and application objects used.

M580 remote drop configuration	Application objects
1 BMXNOM0200 communication module.	2 devices of the EPE Managed Devices product range, each communicating through 1 communication port (2 ports total).



Configuration Stage

The following figure shows the controller configuration with the drop and the communication module.



Instantiation Stage

The following Instances are required to implement communication at the application level with two devices (a TeSys T motor controller and an Altivar variable speed drive) by using Modbus serial.

The *MBPortM58X80_x* Modbus serial communication port instances are to be used when the BMXNOM**** communication module is located in the drop of an M580 controller. In this example, two instances are created, one for each physical communication port used.

Instances are grouped in two folders for clarity, one for each communication port instance.

B						
Identifier	Template	Version	Data	Link	Assigned State	
EthAddM_NOC_CRA	\$EthAddM	1.0.4	Valid	Valid	Unassigned	
MBPortM58X80_Port1	\$MBPortM58X80	1.0.7	Valid	Valid	Unassigned	
Motor_1	\$Motor	2.5.3	Valid	Valid	Unassigned	
TesysTMB_1	\$TesysTMB	2.1.6	Valid	Invalid	Unassigned	

B_2						
Identifier	Template	Version	Data	Link	Assigned State	
ATV12MB_2	\$ATV12MB	2.1.7	Valid	Invalid	Unassigned	
MBPortM58X80_Port2	\$MBPortM58X80	1.0.7	Valid	Valid	Unassigned	
Motor_2	\$Motor	2.5.3	Valid	Valid	Unassigned	

These steps result in the following links.

The communication port instance (*MBPortM58X80_NOM1*) passes the following data to the process instances:

- Port address of the BMENOC.... communication module
- IP address of the BMENOC.... module of the drop
- Port address of the BMXNOM.... communication module

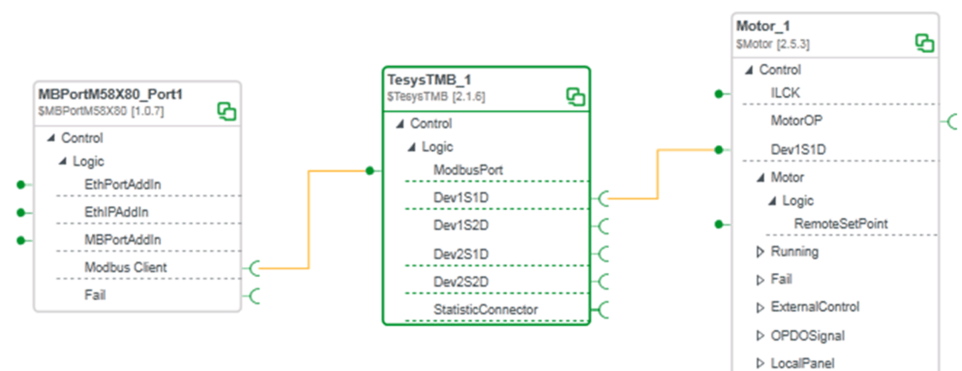
Intermediate Template Instances

Only intermediate template *\$EthAddM_NOC_CRA* is used because more than one communication port instance is used. It is common to both communication port instances. Intermediate template *\$MBAddM* is not used because on each *BMXNOM....* port, only one communication port instance is used.

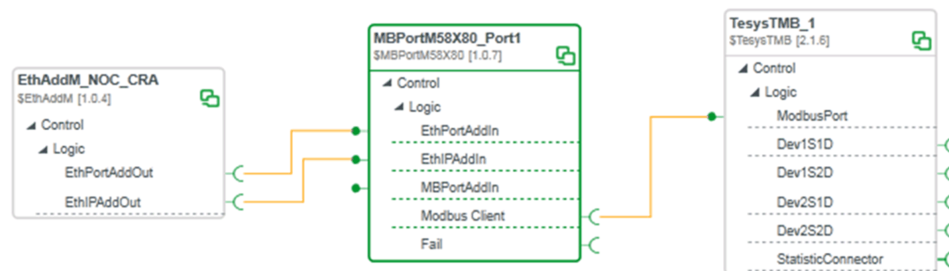
Creating Links

The following links are created in the **Asset Workspace Editor** for the TeSys T device.

The process instance (*Motor_1*), the device instance (*TesysTMB_1*), and the communication port instance (*MBPortM58X80_Port1*) are linked.

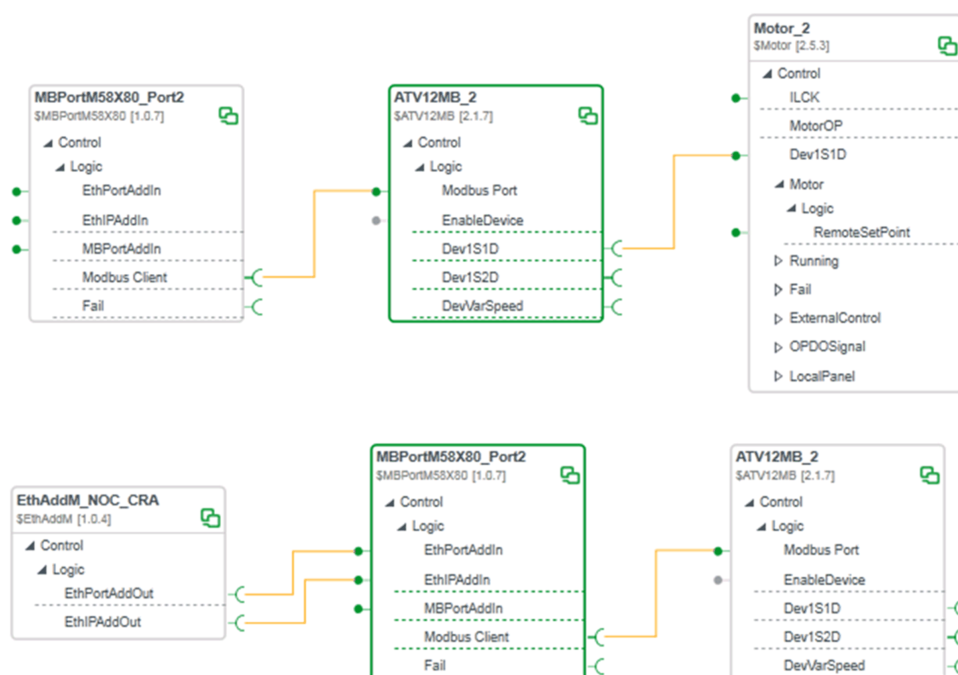


Then, link the *MBPortM58X80_Port1* instance to both interfaces of the *EthAddM_NOC_CRA* Ethernet intermediate template instance.

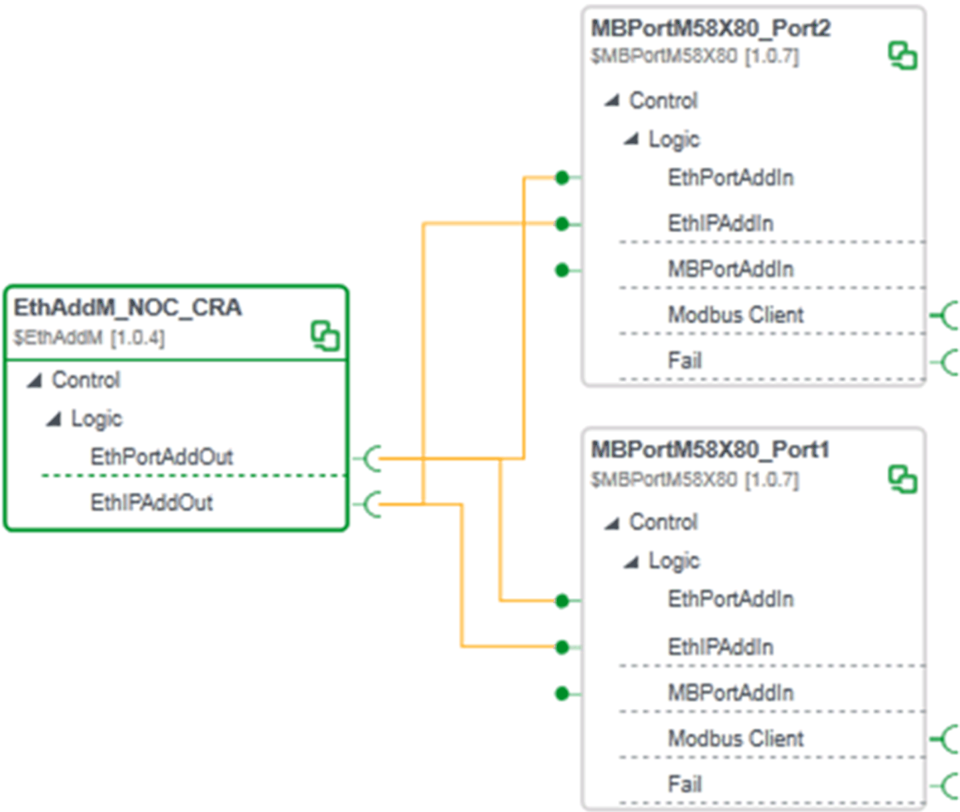


The *EthAddM_NOC_CRA* instance passes the BMENOC.... port address as well as the IP address of the *BMECRA0312* module to the communication port instance.

Similarly, the following links are created in the **Asset Workspace Editor** for the Altivar device.



The following figure show the existing links from the perspective of the intermediate template.



Control Facet Assignment

Facets of the instances are assigned to a Control Participant project in two separate FBD sections.

Folder *B* is assigned to *FBDSection_1*.

MAST										
Containers										
FBDSection_1 - Assignments										
Identifier	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
FBDSection_1	FBDSection_1	TesysTMB_1	\$TesysTMB	Valid	TesysTMB_1_TesysT	\$TesysTMB_UL	Control	0	Assigned	Generated
FBDSection_2	FBDSection_1	MBPortM58X80_Port1	\$MBPortM58X80	Valid	MBPortM58X80_Port1_MBPMS58X80	\$MBPortM58X80_UL	Control	1	Assigned	Generated
	FBDSection_1	EthAddM_NOC_CRA	\$EthAddM	Valid	EthAddM_NOC_CRA_PA	\$EthAddM_UL	Control	2	Assigned	Generated
	FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_DEVCTL	\$DEVCTL_UL	ControlMotor	3	Assigned	Generated
	FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM	\$CONDSUM_UL	ControlFailures	4	Assigned	Generated
	FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	5	Assigned	Generated

Folder *B_2* is assigned to *FBDSection_2*.

MAST										
Containers										
FBDSection_2 - Assignments										
Identifier	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
FBDSection_1	FBDSection_2	ATV12MB_2	\$ATV12MB	Valid	ATV12MB_2_ATV	\$ATV12MB_UL	Control	0	Assigned	Generated
FBDSection_2	FBDSection_2	MBPortM58X80_Port2	\$MBPortM58X80	Valid	MBPortM58X80_Port2_MBPMS58X80	\$MBPortM58X80_UL	Control	1	Assigned	Generated
	FBDSection_2	Motor_2	\$Motor	Valid	Motor_2_DEVCTL	\$DEVCTL_UL	ControlMotor	2	Assigned	Generated
	FBDSection_2	Motor_2	\$Motor	Valid	Motor_2_CONDSUM	\$CONDSUM_UL	ControlFailures	3	Assigned	Generated
	FBDSection_2	Motor_2	\$Motor	Valid	Motor_2_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	4	Assigned	Generated

Generation and Service Mapping

The Control Participant project is generated.

An executable is created in the Control Participant and mapped to the M580 controller.

Hardware Mapping

The **Hardware Mapping Editor** shows the following interfaces.

(Lines have been drawn to show the mappings that you must do.)

mapping				Available for mapping			
HWInstance	HWTemplate	HWMMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface
PriLocal 0:D 0:R 0:BMENP584040	SM580CPUethCRP	MBTCPethM580.EMPotCHM	SEthernetPortCH/HO	ATV12MB_2	ATV12MB_2_ATV	SATV12MB	MBAAddress
PriLocal 0:D 0:R 02:BMENOC03012	SMNOCIOSClient	MBTCPeth.EMPotCHM	SEthernetPortCH/HO	EthAddM_NOC_CRA	EthAddM_NOC_CRA_PA	SEthAddM	EMPotCHM
EthRIO 1:D 0:R 0:BMENOCRA312102	SCRAEth	.CRAIPAddress	SEthernetAddress/HO	EthAddM_NOC_CRA	EthAddM_NOC_CRA_PA	SEthAddM	CRAIPAddress
EthRIO 1:D 0:R 01:BMXNOM02003	SHWNOMMB	MB_M340_M580_Master_CH0.MModbusPortCH	MMModbusPortCH/HO	MBPortM58X80_Port1	MBPortM58X80_Port1_MBPMS8X80	SMBPortM58X80	MBPortCHM
EthRIO 1:D 0:R 01:BMXNOM02003	SHWNOMMB	MB_M340_M580_Master_CH1.MModbusPortCH	MMModbusPortCH/HO	MBPortM58X80_Port2	MBPortM58X80_Port2_MBPMS8X80	SMBPortM58X80	MBPortCHM
				TesysTMB_1	TesysTMB_1_TesysT	STesysTMB	MBAAddress

NOTE: You do not need to map **MBAAddress** interfaces (for example, of facet *TesysTMB_1_TesysT*).

Completing the System Engineering Life Cycle

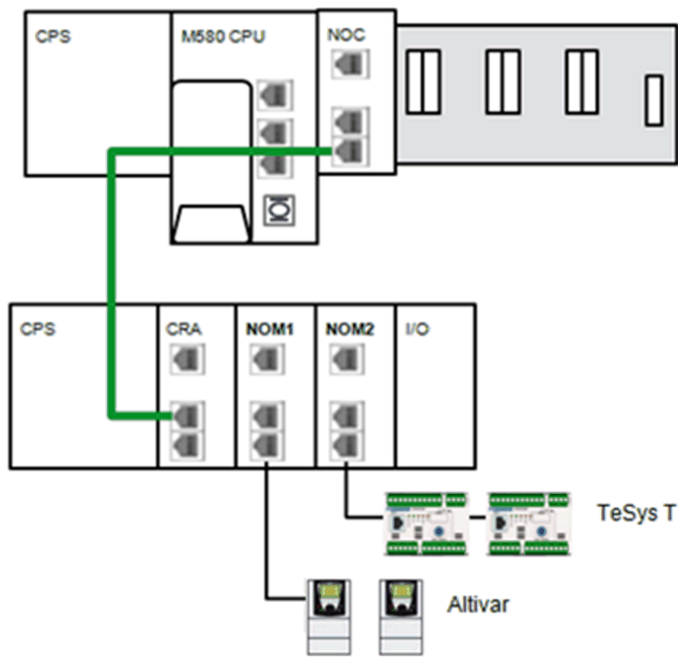
Complete the creation of the system by performing the other steps of the system engineering life cycle.

Example C

Example Description

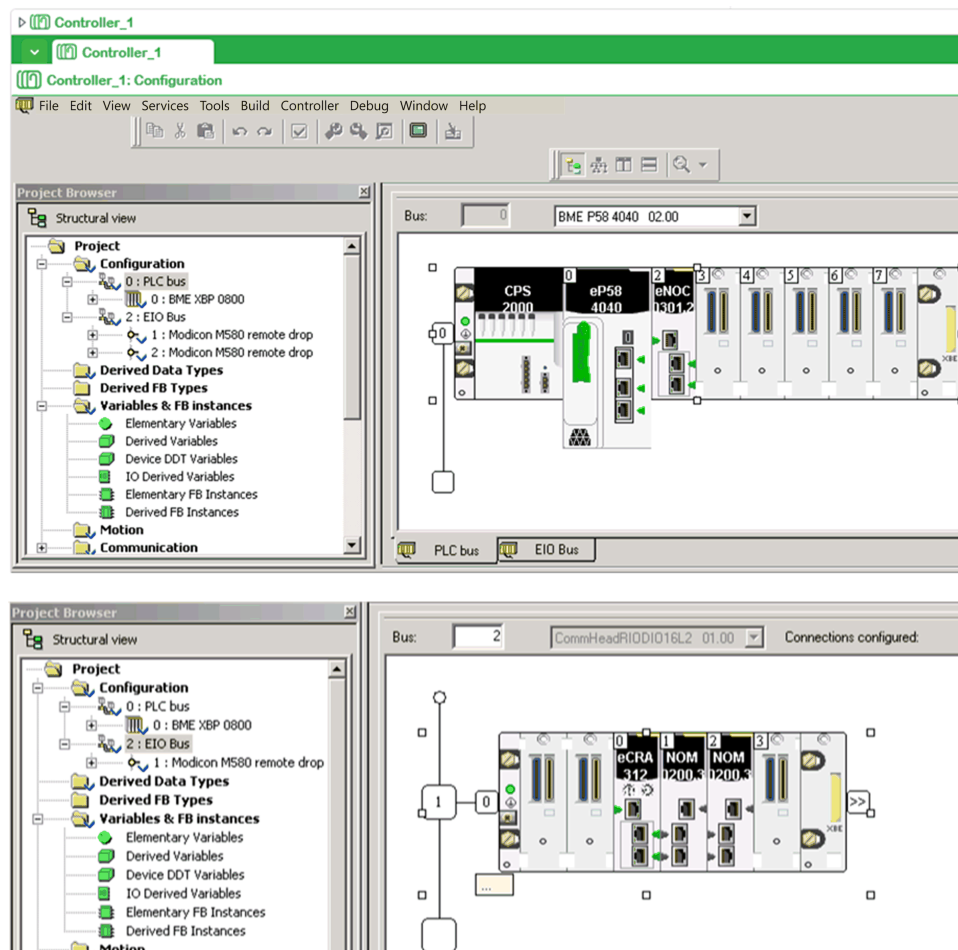
The table indicates the specifics of this example in terms of drop configuration and application objects used.

M580 remote drop configuration	Application objects
2 BMXNOM0200 communication modules	For each module, 2 devices of the EPE Managed Devices product range communicating through 1 communication port.



Configuration Stage

The following figure shows the controller configuration with the drop and the communication module.



Instantiation Stage

The following Instances are required to implement communication at the application level with four devices (two TeSys T motor controllers and two Altivar variable speed drives) by using Modbus serial.

The *MBPortM58X80_x* Modbus serial communication port instances are to be used when the BMXNOM**** communication module is located in the drop of an M580 controller. In this example, two instances are created, one for each physical communication port used.

Instances are grouped in two folders for clarity, one for each communication port instance.

C						
Identifier	Template	Version	Data	Link	Assigned State	
EthAddM_NOC_CRA	\$EthAddM	1.0.4	Valid	Valid	Unassigned	
MBPortM58X80_NOM1	\$MBPortM58X80	1.0.7	Valid	Valid	Unassigned	
Motor_10	\$Motor	2.5.3	Valid	Valid	Unassigned	
Motor_11	\$Motor	2.5.3	Valid	Valid	Unassigned	
TesysTMB_10	\$TesysTMB	2.1.6	Valid	Invalid	Unassigned	
TesysTMB_11	\$TesysTMB	2.1.6	Valid	Invalid	Unassigned	

C_2						
Identifier	Template	Version	Data	Link	Assigned State	
ATV61MB_20	\$ATV61MB	1.2.5	Valid	Invalid	Unassigned	
ATV61MB_21	\$ATV61MB	1.2.5	Valid	Invalid	Unassigned	
MBPortM58X80_NOM2	\$MBPortM58X80	1.0.7	Valid	Valid	Unassigned	
Motor_20	\$Motor	2.5.3	Valid	Valid	Unassigned	
Motor_21	\$Motor	2.5.3	Valid	Valid	Unassigned	

These steps result in the following links.

The communication port instance (MBPortM58X80_NOM1) passes the following data to the process instances:

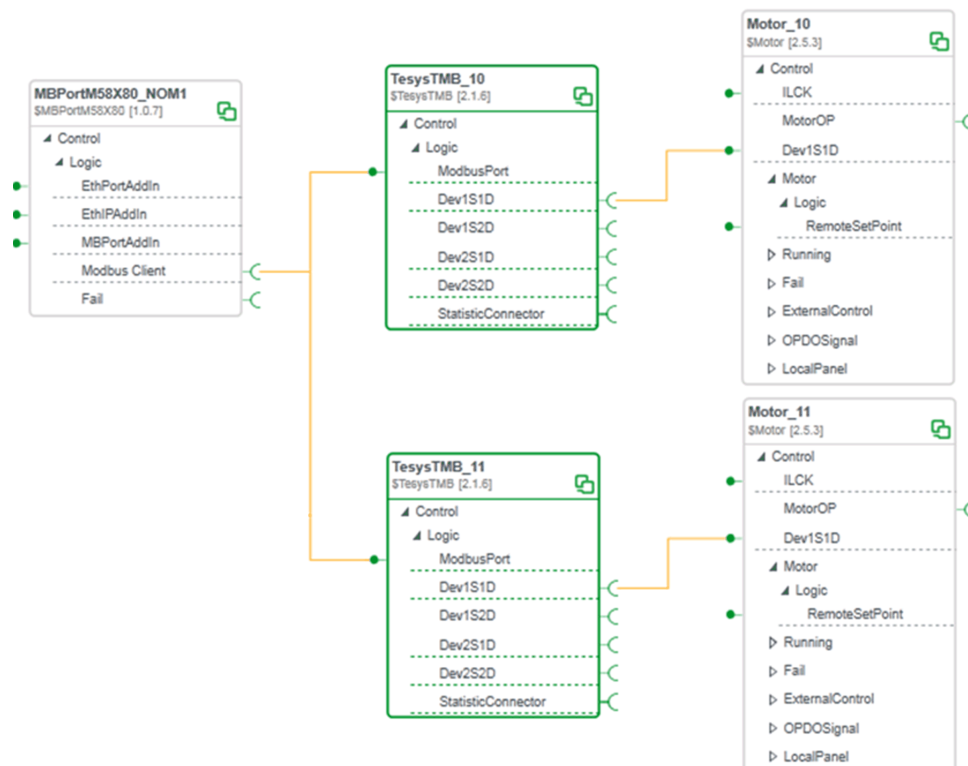
- Port address of the BMENOC.... communication module
- IP address of the BMECRA.... module of the drop
- Port address of the BMXNOM.... communication module

Intermediate Template Instances

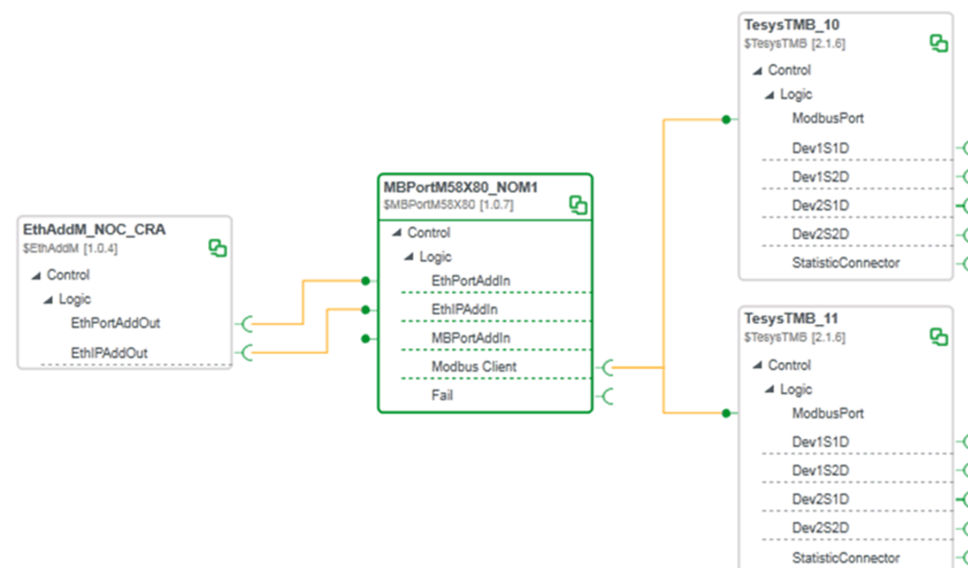
Only intermediate template *\$EthAddM_NOC_CRA* is used because more than one communication port instance is used. It is common to both communication port instances. Intermediate template *\$MBAddM* is not used because on each BMXNOM.... port, only one communication port instance is used.

Application Interface Links

The following links are created in the **Asset Workspace Editor** for both TeSys T devices.

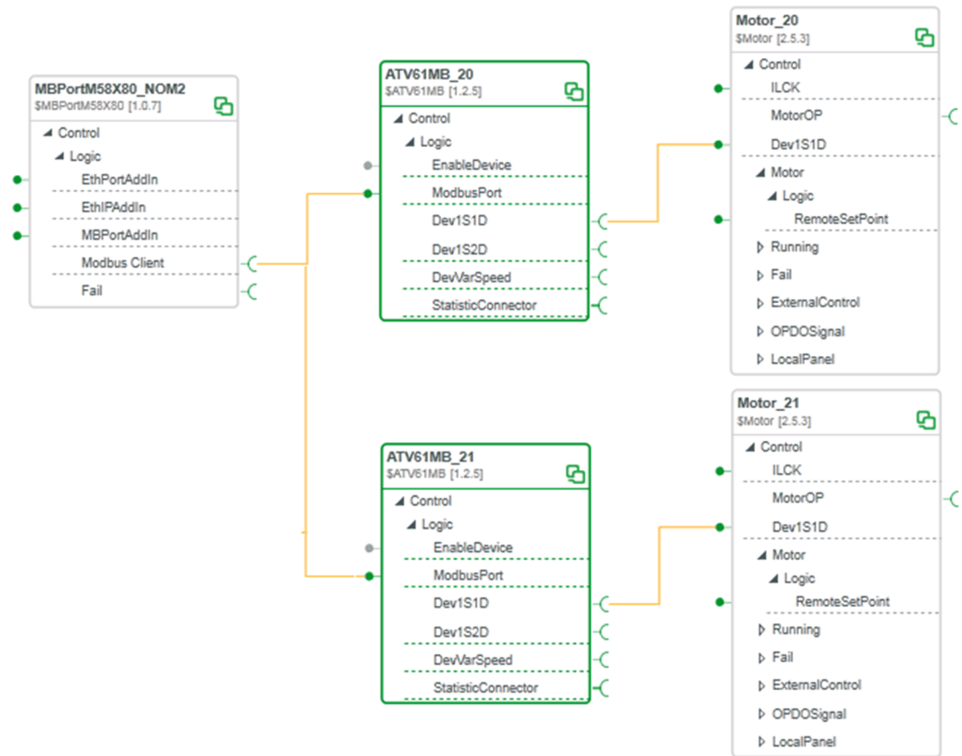


Then, link the communication port instance *MBPortM58X80_NOM1*, which communicates with the *BMXNOM*... communication module located in drop 1 to both interfaces of the intermediate template instance *EthAddM_NOC_CRA*.

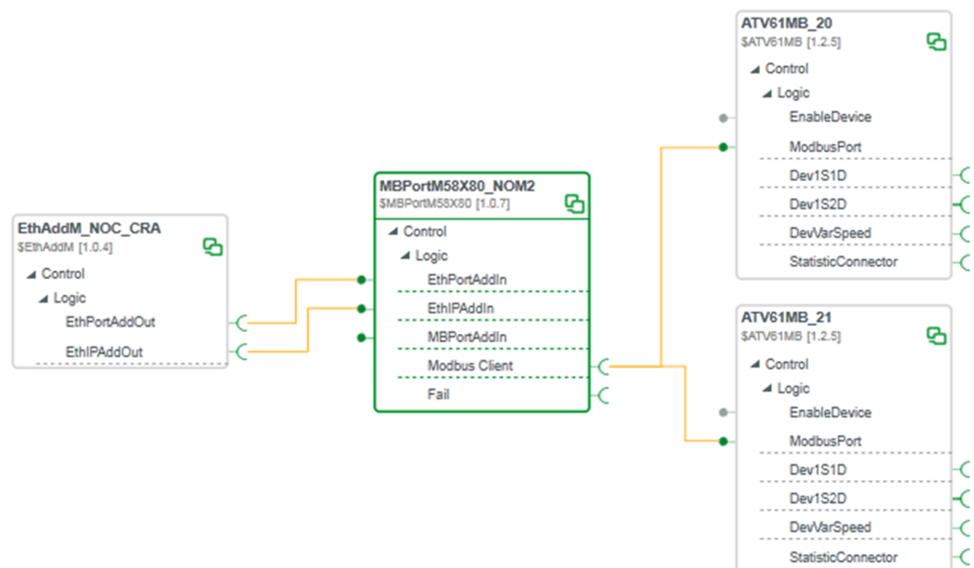


The *EthAddM_NOC_CRA* instance passes the physical *BMENOC*... port address as well as the IP address of the *BMECRA0312* module to the first communication port instance.

Similarly, the following links are created in the **Asset Workspace Editor** for both Altivar devices.

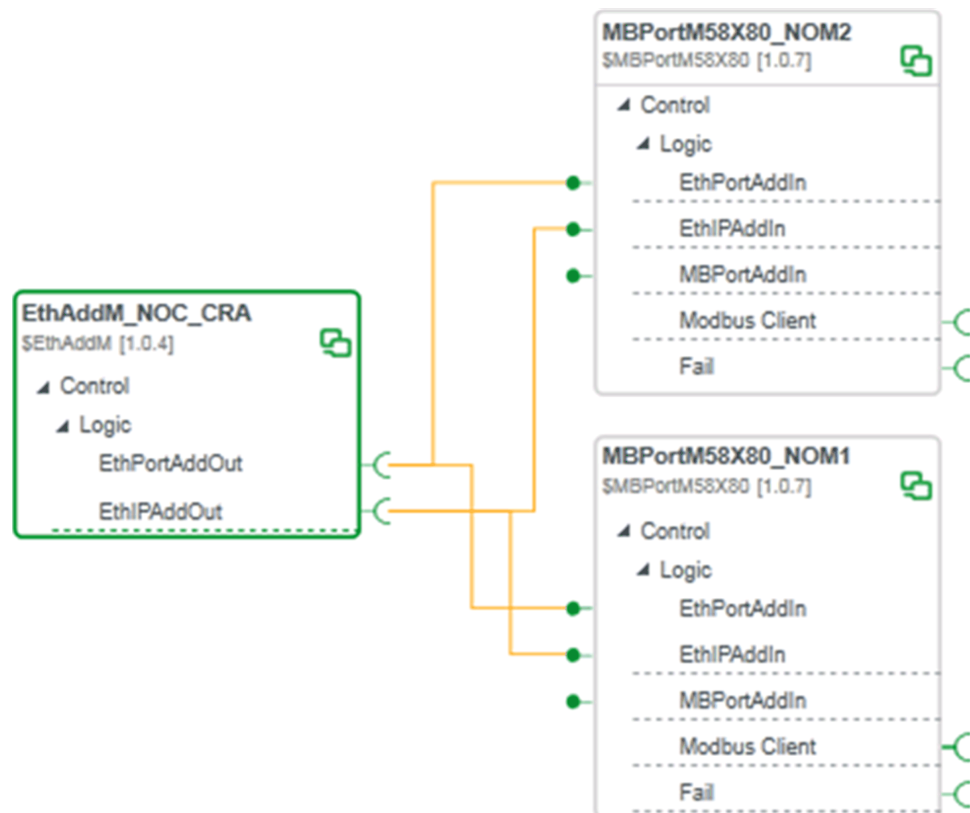


Proceed similarly with the communication port instance *MBPortM58X80_NOM2*, which communicates with the *BMXNOM*.... communication module located in drop 2 by linking it to both interfaces of the intermediate template instance *EthAddM_NOC_CRA*.



The *EthAddM_NOC_CRA* instance passes the physical *BMENOC*.... port address as well as the IP address of the *BMECRA0312* module to the second communication port instance.

The figure shows the existing links from the perspective of the intermediate template.



Control Facet Assignment

Facets of the instances are assigned to a Control Participant project in two separate FBD sections.

Folder C is assigned to *FBDSection_1*.

MAST										
Containers										
FBDSection_1 - Assignments										
Identifier	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
FBDSection_1	FBDSection_1	TesysTMB_10	\$TesysTMB	Valid	Q TesysTMB_10_TesysT	\$TesysTMB_UL	Control	0	Assigned	Generated
FBDSection_1	FBDSection_1	TesysTMB_11	\$TesysTMB	Valid	Q TesysTMB_11_TesysT	\$TesysTMB_UL	Control	1	Assigned	Generated
FBDSection_1	FBDSection_1	MBPortM58X80_NOM1	\$MBPortM58X80	Valid	Q MBPortM58X80_NOM1_MBP	\$MBPortM58X80_UL	Control	2	Assigned	Generated
FBDSection_1	FBDSection_1	EthAddM_NOC_CRA	\$EthAddM	Valid	Q EthAddM_NOC_CRA_PA	\$EthAddM_UL	Control	3	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_10	\$Motor	Valid	Q Motor_10_DEVCTL	\$DEVCTL_UL	ControlMotor	4	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_10	\$Motor	Valid	Q Motor_10_CONDSUM	\$CONDSUM_UL	ControlFailures	5	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_10	\$Motor	Valid	Q Motor_10_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	6	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_11	\$Motor	Valid	Q Motor_11_DEVCTL	\$DEVCTL_UL	ControlMotor	7	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_11	\$Motor	Valid	Q Motor_11_CONDSUM	\$CONDSUM_UL	ControlFailures	8	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_11	\$Motor	Valid	Q Motor_11_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	9	Assigned	Generated

Folder C_2 is assigned to *FBDSection_2*.

MAST										
Containers										
FBDSection_2 - Assignments										
Identifier	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
FBDSection_2	FBDSection_2	ATV61MB_20	\$ATV61MB	Valid	Q ATV61MB_20_ATV	\$ATV61MB_UL	Control	0	Assigned	Generated
FBDSection_2	FBDSection_2	ATV61MB_21	\$ATV61MB	Valid	Q ATV61MB_21_ATV	\$ATV61MB_UL	Control	1	Assigned	Generated
FBDSection_2	FBDSection_2	MBPortM58X80_NOM2	\$MBPortM58X80	Valid	Q MBPortM58X80_NOM2_MBP	\$MBPortM58X80_UL	Control	2	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_20	\$Motor	Valid	Q Motor_20_DEVCTL	\$DEVCTL_UL	ControlMotor	3	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_20	\$Motor	Valid	Q Motor_20_CONDSUM	\$CONDSUM_UL	ControlFailures	4	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_20	\$Motor	Valid	Q Motor_20_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	5	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_21	\$Motor	Valid	Q Motor_21_DEVCTL	\$DEVCTL_UL	ControlMotor	6	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_21	\$Motor	Valid	Q Motor_21_CONDSUM	\$CONDSUM_UL	ControlFailures	7	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_21	\$Motor	Valid	Q Motor_21_CONDSUM1	\$CONDSUM1_UL	ControlInterlocks	8	Assigned	Generated

Generation and Service Mapping

The Control Participant project is generated.

An executable is created in the Control Participant and mapped to the M580 controller.

Hardware Mapping

The **Hardware Mapping Editor** shows the following interfaces.
(Lines have been drawn to show the mappings that you must do.)

Available for mapping				Available for mapping			
HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInterface
0:PHILocal:0:D:0:R:0:BMEP584040	SM580CPUeInCRP	MBTCPeInM580:EMPorCHM	SMEthernetPortCHHO	ATV61MB_20	ATV61MB_20_ATV	SATV61MB	MBAddress
0:PHILocal:0:D:0:R:02:BMENOC0301:2	SMNOCIOSClient	MBTCPeIn:EMPorCHM	SMEthernetPortCHHO	ATV61MB_21	ATV61MB_21_ATV	SATV61MB	MBAddress
2:ETHRIO:1:D:0:R:0:BMECRA31210:2	SCRAETH	:CRAIPAddress	SEthernetAddressHO	ETHAddM_NOC_CRA	ETHAddM_NOC_CRA_PA	SEthAddM	EMPorCHM
2:ETHRIO:1:D:0:R:01:BMXNOM2000:3	SHWNOMMB	MB_M340_M580_Master_CH0.MModbusPortCH	SMModbusPortCHHO	ETHAddM_NOC_CRA	ETHAddM_NOC_CRA_PA	SEthAddM	CRAIPAddress
2:ETHRIO:1:D:0:R:01:BMXNOM2000:3	SHWNOMMB	MBControllerSlave_CH1.ModbusAddress	SMModbusAddressHO	MBPortM58X80_NOM1	MBPortM58X80_NOM1_MBPMS58X80	SMBPMS58X80	MBPortCHM
2:ETHRIO:1:D:0:R:02:BMXNOM2000:3	SHWNOMMB	MB_M340_M580_Master_CH0.MModbusPortCH	SMModbusPortCHHO	MBPortM58X80_NOM2	MBPortM58X80_NOM2_MBPMS58X80	SMBPMS58X80	MBPortCHM
2:ETHRIO:1:D:0:R:02:BMXNOM2000:3	SHWNOMMB	MBControllerSlave_CH1.ModbusAddress	SMModbusAddressHO	TesysTMB_10	TesysTMB_10_TesysT	STesysTMB	MBAddress
				TesysTMB_11	TesysTMB_11_TesysT	STesysTMB	MBAddress

NOTE: You do not need to map **MBAddress** interfaces (for example, of facet *TesysTMB_10_TesysT*).

Completing the System Engineering Life Cycle

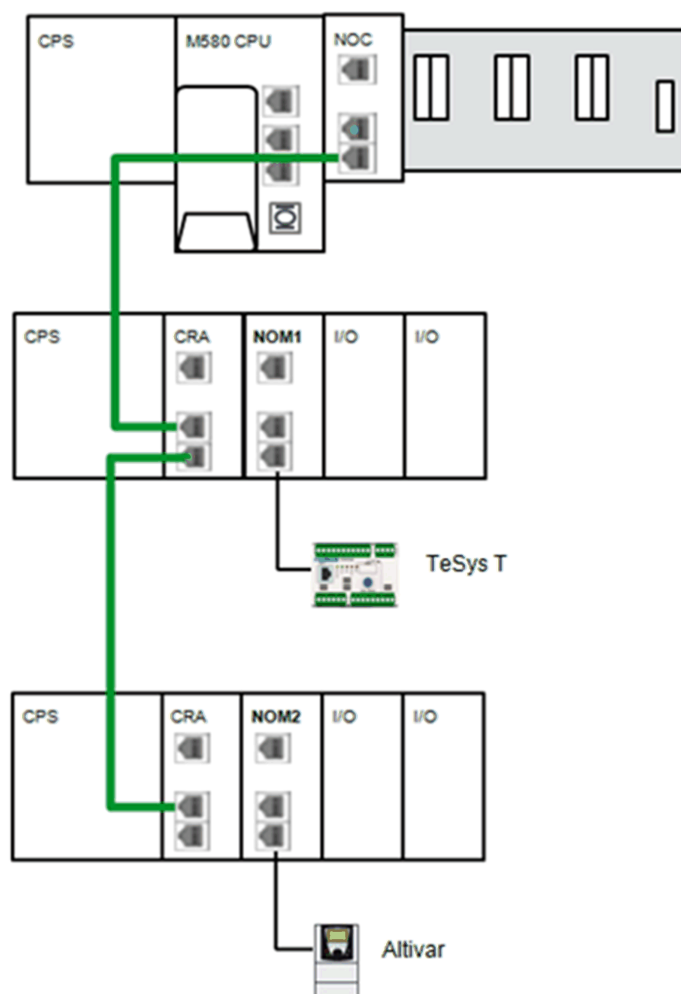
Complete the creation of the system by performing the other steps of the system engineering life cycle.

Example D

Example Description

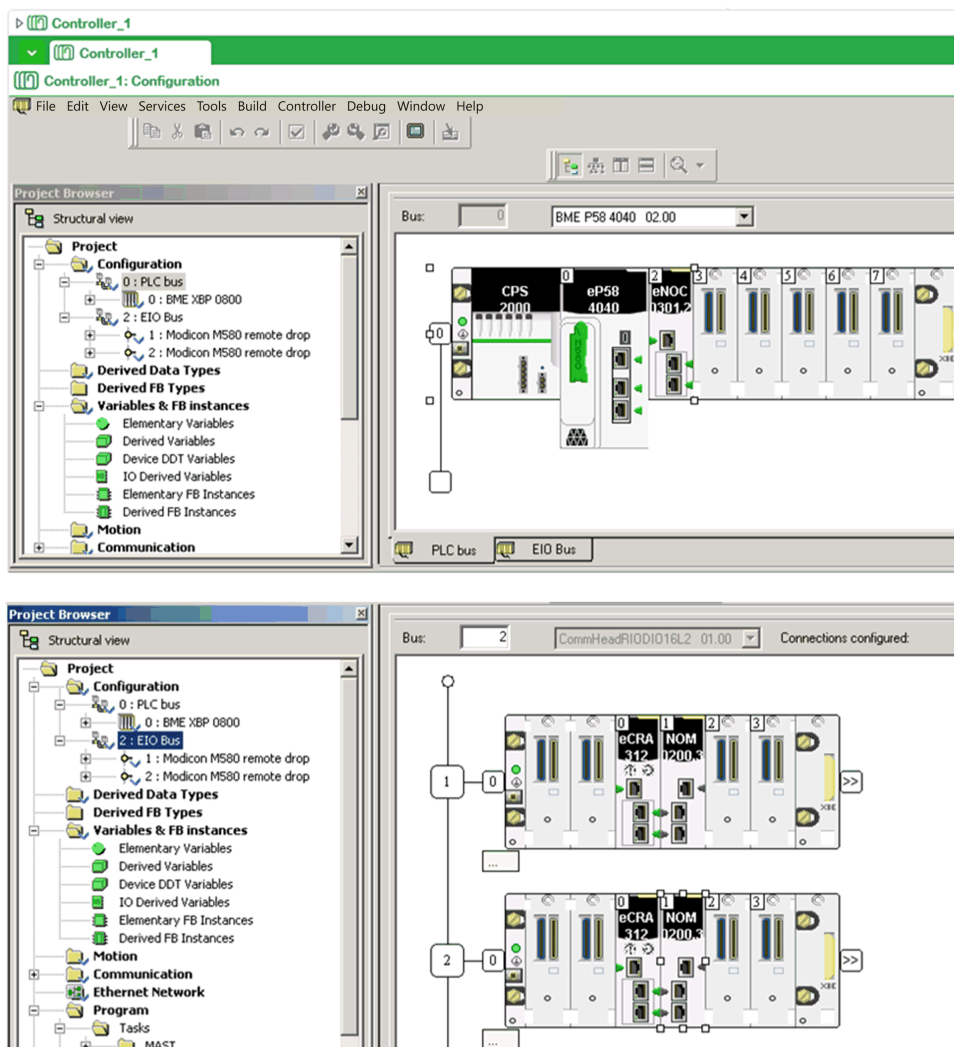
The table indicates the specifics of this example in terms of drop configuration and application objects used.

M580 remote drop configuration	Application objects
2 drops with 1 BMXNOM0200 communication module in each.	For each drop, 1 device of the EPE Managed Devices product range communicating through 1 communication port.



Configuration Stage

The following figure shows the controller configuration with the drop and the communication module.



Instantiation Stage

The following Instances are required to implement communication at the application level with two devices (a TeSys T motor controller and an Altivar variable speed drive) by using Modbus serial.

The *MBPortM58X80_x* Modbus serial communication port instance are to be used when the *BMXNOM...* communication module is located in the drop of an M580 controller. In this example, two instances are created, one for each physical communication port used.

Instances are grouped in two folders for clarity, one for each communication port instance.

D						
Identifier	Template	Version	Data	Link	Assigned State	
EthAddM_NOC	\$EthAddM	1.0.4	Valid	Valid	Unassigned	
MBPortM58X80_NOM1	SMBPortM58X80	1.0.7	Valid	Valid	Unassigned	
Motor_1	\$Motor	2.5.3	Valid	Valid	Unassigned	
TesysTMB_1	\$TesysTMB	2.1.6	Valid	Invalid	Unassigned	

D_2						
Identifier	Template	Version	Data	Link	Assigned State	
ATV61MB_2	\$ATV61MB	1.2.5	Valid	Invalid	Unassigned	
MBPortM58X80_NOM2	SMBPortM58X80	1.0.7	Valid	Valid	Unassigned	
Motor_2	\$Motor	2.5.3	Valid	Valid	Unassigned	

These steps result in the following links.

The communication port instance (*MBPortM58X80_NOM1*) passes the following data to the process instances:

- Port address of the BMENOC•••• communication module
- IP address of the BMECRA•••• module of the drop
- Port address of the BMXNOM•••• communication module

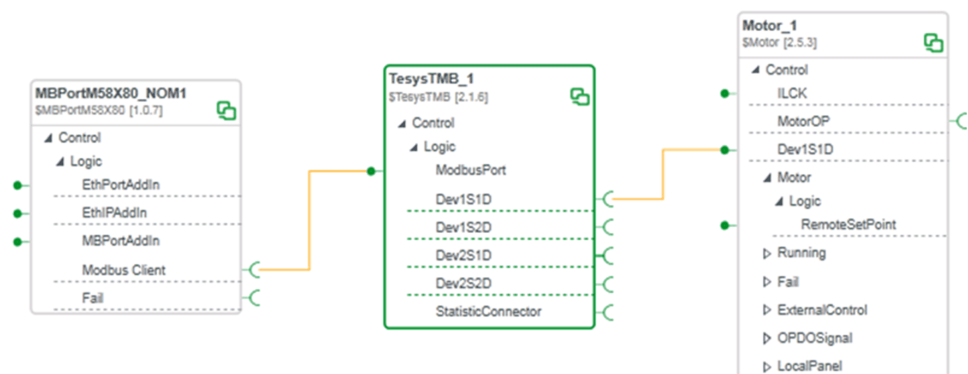
Intermediate Template Instances

Only intermediate template *\$EthAddM_NOC_CRA* is used because more than one communication port instance is used. It is common to both communication port instances. Intermediate template *\$MBAddM* is not used because on each BMXNOM•••• port, only one communication port instance is used.

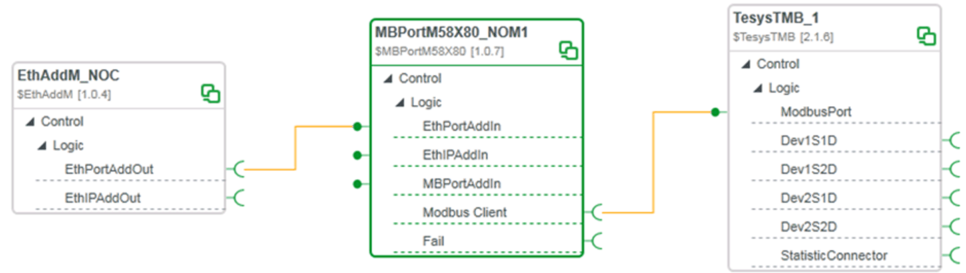
Application Interface Links

The following links are created in the **Asset Workspace Editor** for the TeSys T devices.

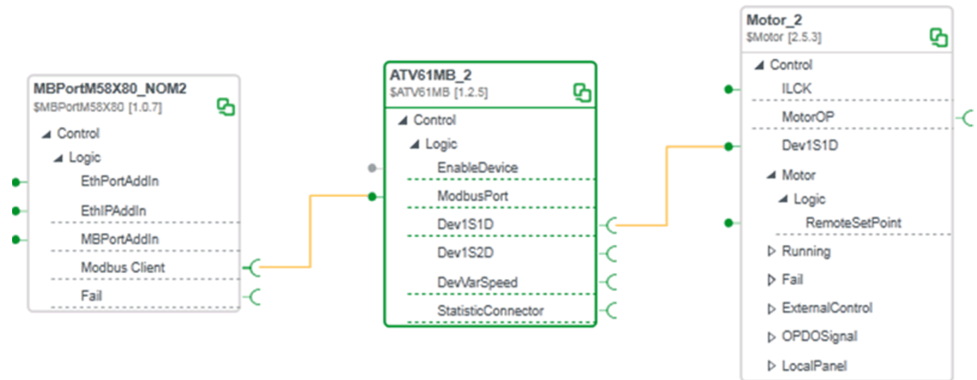
The process instance (*Motor_1*), the device instance (*TesysTMB_1*), and the communication port instance (*MBPortM58X80_NOM1*) are linked.



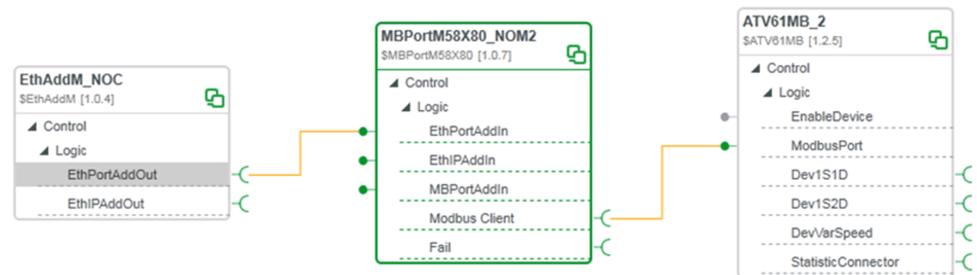
Then, link the *MBPortM58X80_NOM1* instance to the **EthPortAddOut** interface of the *EthAddM_NOC* intermediate template instance.



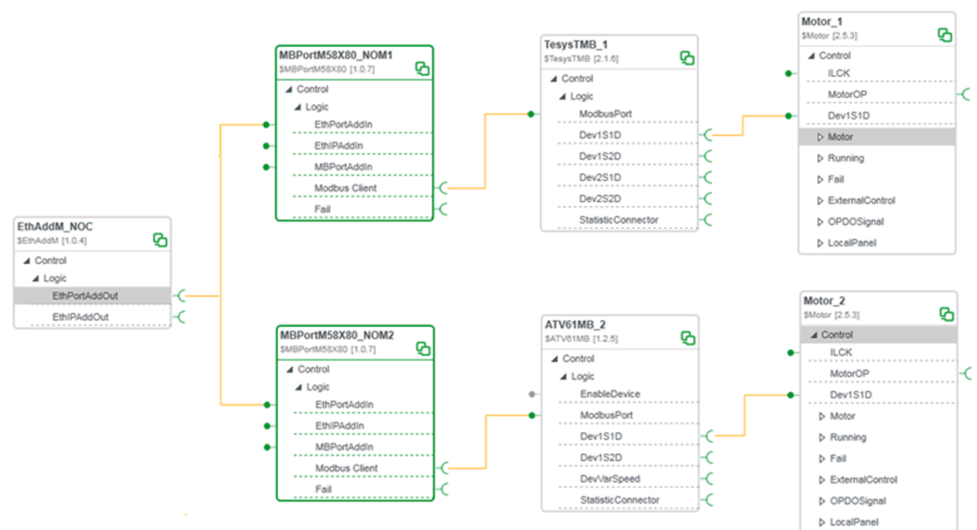
Similarly, the process instance (*Motor_2*), the device instance (*ATV61MB_2*), and the communication port instance (*MBPortM58X80_NOM2*) are linked.



Then, link the *MBPortM58X80_NOM2* instance to the **EthPortAddOut** interface of the *EthAddM_NOC* intermediate template instance.



These steps result in the following links.



The communication port instances (*MBPortM58X80_x*) pass the following data to the process instances:

- IP address of the BMCECRA.... module of each drop
- Port address of the BMXNOM.... communication module of each drop

The intermediate template instance (*EthAddM_NOC*) passes the following data to the process instances:

- Port address of the BMENOC**** communication module in the local rack

Control Facet Assignment

Facets of the instances are assigned to a Control Participant project in two separate FBD sections.

Folder *D* is assigned to *FBDSection_1*.

Identifier	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
FBDSection_1	FBDSection_1	TesysTMB_1	\$TesysTMB	Valid	TesysTMB_1_TesysT	\$TesysTMB_UL	Control	0	Assigned	Generated
FBDSection_1	FBDSection_1	MBPortM5EX80_NOM1	\$MBPortM5EX80	Valid	MBPortM5EX80_NOM1_MBPMS5EX80	\$MBPortM5EX80_UL	Control	1	Assigned	Generated
FBDSection_1	FBDSection_1	EthAddM_NOC	\$EthAddM	Valid	EthAddM_NOC_PA	\$EthAddM_UL	Control	2	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_DEVCTL	\$DEVCTL_UL	Control/Motor	3	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM	\$CONDSUM_UL	Control/Failures	4	Assigned	Generated
FBDSection_1	FBDSection_1	Motor_1	\$Motor	Valid	Motor_1_CONDSUM1	\$CONDSUM1_UL	Control/Interlocks	5	Assigned	Generated

Folder *D_2* is assigned to *FBDSection_2*.

Identifier	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation
FBDSection_2	FBDSection_2	ATV61MB_2	\$ATV61MB	Valid	ATV61MB_2_ATV	\$ATV61MB_UL	Control	0	Assigned	Generated
FBDSection_2	FBDSection_2	MBPortM5EX80_NOM2	\$MBPortM5EX80	Valid	MBPortM5EX80_NOM2_MBPMS5EX80	\$MBPortM5EX80_UL	Control	1	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_2	\$Motor	Valid	Motor_2_DEVCTL	\$DEVCTL_UL	Control/Motor	2	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_2	\$Motor	Valid	Motor_2_CONDSUM	\$CONDSUM_UL	Control/Failures	3	Assigned	Generated
FBDSection_2	FBDSection_2	Motor_2	\$Motor	Valid	Motor_2_CONDSUM1	\$CONDSUM1_UL	Control/Interlocks	4	Assigned	Generated

Generation and Service Mapping

The Control Participant project is generated.

An executable is created in the Control Participant and mapped to the M580 controller.

Hardware Mapping

The **Hardware Mapping Editor** shows the following interfaces.

(Lines have been drawn to show the mappings that you must do.)

HWInstance	HWTemplate	HWMappingInterface	HWInterfaceType	AppInstance	AppFacet	AppTemplate	AppMappingInter
PhLocal 0 D 0 R 0 BMPE5840	\$M580CPUeHCRP	MBTCPeH580 EMPortChM	\$MEthernetPortCHHO	ATV61MB_2	ATV61MB_2_ATV	\$ATV61MB	MBAddress
PhLocal 0 D 0 R 02 BMENOC0301 2	\$MNOCIOSClient	MBTCPeH EMPortChM	\$MEthernetPortCHHO	EthAddM_NOC	EthAddM_NOC_PA	\$EthAddM	EMPotCHM
EthRIO 1 D 0 R 0 BMCEA31210 2	\$CRAEh	CRAIPAddress	\$SEthernetAddressHO	EthAddM_NOC	EthAddM_NOC_PA	\$EthAddM	CRAIPAddress
EthRIO 1 D 0 R 01 BMXNOM0200 3	\$HWNOMMB	MB_M340_M580_Master_CHD MModbusPortCH	\$MModbusPortCHHO	MBPortM5EX80_NOM1	MBPortM5EX80_NOM1_MBPMS5EX80	\$MBPortM5EX80	CRAIPAddress
EthRIO 1 D 0 R 01 BMXNOM0200 3	\$HWNOMMB	MBControllerSlave_CH1 MModbusAddress	\$MModbusAddressHO	MBPortM5EX80_NOM1	MBPortM5EX80_NOM1_MBPMS5EX80	\$MBPortM5EX80	MBPortCHM
EthRIO 2 D 0 R 0 BMCEA31210 2	\$CRAEh	CRAIPAddress	\$SEthernetAddressHO	MBPortM5EX80_NOM2	MBPortM5EX80_NOM2_MBPMS5EX80	\$MBPortM5EX80	CRAIPAddress
EthRIO 2 D 0 R 01 BMXNOM0200 3	\$HWNOMMB	MB_M340_M580_Master_CHD MModbusPortCH	\$MModbusPortCHHO	MBPortM5EX80_NOM2	MBPortM5EX80_NOM2_MBPMS5EX80	\$MBPortM5EX80	MBPortCHM
EthRIO 2 D 0 R 01 BMXNOM0200 3	\$HWNOMMB	MBControllerSlave_CH1 MModbusAddress	\$MModbusAddressHO	TesysTMB_1	TesysTMB_1_TesysT	\$TesysTMB	MBAddress

NOTE: You do not need to map **MBAddress** interfaces (for example, of facet *TesysTMB_1_TesysT*).

Completing the System Engineering Life Cycle

Complete the creation of the system by performing the other steps of the system engineering life cycle.

Monitoring Modbus Data From Supervision

Monitoring Modbus Device Data From Supervision

Overview

This topic provides specific procedures to help you implement connectivity between Modbus TCP and Modbus serial devices of the **EPE Managed Devices** product range and the Supervision infrastructure that is defined in the system.

For Modbus TCP devices, you can use the OPC or OPC UA driver.

This allows, for example, monitoring additional Modbus device variables without adding load to the controller.

The implementation is possible for the following topological entities:

- Modbus TCP I/O devices.
- STB islands.
- PROFIBUS remote masters (PRMs).
- Third-party I/O devices modeled by using the generic topological device template.

It also allows connecting Modbus serial devices by using the generic Modbus TCP I/O device acting as gateway.

The implementation supplements the engineering effort to set up communication between the controller and Modbus devices and to operate these devices from the runtime. However, it does not use the application control module templates that come into play in the usual system engineering life cycle but requires creating your own application objects by using Supervision data facet templates, page 994.

By following the steps described in this topic, EcoStruxure Process Expert generates Supervision tags for variables to be monitored based on the application objects that you have created.

Implementing Monitoring of Modbus TCP Device Data

The table describes the specific steps to monitor data of Modbus TCP devices directly from the Supervision infrastructure. It is assumed that the required topological entities, Control and Supervision Participant projects, the application instances, assignments, and mappings to implement communication with Modbus TCP devices, page 935 already exist.

Step	Description	Stage
1	<p>Instantiate a Supervision data facet template for each variable of a Modbus TCP device that you want to monitor from Supervision. You can use the following templates:</p> <ul style="list-style-type: none"> • <i>\$VariableTag_CD</i> • <i>\$AdvancedAlarm_CD</i> • <i>\$TrendTag_CD</i> <p>Configure the identifiers of each instance so that the relationship with the corresponding Modbus TCP device can be seen.</p> <p>NOTE: Select Facets as filter in the Browser pane.</p>	Instantiation
2	<p>Configure instances of <i>\$VariableTag_CD</i> as follows:</p> <ul style="list-style-type: none"> • <i>Addr</i> parameter (<i>Configuration</i> category): <ul style="list-style-type: none"> ◦ For the OPC driver: Address of the variable by using the format <i>Alias!%MW<j></i>. For example, <i>ATV61IOD_IODEVICE_1!%MW3233</i>. ◦ For the OPC UA driver: Name of the variable in the symbol table. 	

Step	Description	Stage
	<ul style="list-style-type: none"> <i>Type</i> parameter (<i>Configuration</i> category): Data type of the variable. For example, UINT. <p>NOTE: Alternatively, for OPC, you can specify the address for the device by adding an appropriate entry in the <i>citect.ini</i> file. For details, refer to <i>OPC Access Path Parameters</i> in the <i>OPC Driver</i> topic of the <i>Driver Reference Help</i>, which you can open from the Windows Start menu or from within the Supervision Participant (Topology/I/O Devices).</p>	
3	If you are using trend tags, open the Asset Workspace Editor and link the trend tag instance to the variable tag instance so that the instance status becomes valid.	
4	Create additional tag containers, page 353; one for each Modbus TCP device that you want to monitor from Supervision and configure Protocol as either OPC or OPC UA.	Project Definition (Supervision)
5	Create a dedicated communication board, page 312 and configure Type as either OPC or OPC UA.	
6	Create a communication port for each Modbus TCP device.	
7	Create an I/O device, page 309 for each tag container and configure the following parameters: <ul style="list-style-type: none"> Address: <ul style="list-style-type: none"> For the OPC driver: Enter <i>Schneider-Aut.OFS</i>, which corresponds to the Prog ID of the OFS server. For the OPC UA driver: The address parameter is not used. TagContainers: Associate each I/O device to a separate tag container. Ports 	
8	Assign the Supervision facets related to each device to the corresponding tag container. For example, if you have created two Modbus TCP devices (<i>EATV71HW_1</i> and <i>ETesystHW_1</i>) and for each device, you have create one variable tag instance (<i>ATV71_Var1</i> and <i>TesysT_Var1</i>), then you must assign: <ul style="list-style-type: none"> <i>ATV71_Var1</i> to <i>TagContainer_ATV71</i> <i>TesysT_Var1</i> to <i>TagContainer_TesysT</i> 	Assignment
9	Generate the project.	Generation
10	In the Communication Mapping tab, page 769, perform the following additional mappings for each I/O device that you have created for a Modbus TCP device: <ul style="list-style-type: none"> Select the corresponding topological entity (the Modbus TCP device). Select its communication service (MBTCPServiceServer or device-specific service). Verify the Modbus Unit ID value for the device and modify it as needed (refer to the documentation the device). 	Mapping (communication mapping for Supervision executables)
11	When using the OPC UA driver, you must create a symbol table (.csv) where the variable name and data type correspond to the values that you have entered for the <i>Addr</i> and <i>Type</i> parameters (see step 2). For details, refer to <i>Modbus Device Address Space</i> in the help of EcoStruxure OPC UA Server Expert Configuration Tool.	EcoStruxure OPC UA Server Expert Configuration Tool
12	Associate the symbol table file to the device alias by using the Symbol table file browse button. Verify that Symbol table is selected under Preload settings .	

To complete the other steps to engineer the system, follow the system engineering life cycle described in this document.

Implementing Monitoring of Modbus Serial Device Data

The table describes the specific steps to monitor data of Modbus serial devices directly from the Supervision infrastructure by using the OPC driver. It is assumed

that the required topological entities, Control and Supervision Participant projects, the application instances, assignments, and mappings to implement communication with devices by using Modbus explicit messaging, page 952 already exist.

Step	Description	Stage
1	Create a Modbus TCP device entity by using the <i>\$EGenericDeviceHW</i> template, page 603. The entity acts as a gateway.	Configuration
2	Configure the <i>IPAddress</i> and <i>SubnetMask</i> parameters of the gateway entity. If you are already using an application template of the Modbus serial device (for example, an instance of <i>\$ATV71MB</i>) and have linked it to an instance of the <i>\$EGtwMB</i> gateway application template, enter the GatewayAdd parameter value of the gateway instance. Otherwise, enter the IP address of the gateway. NOTE: Enter values for the read/write length and address parameters only if you are using I/O scanning with the gateway entity.	
3	Connect the gateway entity to the same logical network to which the workstation that is mapped to the I/O server is connected.	
4	Instantiate a Supervision data facet template for each variable of a Modbus serial device that you want to monitor from Supervision. You can use the following templates: <ul style="list-style-type: none"> <i>\$VariableTag_CD</i> <i>\$AdvancedAlarm_CD</i> <i>\$TrendTag_CD</i> Configure the identifiers of each instance so that the relationship with the corresponding Modbus serial device can be seen. NOTE: Select Facets as filter in the Browser pane.	Instantiation
5	If you are using trend tags, open the Asset Workspace Editor and link the trend tag instance to the variable tag instance so that the instance status becomes valid.	
6	Configure instances of <i>\$VariableTag_CD</i> as follows: <ul style="list-style-type: none"> Addr parameter (<i>Configuration</i> category): Address of the variable by using the format <i>Alias!%MW<i></i>. For example, <i>ATV61IOD_IODEVICE_1!%MW3233</i>. Type parameter (<i>Configuration</i> category): Data type of the variable. For example, <i>UINT</i> for <i>ATV61IOD_IODEVICE_1%MW3233</i>. NOTE: Alternatively, you can specify the address for the device by adding an appropriate entry in the <i>citect.ini</i> file. For details, refer to <i>OPC Access Path Parameters</i> in the <i>OPC Driver</i> topic of the <i>Driver Reference Help</i> , which you can open from the Windows Start menu or from within the Supervision Participant (Topology\I/O Devices).	
7	Create additional tag containers, page 353; one for each Modbus serial device that you want to monitor from Supervision. Configure Protocol as <i>OPC</i> .	Project Definition (Supervision)
8	Create a dedicated communication board, page 312 and configure Type as <i>OPC</i> .	
9	Create a communication port for each Modbus TCP device.	
10	Create an I/O device, page 309 for each tag container and configure the following parameters: <ul style="list-style-type: none"> Address: Enter <i>Schneider-Aut.OFS</i>, which corresponds to the Prog ID of the OFS server. TagContainers: Associate each I/O device to a separate tag container. Ports 	
11	Assign the Supervision facets related to each device to the corresponding tag container. For example, if you want to monitor two Modbus serial devices and for each device, you have create one variable tag instance (<i>ATV71MB_Var1</i> and <i>TesysTMB_Var1</i>), then you must assign:	Assignment

Step	Description	Stage
	<ul style="list-style-type: none"> • <i>ATV71MB_Var1</i> to <i>TagContainer_ATV71MB</i> • <i>TesysT_Var1</i> to <i>TagContainer_TesysT</i> 	
12	Generate the project.	Generation
13	<p>In the Communication Mapping tab, page 769, perform the following additional mappings for the first I/O device that you have created for a Modbus serial device:</p> <ul style="list-style-type: none"> • Select the gateway as topological entity (instance of <i>\$EGenericDeviceHW</i>). • Select its communication service (MBTCPServiceServer). • Enter the Modbus Unit ID value for the device. If you have already configured an application template of the device (for example, an instance of <i>\$ATV71MB</i>), enter the ModbusAddress parameter value of the instance. Otherwise, enter a unique ID for the device. 	Mapping (communication mapping for Supervision executables)
14	Repeat step 13 for each additional Modbus serial device by selecting the same gateway entity but entering a new Modbus Unit ID value.	

To complete the other steps to implement the system, follow the system engineering life cycle described in this document.

Communication by Using a PROFIBUS Primary Module

Communication With Devices by Using a PROFIBUS Primary Module

Overview

The topology supports the PMPXM0100 PROFIBUS primary module. The configuration of the module is performed partially in the **Topology Explorer** and completed by using the external ProSoft Configurator for Modicon. Additional steps must be performed in various stages of the system engineering life cycle.

Prerequisites

- Read and understand the *PMPXM0100 ProSoft User Manual*.
- Install ProSoft Configurator for Modicon. On a computer on which the engineering client is installed, you can use the EcoStruxure Process Expert installer.

NOTE: To download the latest version of the tool and the document, visit www.se.com/en/download.

Workflow

The table describes the workflow to integrate a PMPXM0100 PROFIBUS primary module and secondary devices into a system by using a standalone or redundant M580 controller.

Step	Description	Stage
1	Add a PMPXM0100 PROFIBUS module to an M580 standalone or redundant controller. NOTE: For details, including steps that you must perform when using a redundant controller, refer to the topic describing the configuration of the PMPXM0100 PROFIBUS module. (see EcoStruxure™ Process Expert, Control Participant Services, User Guide)	Configuration
2	If you have templates of PROFIBUS secondary devices, create instances thereof and configure them as needed. NOTE: If you want to use device templates of a Schneider Electric Library (for example, <i>\$ATV71PB</i> or <i>\$TeSysTPB</i>), refer to the specific procedure describing how to use them.	Instantiation
3	Assign the facets of application objects to the Control project and generate the project.	Assignment and Generation
4	With the help of the ProSoft Configurator for Modicon, configure the PMPXM0100 module and secondary devices by following the steps described in the <i>PMPXM0100 ProSoft User Manual</i> . NOTE: In the Export EcoStruxure Control Expert Export Options dialog box select <i>Enhanced for I/O Scan Mode</i> .	External
5	Refine the Control project and create the device DDT variable, page 455 of the PMPXM0100 module. NOTE: If you are using a redundant controller, create both device DDT variables.	Refinement
6	Import the exported project file by following the steps described in the <i>PMPXM0100 ProSoft User Manual</i> .	Refinement
7	Analyze the Control project.	
8	Build and deploy the Control project to the controller.	Build and Deployment

Step	Description	Stage
9	With the help of the ProSoft Configurator for Modicon, download the primary and secondary device configuration to the module by following the steps described in the <i>PMEPXM0100 ProSoft User Manual</i> .	External
10	Refine the deployed Control project online and follow the steps described in the <i>PMEPXM0100 ProSoft User Manual</i> to set the operating mode of the PMEPXM0100 module.	Execution

To complete the other steps to engineer the system, follow the system engineering life cycle described in this document.

Managing, Modifying, and Creating Templates


What's in This Part

Global Templates Library	991
Global Templates Editors	1006
Managing Global Templates	1034

Overview

Creating, modifying, updating, replacing, or duplicating templates may affect the function of these templates and/or systems and these actions must be performed by qualified personnel.

Also refer also to the topic containing information about template design best practices and key aspects (see *EcoStruxure Process Expert, Global Templates, Reference Manual*).

 **WARNING**

LOSS OF CONTROL

- Ensure that templates generate correct addresses when their mapping interfaces are mapped by using the **Hardware Mapping Editor**.
- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

Global Templates Library

What's in This Chapter

Global Templates Explorer	992
Global Templates Folders	998
Global Templates	1000
Managing the Control Constituent Password	1002

Overview

This chapter describes the Global Templates Library, the tools that are available to manage it, and context menu commands.

Global Templates Explorer

Overview

The **Global Templates Explorer** shows the templates of the Global Templates Library, which regroups the Schneider Electric Libraries for EcoStruxure Process Expert that are installed (for example, the Foundation Library, the General Purpose Library) as well as templates that you create and/or import.

These templates are displayed in a folder structure and can be of the following types:

- Interface models
- Facet templates
- Composite templates

The **Global Templates Explorer** allows you to perform the following actions on Global Templates:

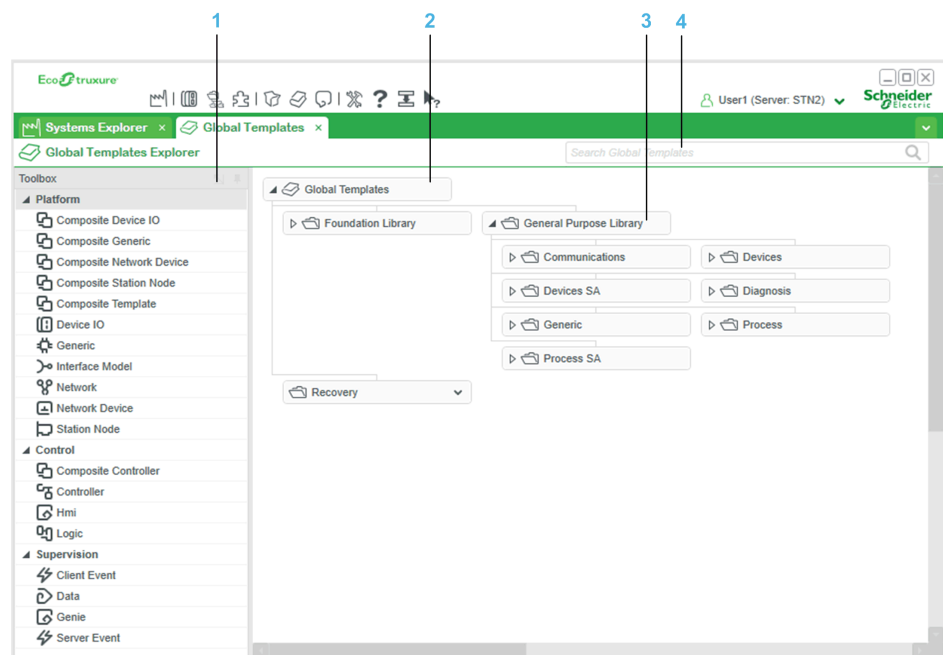
- Browse Global Templates.
- Export and import templates, page 1035.
- Open template editors, page 1006.
- Create copies of templates, page 1038.
- Create (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) your own folders, page 998 and templates.
- Update and replace templates, page 1051.

Opening the Global Templates Explorer

Open the **Global Templates Explorer** by clicking the **Global Templates** button in the toolbar, page 104.

Global Templates Explorer

The following figure shows an example of the **Global Templates Explorer** window. The content of the folder structure may vary depending on the Libraries that are installed.



Item	Description
1	Toolbox pane, page 994. The pin button allows you to minimize the Toolbox pane to the left edge of the screen. Move the pointer above the Toolbox tab to display the pane. Click the pin button again to restore the pane.
2	Global Templates Library root folder.
3	Tree view of the installed Libraries. You can modify the layout, page 996 of the tree view and open parts of it in new windows, page 996.
4	Search field, page 993.

Global Templates Explorer Actions

Right-clicking the white space of the **Global Templates Explorer** opens a context menu containing [commands, page 999](#) that apply to the Global Template Library except for the **Settings** command, which lets you [modify the folder layout, page 996](#) inside the workspace.

Global Templates Naming Convention

The Schneider Electric Global Templates follow a naming convention:

`$NAME_XX`

Where

- `$` indicates that the template is provided by Schneider Electric.
- `NAME` is the name of the template
- `_XX` indicates the type of the template. Control module templates have no suffix.

NOTE: Some templates provided by Schneider Electric may not have the `$` prefix to indicate that you can modify them to meet your specific requirements.

The table describes the suffixes that are used to identify types of Global Templates.

Suffix	Description
<code>_UL</code>	Control logic facet template.
<code>_UH</code>	Control HMI facet template.
<code>_UC</code>	Control logic and HMI composite template.
<code>_CD</code>	Supervision data composite and facet templates.
<code>_CG</code>	Supervision genie composite and facet templates.
<code>_CS</code>	Supervision composite template.
<code>_CR</code>	Supervision server composite template.
<code>_CC</code>	Supervision client composite template.

Searching Global Templates

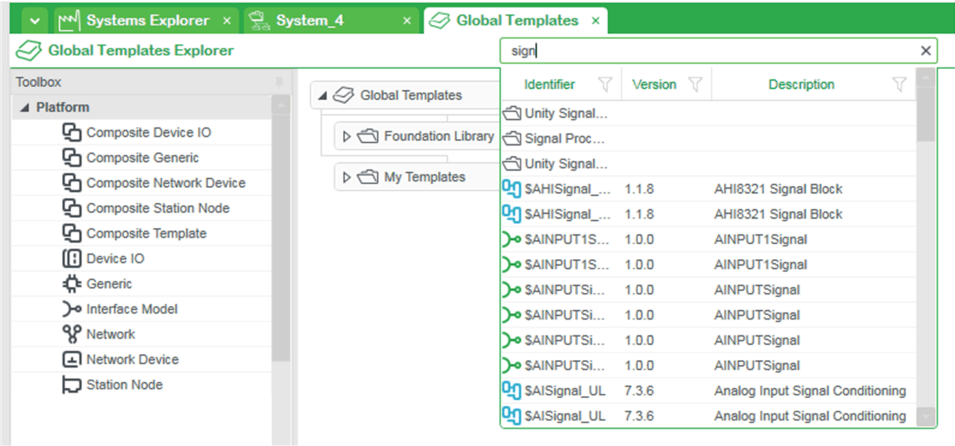
The search field of the **Global Templates Explorer** lets you find templates and folders by inputting all or part of their identifier. You do not need to use wildcards. The search is not case-sensitive. Results are displayed as you type in a pane with columns. Your last three matching search keywords are stored and appear in a keyword menu. The stored keyword entries persist even after the system server is restarted.

For example, `va` returns templates and folders that contain `va` in their identifier, such as templates `$Valve`, `$HValve_CD`, and folder `Advantys`.

By default, results are displayed in ascending alphabetical order with folders shown first, templates next.

The results pane provides basic information on templates and remains open until you close it by clicking the close button. A tooltip provides additional information.

The following figure shows an example of the search results that can be shown when you enter `sign` in the search field.



Right-click a search result in the results pane to open a context menu with the following commands.

Command	Description
Navigate	<ul style="list-style-type: none">For a template: Opens/expands the containing folder and highlights the instance.For a folder: Expands the folder structure and selects the folder. <p>You can also double-click the item.</p> <p>NOTE: If the template is hidden because a filter is applied, the template is not shown but the border of the folder containing the instance flashes briefly.</p>
Open	<p>The command is available for folders only.</p> <p>Opens a new Global Templates Explorer workspace and displays only the folder.</p> <p>NOTE: Searches that you perform by using the search field of this workspace are limited to the contents of the folder.</p>
Inspect	<p>These commands are available for templates only.</p> <p>For a description of these commands, refer to the topic describing Global Templates actions, page 1001.</p>
Read-Only	
Edit	

You can sort and filter, page 120 search results.

NOTE: Search results are not updated automatically if templates or folders were created, modified, or deleted in the meantime. You need to restart the search by clicking the search field and pressing **Enter**.

Global Templates Toolbox

The **Toolbox** contains the following base Global Templates, which you can use to create new templates or modify, page 1006 existing ones:

- Interface models
- Facet templates
- Composite templates

Templates are grouped in the following categories:

- **Platform:** Templates and interface models that are used at the platform level. These templates can reference components of this category and/or of the other two categories.
- **Control:** Facet templates that encapsulate constituents of the Control Participant. You can reference them inside composite templates.
- **Supervision:** Facet templates that encapsulate constituents of the Supervision Participant. You can reference them inside composite templates.

The table describes the items that are in the **Platform** category.

Item	Description
Composite Template	Generic template that you can use to: <ul style="list-style-type: none"> • Reference facet templates and other composite templates. • Build control module templates.
Composite Device IO and Device IO (facet templates)	Topological templates that you can use to model devices and their decentralized I/O modules in the topology of the system: <ul style="list-style-type: none"> • Devices (energy and motor devices, instrumentation...) • STB island I/O modules <p>NOTE: You can use this type of template to model third-party devices in the topology of a system if they will be connected to the network of such a system as nodes.</p>
Composite Generic and Generic (facet templates)	Topological templates that you can use to model unknown devices in a generic manner.
Interface Model	Object that you can define to create interfaces, which you can use to: <ul style="list-style-type: none"> • Link to each other facet and/or composite templates referenced by a control module template during template creation. • Expose connectors that allow linking instances to each other during instantiation.
Network	Topological template that you can use to model the following types of networks that are used in the topology of a system: <ul style="list-style-type: none"> • Ethernet • Modbus • CANopen • PROFIBUS DP • RIO S908 • RIO Ethernet • Local bus of topological entities
Composite Network Device and Network Device (facet templates)	Topological templates that you can use to model network devices that are used in the topology of a system (for example, switches, routers, gateways...).
Composite Station Node and Station Node (facet templates)	Topological templates that you can use to model a workstation in the topology of a system. <p>Such a workstation instance supports the following hardware and software subcomponents:</p> <ul style="list-style-type: none"> • Network interface card (NIC) • Supervision software • OPC Factory Server and OPC UA Server Expert software • Control Expert software • Platform and platform backup servers

The table describes the templates that are in the **Control** category.

Item	Description
Composite Controller and Controller (facet templates)	Topological templates that you can use to model controller hardware modules (CPUs, power supplies, network cards, I/O modules, and so on) of supported platforms (see EcoStruxure™ Process Expert, Control Participant Services, User Guide). NOTE: Do not use this type of template to model distributed I/O modules such as STB island I/O modules, for example in the topology of a system. Use a Platform DeviceIO template instead.
HMI	Application Control facet template that you can use to model one operator screen.
Logic	Application Control facet template that you can use to encapsulate Control constituents to implement a specific function. For example, you can encapsulate a DFB and one or more variables to manage on/off motor functions.

The table describes the templates that are in the **Supervision** category:

Item	Description
Client Event	Application client script facet template that you can use to create an event to be executed during runtime in the context of the Supervision client.
Data	Application Supervision facet template that you can use to model tags and user interface messages required for runtime data acquisition.
Genie	Application Supervision facet template that you can use to encapsulate an animated graphic (Genie), which is contained in an included project, page 510 and which can display data of the associated control module during runtime.
Server Event	Application server script facet template that you can use to create an event to be executed during runtime in the Supervision server.

Modifying the Tree View Layout

By using the **Settings** command, you can configure how many folders to display per row in the workspace of the **Global Templates Explorer**.

The setting applies to all the **Global Templates Explorer** windows that you open on the computer.

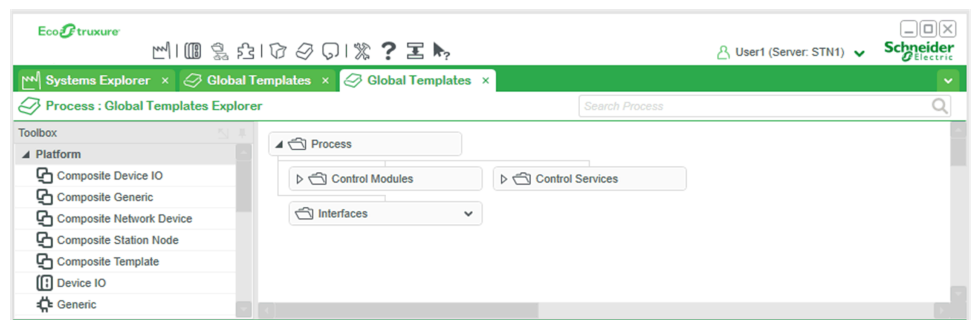
Opening Items in a New Window

You can open part of a folder structure that appears in the **Global Templates Explorer** in a new explorer window.

To open a folder structure in a new window, proceed as follows.

Step	Action
1	Drag the folder onto the tabs bar of the work area. Result: The folder opens in a new window as a parent folder. NOTE: If several work areas are open, you can drag the folder to the tabs bar of another work area.

The following figure shows an example where by expanding the sample **General Purpose Library** folder and dragging the **Process** folder to the tabs bar, a new explorer window opens in which the **Process** folder is the parent folder of the structure.



Recovery Folder

When you start the system server, if missing references are detected in templates, it creates the **Recovery** folder. You can see the folder in the **Global Templates Explorer**.

This folder contains the templates that are missing.

This can be the case, for example, after migrating the database of an earlier version. If the database contains a template that is used by another template (determined by using the **Used By** command) but you cannot locate this parent template by searching for it, most likely you can find it in the **Recovery** folder.

You can work with templates inside the **Recovery** folder like with any other templates by right-clicking them and selecting an action, page 1001.

Global Templates Folders

Creating Global Templates Folders

To create a Global Templates folder, proceed as follows.

Step	Action
1	Right-click the Global Templates root folder or an existing folder in the Global Templates Explorer and select Create Folder . Result: It creates a folder with the identifier in edit mode and displays it in the tree view.
2	To change the identifier, type it; otherwise, press Enter to accept the default name.

Global Templates Folder Actions

Right-click a Global Templates folder to open a context menu containing the following commands.

Command	Description
Template Creation Wizard	Opens the Template Creation Wizard , which lets you create a control module template (see EcoStruxure™ Process Expert, Global Templates Reference Manual) with a few clicks given you have completed the necessary preparation work.
Create Folder	Creates a new folder and displays it in the tree view.
Create	Opens a submenu that allows you to create in the folder any base composite and facet template, which is available in the Toolbox .
Control Constituent Password	Opens a submenu containing the following commands. The menu is only available if the Control constituent application password protection setting, page 1002 for Control facet templates is enabled. NOTE: Using either command starts an operation that may take time and during this time, you cannot use EcoStruxure Process Expert.
	Manage Password Opens the Manage Password dialog box, which lets you set and change, page 1004 the Control constituent password.
	Clear Password Opens the Clear Password dialog box, which lets you remove, page 1004 the Control constituent password.
	Forgot Password Opens the Forgot Password dialog box, which contains an authentication code that you need to reset, page 1005 the forgotten password.
Copy , page 1038	Copies the folder, any subfolders and their contents.
Paste , page 1038	Pastes the items that you have copied last to a new location.
Export , page 1039	Opens the Export dialog box, which allows you to select which templates contained in the folder, including any subfolders you want to export to a file in .sbk format.
Import , page 1042	Opens the Import dialog box, which allows you to select an export file in .sbk format and displays the compatible contents that you can select for import.
Purge , page 1049	Opens the Purge dialog box, which allows you to select templates to purge in the folder and any subfolder.
Delete	Deletes the folder and its contents from the database after you confirm the command. The command is not available if the folder or any subfolder contains a template.
Update	Opens the Update Templates in Folder window, page 1058, which allows you to create a new version of templates contained in the folder and, optionally, in its subfolders. It lists the templates for which at least one referenced child template with a later version exists in the Global Templates Library. You can select the templates that you want to update and define various update settings. The command is not available from the Global Templates Library root folder.
Rename	Lets you enter a new name for the folder. The command is not available if the folder or any subfolder contains a template.
Settings	Opens the Global Settings window, which lets you configure parameters, page 1002 that impact template creation and modification. The command is only available from the Global Templates Library root folder.

Global Templates

Global Templates Description

The following fields are displayed in a Global Templates folder in the tree view of the **Global Templates Explorer**.

Field	Description
Identifier	The fields come from the header of the composite/facet template or interface. Refer to the following topics for a description of the fields for: <ul style="list-style-type: none">• Composite templates (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>)• Facet templates (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>)• Interface models (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>)
Version	
Type	
Subtype	
Valid	
State	
Description	

Global Templates Actions

Right-click a template in the **Global Templates Explorer** to open a context menu with the following commands.

Command	Description
Show Changes Log	opens the Changes Log pane, page 1028, which lets you view the history of changes that were made to the template and saved.
View	Opens the template in the corresponding editor in view-only mode. The editor allows you to view the composition of the template or interface. NOTE: A password may be required to view a template.
Edit , page 1007	Opens the template in the corresponding editor in edit mode. The editor allows you to view and modify the definition and composition of the template or interface. NOTE: A password may be required to edit a template.
State	Opens a submenu that allows you to set the usability state (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) of the template to either Obsolete or Deprecated . The command is available only if: <ul style="list-style-type: none"> The template is used by an instance or referenced by another template. A higher version of the template exists. The usability state of the next highest version of the template is Approved.
Inspect	Allows you to open the Used By , Dependencies Tree , or External References panes, page 1028 in which you can view respectively: <ul style="list-style-type: none"> The templates that reference it. Templates that it references. For each reference you can view its interfaces. Systems in which an instance of the template exists.
Locations	Opens a submenu that allows you view where the template is located. If the template is present at more than one location: <ul style="list-style-type: none"> The present location is grayed out. You can click the other locations to open the folder and highlight the template.
Copy , page 1038	Copies the template for pasting.
Export , page 1039	Opens the Export dialog box, which allows you to save the template data to an export file in <i>.sbk</i> format.
Remove	Removes the copy of the template. The command is available only if copies, page 1038 of the template exist. When only one instance of the template remains, the command is replaced by Delete .
Delete	Deletes the template from the database after you confirm the command. The command is not available if either condition is satisfied: <ul style="list-style-type: none"> The template is referenced by another template, which is not being deleted. The template is used by a facet of in a system.
Update , page 1054	Opens the Update dialog box, which allows you to create a new version of the template by using the latest version of any referenced child templates that exist in the Global Templates Library.
Duplicate , page 1065	Opens the Duplicate window, which allows you to create, in a single step, a copy or new version of the template with the possibility to keep, duplicate, or replace any child template that is references.

Global Template Settings

Section	Description
Template Editors	<p>When the Defer New Parameters And Interfaces Automatically setting is enabled (default), the new parameters and interfaces of a template and its elements (referenced templates) are automatically deferred (see EcoStruxure™ Process Expert, Global Templates Reference Manual) to the next higher level.</p> <p>This applies to the following operations on templates:</p> <ul style="list-style-type: none"> Referencing a template that has parameters and/or interfaces (some of which can be deferred from its own child elements). Updating a template or an element of a template with changes that include adding a new parameter or interface. Replacing an element with a template that has parameters and/or interfaces. <p>Existing parameters and interfaces are not deferred.</p> <p>You can open the Global Settings window from the Global Templates Library root folder context menu to disable the functionality.</p>
Control Facet Templates	<p>The Control constituent application password protection setting lets you enable the Control Constituent Password context menu of the Global Templates root folder, page 999, which lets you set and manage a password, page 1002 for the Control facet templates of the Global Templates Library.</p> <p>The default value is inherited from the system server Control application and facet template password protection setting (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>), which is configured in the Control Passwords section of the System Server Configuration Wizard. When the server-level setting is enabled, the Control constituent application password protection setting is enabled and read-only.</p> <p>Possible values:</p> <ul style="list-style-type: none"> Enabled: You are required to set a password for Control facet templates in the Manage Password dialog box after you enable the setting and click OK in the Settings window. Disabled (default): You cannot set a password for Control facet templates. <p>To disable the setting, you must enter the password in the Manage Password dialog box, which also clears it.</p> <p>To change the setting, no other tab than the Global Templates Explorer must be open in the engineering client and no other clients must be connected to the system server.</p> <p>NOTE: Changing the setting may take time, page 1002 and during this time, you cannot use EcoStruxure Process Expert.</p>

Managing the Control Constituent Password

Overview

The **Control Constituent Password** that you manage at the Global Templates level by using the folder context menu has the following purpose:

- It is used as application password in the Control Participant projects (.STU), which encapsulate Control constituents in a Control facet template.
- It enables file encryption for these Control projects.

Setting, changing, or clearing the password takes about 45 min. to complete with the Schneider Electric Global Templates Library. More time is required if you have installed additional libraries and/or templates. During this time, you cannot use EcoStruxure Process Expert.

NOTE: For an overview of the entire Control project password management functionality, refer to the topic describing the system and Control facet template passwords (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

Prerequisites

The following are prerequisites to setting and managing the password:

- The **Control constituent application password protection** setting for Global Templates is enabled.
- No other tab than the **Global Template Explorer** is open in the engineering client.
- No other client is connected to the system server.

Scope of the Control Constituent Password Protection

You are required to enter the password that is set for Control facet templates to perform the following actions:

- Click **Templatizer** in the **Facet Editor** when editing a Control facet template.
- Open the Global Templates creation wizard.
- Export and import templates (includes operations involving template export/import, such as system backup and restore).

Further, the application password and file encryption is applied to constituents that you encapsulate in new Control facet templates (*Logic*) that you create.

NOTE: After you have entered password once, you are not required to enter it again until close the engineering client or restart the system server.

Password Management When Exporting and Importing Templates

When you export a template from the Global Templates Library, its application password is removed and file encryption is disabled.

When you import a template, if the **Control constituent application password protection** setting of the Global Templates Library in which you import is enabled and the template supports the functionality, an application password is set to its Control constituents and file encryption is enabled.

Information about Control facet templates to which the application password and file encryption could not be applied is shown in the system server console and in the notification panel of engineering clients.

Contact Schneider Electric support if you are faced with this situation and want password protection and file encryption to be applied.

NOTE: Exporting and importing templates when a Control constituent password is set takes time because of the process to remove and set the password.

Control Constituent Password Requirements

To be valid, the Control constituent password must contain the following:

- Between 8 and 16 characters.
- At least one uppercase letter from the classical Latin alphabet (A...Z).
- At least one lowercase letter from the classical Latin alphabet (a...z).
- At least one base-10 digit (0...9).
- At least one special character (~, !, @, \$, %, ^, &, *, _ , +, -, =, ` , |, \, (,), [,], ; , “ , ‘ , < , > , { , } , , , #).

Changing the Control Constituent Password

To change the Control constituent password of Control facet templates, proceed as follows.

Step	Action
1	In the Global Template Explorer , right-click the Global Templates folder (root) and click Control Constituent Password > Manage Password . Result: The Manage Password dialog box opens.
2	Enter the password in the Current Password text box.
3	Enter the new password and confirm the new password in the corresponding text boxes.
4	Click OK . Result: The new password is applied to the Control facet templates of the Global Templates Library. During this operation, you cannot use EcoStruxure Process Expert.

Clearing the Control Constituent Password

Clearing the Control constituent password of facet templates has the following impact:

- Clears the application password of Control Participant projects (.STU), which encapsulate Control logic in Control facet templates of the Global Templates Library.
- Disables file encryption for these Control Participant projects.

NOTE: You cannot clear the password when the **Control application and facet template password protection** setting (see EcoStruxure Process Expert, Installation and Configuration Guide) is enabled at the system server level.

To clear the Control constituent password of Control facet templates, proceed as follows.

Step	Action
1	In the Global Template Explorer , right-click the Global Templates folder (root), click Control Constituent Password > Clear Password and confirm the command. Result: The Clear Password dialog box opens.
2	Enter the password in the Current Password text box.
3	Click OK and confirm the command. Result: The password is cleared for each Control facet template of the Global Templates Library. During this operation, you cannot use EcoStruxure Process Expert.

Resetting a Forgotten Control Constituent Password

To reset the Control constituent password of Control facet templates, proceed as follows.

Step	Action
1	In the Global Template Explorer , right-click the Global Templates folder (root) and click Control Constituent Password > Forgot Password . Result: The Forgot Password dialog box opens and an authentication code is displayed.
2	Copy the authentication code, contact Schneider Electric support and provide the code. Result: You receive a temporary password.
3	Right-click the Global Templates folder and click Control Constituent Password > Manage Password . Result: The Manage Password dialog box opens.
4	Follow the procedure to change the password , page 1004 and enter the temporary password in the Current Password text box.

Global Templates Editors

What's in This Chapter

Opening Global Templates Editors	1007
Interface Editor.....	1008
Facet and Composite Template Editors	1013
Common Template Editor Components	1021

Overview

This chapter describes the Global Templates editors, which allow you to view and edit the entire definition of Global Templates. They also allow you to create new templates based on existing ones or by starting from empty base templates.

You can view and edit:

- Composite templates by using the **Composite Editor**.
- Facet templates by using the **Facet Editor**.
- Interface models and interfaces by using the **Interface Editor**.

Starting from the control module template level, you can drill down through the entire composition of the template. At each level, you can view dependencies and which templates reference the one you are viewing or editing.

NOTE: Some of the commands that are described in this chapter may not be available when you open a template editor in read-only mode.

Opening Global Templates Editors

Opening Global Templates Editors

Overview

Modifying Schneider Electric Global Templates or templates created by users may affect the function of these templates and must be performed by qualified personnel. Before proceeding, refer to the [Overview of this part, page 990](#).

Opening Global Templates Editors

To open a Global Templates editor, you need to open a template. You cannot open a Global Templates editor without opening a template either in read-only or editing mode.

Step	Action
1	In the tree view of the Global Templates Explorer , open the folder that contains the template that you want to view or edit.
2	<p>Right-click the template and select:</p> <ul style="list-style-type: none">• Read-only to open the template in the corresponding editor in read-only mode. This mode does not allow you to change the definition of the template. The background of the workspace has a light shade of gray to distinguish it from the editing mode.• Edit to open the template in the corresponding editor in editing mode. This mode allows you to modify the entire definition of the template. <p>NOTE: Double-clicking the template opens the corresponding editor in read-only mode by default.</p>

NOTE: From the **Global Templates explorer**, you can also look up templates by entering a key word in the [search field, page 992](#). In the list of results, right-click the template and select **Read-only** or **Edit**. The selected template opens in the corresponding editor in either mode.

Working Copy of Templates

When you open a template in read-only or editing mode, the software creates a temporary working copy of the template with the same identifier and a specific version (see *EcoStruxure Process Expert, Global Templates, Reference Manual*). It removes this working copy when you close the template editor.

Interface Editor

Interface Editor

Overview

The graphical **Interface Editor** allows you to manage the entire definition of interface models and interfaces.

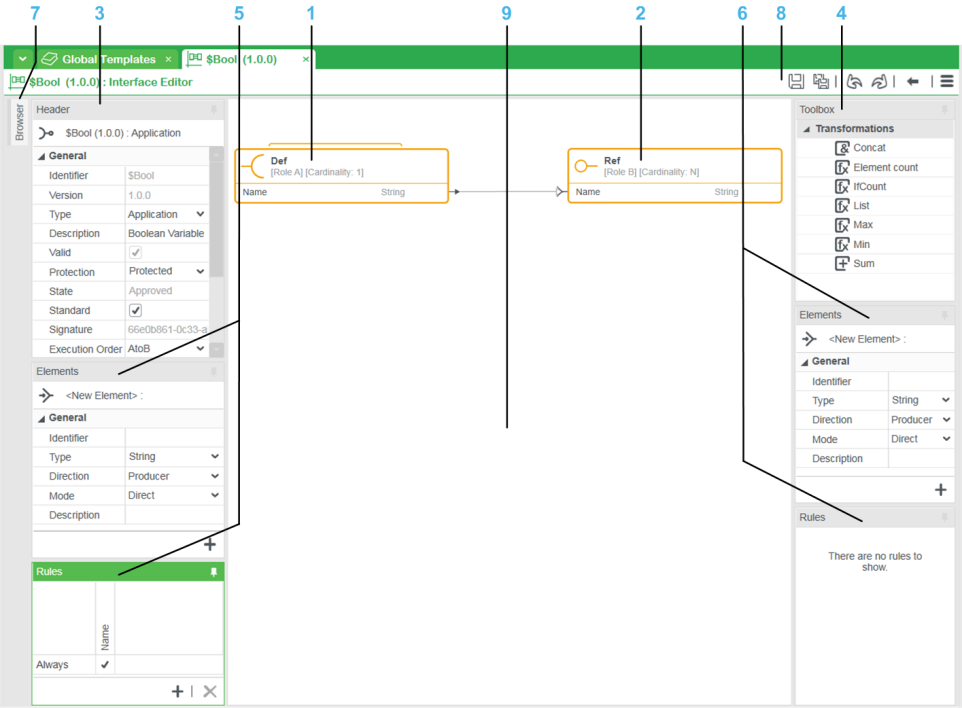
This topic describes the panes, toolbars, and menus of the **Interface Editor**.

Certain menus and commands may be available only in editing mode.

NOTE: This topic does not contain information on the definition of interface models or interfaces (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).

Interface Editor

The following figure shows an example of the **Interface Editor** in edit mode.



Item	Description
1	Graphical representation of role A of the interface
2	Graphical representation of role B of the interface
3	Interface Header pane
4	Toolbox pane
5	Elements and Rules panes for role A of the interface
6	Elements and Rules panes for role B of the interface
7	Button to show the interface Browser pane
8	Editor toolbar containing commands that apply to the interface being viewed or edited
9	Workspace. You can use the mouse to move the view, page 108.

Editor Panes

The **Interface Editor** uses various panes to group properties and data that are related. The table describes the purpose of the panes.

Pane	Description
Header	Allows you to view and/or edit general properties of the interface.
Toolbox	Allows you to browse and select transformation functions (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) to be used in the template.
Elements	Allows you to view, create, and modify elements of a role of the interface.
Rules	Allows you to view, create, and modify rules for the elements of a role of the interface.
Browser	Allows you to search interface models and interfaces that are available in the Global Templates Library. Move the pointer over the Browser tab to display the pane. Click the pin button to toggle between minimized and visible pane view. NOTE: The pane is available only in editing mode.
Used By⁽¹⁾	Allows you to view the templates and interfaces that reference the interface, which is open in the editor. You can right-click a template or interface in the Used By pane to open it in a new editor window. You can also view its dependencies, page 1009 or which other templates reference it.
Locations⁽¹⁾	Allows you to view the path to the interface and its copies, page 1038. Right-click an entry and select Open Reference Container to open the corresponding folder in the Global Templates explorer.
Changes Log⁽¹⁾	Keeps track of changes that are made to the interface and saved. The pane indicates: <ul style="list-style-type: none"> • The action that was performed. • The state of the interface after the change. • The mandatory description of changes that is entered when the interface is saved. • The version of the interface after the change. • The user name of the user logged in to the engineering client.
(1) Use the corresponding command in the editor toolbar menu to open the pane.	

NOTE: For information on working with panes, refer to the topic describing the engineering client workspace, page 113.

Additional Information

The list indicates where you can find a detailed description of the contents of the **Interface Editor** panes:

- Header definition (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) and header common definition (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) (shared among interface models, facet, and composite templates).
- Elements definition (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).
- Rules definition editors (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).
- Interface toolbox (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).

Editor Toolbar

The figure shows the toolbar that is located in the top right corner of the **Interface Editor**.



Item	Description
1	Edit button. Switches the editor from read-only to editing mode.
2	Save button. Opens the Save dialog box. For more information, refer to the topic describing how to save changes in templates, page 1035.
3	Save as button. Opens the Save as dialog box. For more information, refer to the topic describing how to save changes in templates, page 1035.
4	Undo and redo buttons. Lets you undo and redo most actions.
5	Shows the parent from which you have opened the interface and highlights it. If you had opened an interface that is an element of a template, only the template is highlighted. The parent template must be open for the command to be enabled. The command is not enabled when the interface is opened from the Global Templates Explorer .
6	Opens the editor toolbar menu that contains editor-specific commands.

Editor Menu

The table describes the commands that are available in the editor toolbar menu of the **Interface Editor**.

Command		Description
New...		Opens the Global Templates dialog box, which contains the same base templates as the Toolbox of the Global Templates explorer. It allows you to create a new template in one or more existing folders of the Global Templates Library and open it in edit mode in the corresponding editor. For more information, refer to the topic describing how to create templates (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).
View	Show Changes Log	The commands allow you to bring the corresponding pane to the front.
	Used By	If the pane is closed, the command opens it.
	Locations	For a description of the panes, refer to the topic describing editor panes, page 1009.
Save		Save button. Opens the Save dialog box. For more information, refer to the topic describing how to save changes in templates, page 1035.

Command		Description
Save As		<p>Save as button.</p> <p>Opens the Save As dialog box.</p> <p>For more information, refer to the topic describing how to save changes in templates, page 1035.</p>
Export	Standard Backup	<p>Opens the Export window, which allows you to export the interface definition to file (.sbk).</p> <p>You can only export an interface if you have saved changes.</p> <p>For more information on export feature, refer to the topic describing template export, page 1039.</p>
	Various image file formats	<p>Each command opens a Save dialog box, which allows you to create an image file with the corresponding file extension. The image captures the contents of the workspace as if it were shown with the Fit to content display ratio.</p> <p>Panes and toolbars are not captured.</p>
Close		<p>Closes the template editor.</p> <p>If you have made changes, opens the Save dialog box.</p> <p>For more information, refer to the topic describing how to save changes in templates, page 1035.</p>

Workspace Actions

Right-click an empty area of the workspace to open a context menu with the following commands.

Command	Description
Show changes log	Opens the Changes log pane.

Interface and Interface Element Actions

Right-click the header of an interface role in the workspace to open a context menu with the following command.

Command	Description
Rename	<p>Allows you to modify the identifier of the role.</p> <p>For more information, refer to the topic describing interface roles (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).</p>

Right-click an element of an interface role to open a context menu with the following commands. Commands may vary depending on the role of the element.

Command	Description
Switch to Optional	<p>Sets the rule for the element, which is the producer to <i>optional</i>. In this case, the element is not required to provide data for the interface link to be valid.</p> <p>The default value is <i>required</i>.</p> <p>NOTE: If you set a role to optional and later you change the role to consumer, the role is reset to <i>required</i>.</p>
Switch to Required	Sets the rule for the element, which is the producer to <i>required</i> (default value). In this case, the element is required to provide data for the interface link to be valid.
Use as Consumer	<p>Switches the direction from <i>producer</i> to <i>consumer</i>.</p> <p>NOTE: Using this command resets the element rule to <i>required</i> if it was set to <i>optional</i>.</p>
Use as Producer	Switches the direction from <i>consumer</i> to <i>producer</i> .

Command	Description
Switch Mode	<p>Switches the mode of the link between role A and role B of an element from <i>Direct</i> to <i>Transform</i> and the other way around.</p> <p>Right-clicking an element and switching the mode to:</p> <ul style="list-style-type: none"> • <i>Transform</i> <ul style="list-style-type: none"> ◦ Removes the direct link between role A and role B of the element. ◦ Removes the element from the other role. ◦ Requires that you create a new element in the other role with mode <i>Transform</i> and with the opposite direction. • <i>Direct</i> <ul style="list-style-type: none"> ◦ Removes the link between the element and the transformation function. ◦ Creates the element in the other role of the interface. ◦ Creates a direct link between both roles of the element. ◦ Requires that you reconfigure the transformation function, and possibly the elements that are still linked to it. <p>NOTE: When you switch the mode to <i>Transform</i>, verify that the cardinality that is defined in the Header pane for either role A or role B makes it possible to use transformation functions.</p> <p>NOTE: The command is not available for nested interfaces.</p>
Delete	Deletes the element from both roles of the interface.

Right-click a nested interface of an interface role to open a context menu with the following commands. Commands may vary depending on the role of the element.

Command	Description
Switch to Optional	<p>Sets the rule for the element to <i>optional</i>. In this case, the element is not required to provide data for the interface link to be valid.</p> <p>The default value is <i>required</i>.</p> <p>NOTE: The command is available by default for the <i>Def</i> role of the nested interface. To make the command available for the <i>Ref</i> role, add the element to the Rules table first by using the Add to Rules table command first.</p>
Switch to Required	Sets the rule for the element, which is the producer to <i>required</i> (default value). In this case, the element is required to provide data for the interface link to be valid.
Add to Rules table	Adds the element that has the <i>Ref</i> role to the Rules table so that you can define a rule for it (<i>optional</i> or <i>required</i>).
Remove from Rules table	<p>Removes the element that has the <i>Ref</i> role from the Rules table.</p> <p>NOTE: Removing the element from the Rules table makes the Switch to Optional and Switch to Required commands unavailable.</p>
Inspect	Opens a submenu with the Used By and Dependencies Tree , page 1009 commands.
Read-only	opens the nested interface in another Interface Editor in read-only mode.
Edit	opens the nested interface in another Interface Editor in editing mode.
Update	Allows you to update the version of the nested interface to the latest one, which exists in the Global Templates Library.
Replace	Opens the Replace dialog box. It allows you to replace the nested interface by a different one, which exists in the Global Templates Library.
Delete	Deletes the nested interface from both roles of the interface.

Saving Changes

Refer to Saving Changes in Global Templates, page 1035.

Facet and Composite Template Editors

Overview

This topic describes the user interface of the **Facet Editor** and **Composite Editor**.

Facet Editor

Overview

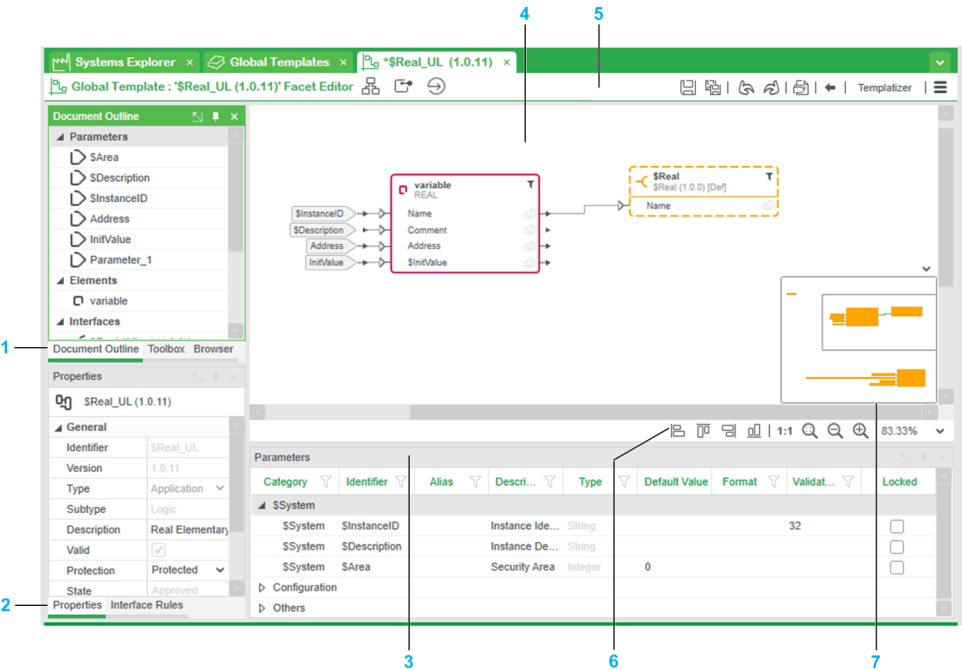
The graphical **Facet Editor** allows you to manage the entire definition of Control and Supervision facet templates.

For a description of panes, toolbars, and menus, refer to the topic describing common template editor components, page 1021.

NOTE: This topic does not contain information on the definition of facet templates (see *EcoStruxure Process Expert, Global Templates, Reference Manual*).

Facet Editor

The following figure shows an example of the **Facet Editor** in editing mode.



Item	Description
1	Document Outline, Toolbox, and Browser panes.
2	Properties and Interface Rules panes.
3	Parameters pane.
4	Workspace area displaying a graphical representation of the elements of the template that you are editing.
5	Editor toolbar, page 1022 containing commands that apply to the template being viewed or edited and/or its elements.
6	Workspace toolbar, page 1023.
7	Radar view, page 1031.

Workspace Actions

For a description of the context menu that opens when you right-click an empty area of the workspace, refer to the description of the workspace context menu, page 1026.

Element Actions

Right-click the header of an element in the workspace of the **Facet Editor** to open a context menu with the following commands.

Command	Description
Optimize Bindings	Reroutes bindings of the element correctly.
Show/Hide Unchecked	The commands have been replaced by a filter menu , page 1030 at the element level.
Show/Hide Unbound	
Go To	Opens a submenu that contains the name and path of other objects and/or their properties that are linked to the element through bindings so you can locate them (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>). When you select an entry, the object and the binding are highlighted in the workspace.
Copy	Copies the element for pasting in this template or another template.
Exclude	After you confirm the command, removes the element from the workspace and makes it available again in the Elements section of the Document Outline pane. Any existing bindings with the element are discarded. NOTE: The command is available only for Control elements.
Delete	After you confirm the command, removes the element from the facet template. Any existing bindings with the element are discarded. NOTE: The command is available only for certain types of elements.
Rename	Allows you to modify the identifier of the element.
Properties	Displays the properties of the element in the Properties pane.

Right-click the property of an element in the workspace of the **Facet Editor** to open a context menu with the following commands.

Command	Description
Create Parameter	Allows you to create a parameter that you can customize (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).
Create Input Value	Allows you to enter an absolute value.
Show/Hide \$IsConnected	Displays or hides the <i>\$IsConnected</i> property, which outputs the boolean value <i>TRUE</i> when the interface link is successfully established. This implies that applicable interface and interface element rules are satisfied; otherwise the value is <i>FALSE</i> . The command is available for interfaces only.
Go To	Opens a submenu that contains the name and path of other objects and/or their properties that are linked to this property through bindings so you can locate them (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>). When you select an entry, the object and the binding are highlighted in the workspace.
Binding To Here From...	Use this command (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) on the property that is the destination of the binding to be created. It enables the Bind to command.
Binding From Here To...	Use this command (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) on the object that is the source of the binding to be created. It enables the Bind from command.
Properties	Displays the properties of the element in the Properties pane.

Right-click the header of an interface element in the workspace of the **Facet Editor** to open a context menu with the following commands.

Command	Description
Switch to Optional/Required	<p>Optional: Makes the element optional in the context of the template.</p> <p>The element is represented with a dotted outline.</p> <p>Required: Makes the element mandatory in the context of the template.</p> <p>The element is represented with a solid outline.</p>
Optimize Bindings	Reroutes bindings of the element correctly.
Show/Hide Unchecked	The commands have been replaced by a filter menu, page 1030 at the element level.
Show/Hide Unbound	
Inspect	The menu entry has been removed. Use the left-hand toolbar buttons instead to open the Used By and Dependencies Tree panes, page 1028.
View	Opens the element in the corresponding editor in read-only mode.
Edit	Opens the element in the corresponding editor in edit mode.
Edit/Extend Interface	<p>Opens the Edit/Extend Interface window, page 1070, which lets you edit, add, and remove elementary elements (in direct mode only) in the interface without the need to edit the interface by using the Interface Editor.</p> <p>The command is not available for deferred interfaces and when the interface element contains a nested interface.</p> <p>Updating the templates that reference the other role of the interface is required.</p>
Go To	Opens a submenu that contains the name and path of other objects and/or their properties that are linked to the element through bindings so you can locate them (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>). When you select an entry, the object and the binding are highlighted in the workspace.
Copy	Copies the element for pasting in this template or another template.
Update , page 1054	Allows you to update the version of the interface model that is used by the interface to the latest one, which exists in the Global Templates Library.
Replace , page 1060	Opens the Replace dialog box. It allows you to replace the interface model that is used by the interface by a different one, which exists in the Global Templates Library.
Delete	Removes the element from the template.
Rename	Allows you to modify the identifier of the element.
Properties	Displays the properties of the element in the Properties pane.

Additional Information

The list indicates where you can find a detailed description of the contents of the various panes of the **Facet Editor**:

- Properties (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) and properties common definition (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) (shared among interface models, facet, and composite templates).
- Facet Elements (see *EcoStruxure Process Expert, Global Templates, Reference Manual*)
- Interfaces Rules (see *EcoStruxure Process Expert, Global Templates, Reference Manual*)
- Toolbox: binding functions (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) and/or Supervision Participant elements, page 1028.
- Parameter pane (see *EcoStruxure Process Expert, Global Templates, Reference Manual*)

Saving Changes

Refer to Saving Changes in Global Templates, page 1035.

Composite Editor

Overview

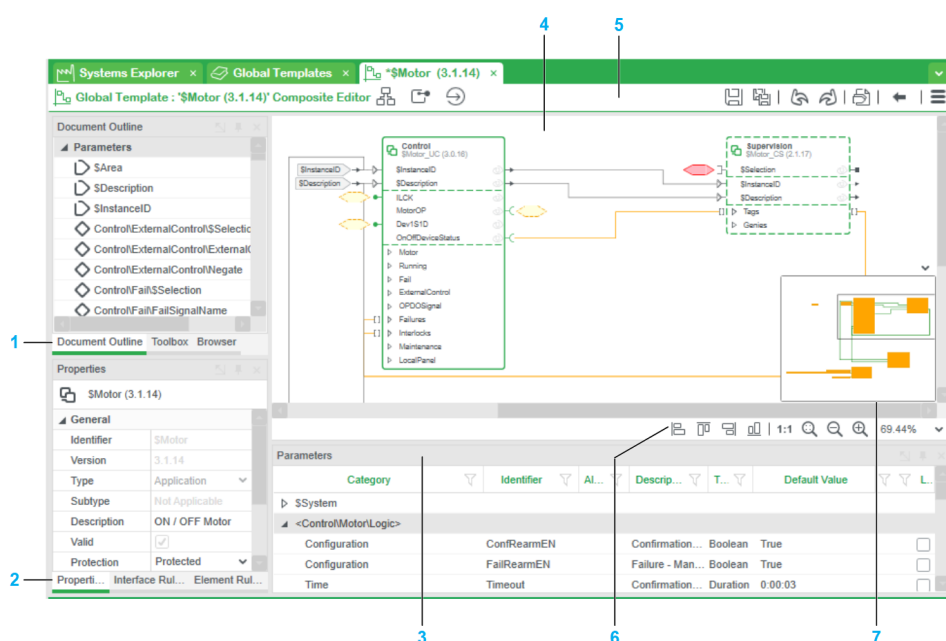
The graphical **Composite Editor** allows you to manage the entire composition and definition of composite templates.

For a description of panes, toolbars, and menus, refer to the topic describing common template editor components, page 1021.

NOTE: This topic does not contain information on the definition of composite templates (see *EcoStruxure Process Expert, Global Templates, Reference Manual*).

Composite Editor

The following figure shows an example of the **Composite Editor** in editing mode.



Item	Description
1	Document Outline, Toolbox, and Browser panes.
2	Properties, Interface Rules, and Element Rules panes.
3	Parameters pane.
4	Workspace area displaying a graphical representation of the elements of the template that you are editing.
5	Editor toolbar, page 1022 containing commands that apply to the template being viewed or edited.
6	Workspace toolbar, page 1023.
7	Radar view, page 1031.

Workspace Actions

For a description of the context menu that opens when you right-click an empty area of the workspace, refer to the description of the workspace context menu, page 1026.

Element Actions

Right-click the header of an element in the workspace of the **Composite Editor** to open a context menu with the following commands.

Command	Description
Switch to Optional/Required	Optional: makes the element optional in the context of the template. The element is represented with a dotted outline. Required: makes the element mandatory in the context of the template. The element is represented with a solid outline.
Defer	Defers items to make them available in the top-level template referencing the composite that you are editing (for example, in the control module). Selecting the command displays a submenu with the following commands: <ul style="list-style-type: none"> • Unbound selection/parameters: Makes available any element selection (only optional elements) and parameters for which no binding exists. • Unbound interfaces: Makes available any interfaces for which no binding exists. • All unbound: Makes available any element selection (only optional elements), parameters, and interfaces for which no binding exists.
Optimize Bindings	Reroutes bindings of the element correctly.
Show/Hide Unchecked	The commands have been replaced by a filter menu, page 1030 at the element level.
Show/Hide Unbound	
Inspect	The menu entry has been removed. Use the left-hand toolbar buttons instead to open the Used By , Dependencies Tree , and External References panes, page 1028.
View	Opens the element in the corresponding editor in read-only mode.
Edit	Opens the element in the corresponding editor in edit mode.
Edit/Extend Interface	For interfaces only. Opens the Edit/Extend Interface window, page 1070, which lets you edit, add, and remove elementary elements (in direct mode only) in the interface without the need to edit the interface by using the Interface Editor . The command is not available for deferred interfaces and when the interface element contains a nested interface. Updating the templates that reference the other role of the interface is required.
Go To	Opens a submenu that contains the name and path of other objects and/or their properties that are linked to the element through bindings so you can locate them (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>). When you select an entry, the object and the binding are highlighted in the workspace.
Copy	Copies the element for pasting in this template or another template.
Update , page 1054	Allows you to update the version of the template that is used by the element to the latest one, which exists in the Global Templates Library. In case of an interface element, lets you replace the interface model.
Replace , page 1060	Opens the Replace dialog box. It allows you to replace the template that is used by the element by a different one, which exists in the Global Templates Library. In case of an interface element, lets you replace the interface model.
Delete	Removes the element from the template.
Rename	Allows you to modify the identifier of the element.
Properties	Displays the properties of the element in the Properties pane.

Right-click the parameter of an element in the workspace of the **Composite Editor** to open a context menu with the following commands.

Command	Description
Create Deferred	Defers the parameter to make it available in the top-level template referencing the composite that you are editing.
The other commands are described in the table describing the context menu commands of properties of elements of facet templates, page 1015.	

Right-click the interface of an element in the workspace of the **Composite Editor** to open a context menu with the following commands.

Command	Description
Create Deferred	Defers the interface to make it available in the top-level template referencing the composite that you are editing.
Create Extended	Shows the interface as an element of the template.
Explode Implode	<ul style="list-style-type: none"> Collapsed state: Makes a connector available that represents the complete interface. The connection is made by using an interface link (shown in orange). Expanded state: Makes each element of a multi-element interface available individually for connection to other elements by using bindings (shown in gray).
Inspect	<p>The menu entry has been removed. Use the left-hand toolbar buttons instead to open the Used By and Dependencies Tree panes, page 1028.</p> <p>They allow you to view respectively:</p> <ul style="list-style-type: none"> The templates that reference it. Nested interfaces that it references.
View	Opens the interface in the corresponding editor in read-only mode.
Edit	Opens the interface in the corresponding editor in edit mode.
Go To	Opens a submenu that contains the name and path of other objects and/or their properties that are linked to the interface through bindings so you can locate them (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>). When you select an entry, the object and the binding are highlighted in the workspace.
Properties	Displays the properties of the interface in the Properties pane.

NOTE: Additional commands are described in the topic describing actions for properties of elements, page 1015 in the **Facet Editor**.

Additional Information

The list indicates where you can find a detailed description of the contents of the various panes of the **Composite Editor**:

- Properties (Header) (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) and properties common definition (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) (shared among interface models, facet, and composite templates).
- Composite Elements (see *EcoStruxure Process Expert, Global Templates, Reference Manual*)
- Interfaces Rules (see *EcoStruxure Process Expert, Global Templates, Reference Manual*)
- Toolbox: binding functions (see *EcoStruxure Process Expert, Global Templates, Reference Manual*)
- Parameter pane (see *EcoStruxure Process Expert, Global Templates, Reference Manual*)

Saving Changes

Refer to Saving Changes in Global Templates, page 1035.

Common Template Editor Components

Overview

The **Facet Editor** and **Composite Editor** have several components in common, which are described in this topic. Items that are specific to either editor are pointed out.

Common Template Editor Toolbars and Menus

Overview

The **Facet Editor** and **Composite Editor** feature toolbars and menus, which allow you to access the various tools and functions that you require to create and edit Global Templates.

This topic describes toolbars and menus that the **Facet Editor** and **Composite Editor** have in common. Those that are specific to either editor are pointed out.

Certain menus and commands may be available only in editing mode.

NOTE: Unless otherwise mentioned, toolbars contain commands that apply to the template that is being viewed or edited. To interact with an element of a template, open the context menu of the element by right-clicking it.

Editor Toolbars

The figure shows the toolbar that is located in the top left corner of the editors.



The buttons apply to the template that is being edited. However, if an element inside this template is selected, they apply to this element.

Item	Description
1	Displays the Dependencies Tree pane, page 1028.
2	Displays the Used By pane, page 1028.
3	Displays the External References pane, page 1028.

The figure shows the editor toolbar that is located in the top right corner of the editors.



Item	Description
1	Edit button. Switches the editor from read-only to editing mode.
2	Save button. Opens the Save dialog box. For more information, refer to the topic describing how to save changes in templates, page 1035.
3	Save as button. Opens the Save As dialog box. For more information, refer to the topic describing how to save changes in templates, page 1035.
4	Undo and redo buttons. Lets you undo and redo most actions.
5	Displays the Parameters pane, page 1028.
6	Shows the parent from which you have opened the template and highlights the template. If you had opened an element inside a template, only the template is highlighted. The parent template must be open for the command to be enabled. The command is disabled when the child is opened from the Global Templates Explorer .
7	Opens the Select Variables or Select Genie window, which allows you to encapsulate constituents in Control and compatible Supervision facet templates respectively. It also allows you to modify encapsulated elements with the help of the corresponding Participant. If a Control facet contains no constituents, the Content Not Found dialog box opens, which allows you to start the encapsulation process. For more information, refer to the topic describing the configuration of facet templates (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>). NOTE: The command is available only in the Facet Editor in edit mode.
8	Opens the editor menu, page 1024 that contains editor-specific commands.

Workspace Toolbar

The figure shows the workspace toolbar that is located in the bottom right-hand corner of the **Facet Editor** and **Composite Editor**.



Item	Description
1	Aligns the left edge of any selected item in the workspace with the left edge of the item that you have selected first. NOTE: The graphical alignment rules do not apply to parameters and input values, except for the three <i>\$System</i> parameters <i>\$InstanceID</i> , <i>\$Description</i> , and <i>\$Area</i> .
2	Aligns the top edge of any selected item in the workspace with the top edge of the item that you have selected first.

Item	Description
3	Aligns the right edge of any selected item in the workspace with the right edge of the item that you have selected first.
4	Aligns the bottom edge of any selected item in the workspace with the bottom edge of the item that you have selected first.
5	Reverts to the default zoom level (100%).
6	Adjusts the zoom level to fit the elements of the template inside the visible workspace. If the layout of the template is very large, some objects may not fit inside the visible workspace. You can use the radar view , page 1031 to locate these objects.
7	Zooms out, page 109 . Alternatively, use the minus (-) keyboard key. When you zoom out to 50%, the display switches to the simplified view , page 1032 . To return to the normal view, zoom in.
8	Zooms in. Alternatively, use the plus (+) keyboard key.
9	Zoom level. Use a predefined level by clicking the arrow and selecting a value from the list. You can set a custom level by double-clicking the value, entering a new one, and pressing Enter . Range: 50 to 200.

NOTE: For information on commands that allow you to select multiple items in the workspace and move the workspace around, refer to the topic describing the [engineering client workspace](#), [page 108](#).

Editor Menu

The table describes the submenus and commands of the menu located in the toolbar of template editors.

Command		Description
New...		<p>Opens the Global Templates dialog box, which contains the same base templates as the Toolbox of the Global Templates explorer. It allows you to create a new template in one or more existing folders of the Global Templates Library and open it in editing mode in the corresponding editor.</p> <p>For more information, refer to the topic describing how to create templates (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).</p>
Edit	Copy	<p>Copies the selected element for pasting in this template or another template.</p> <p>You can copy multiple elements by selecting them first.</p>
	Paste	<p>Pastes copied elements.</p> <p>When a naming rule applies to elements, the <code>_n</code> suffix (where <i>n</i> is an incremental number starting at 1) is used to create unique names.</p>
	Select All	Selects any element in the workspace.
	Select None	Clears any selected element in the workspace.
	Invert Selection	Selects elements that are not selected and clears any selected element in the workspace.
	Find	<p>Displays a search field, which allows you to find strings in any element of the workspace by entering entire or partial key words.</p> <p>Strings that contain the word you entered appear in a list with an indication of their path.</p> <p>Right-click a result and select Navigate to highlight it in the workspace.</p> <p>For example, entering <code>in</code> returns various results, such as:</p> <ul style="list-style-type: none"> The <code>\$InstanceID</code> system parameter. The <code>\$InstanceID</code> parameter of an element. The <code>In</code> input of a binding function. <i>Interfaces</i> to indicate that an element references interfaces.
View	Show Changes Log	<p>The commands allow you to bring the corresponding pane to the front.</p> <p>If the pane is closed, the command opens it.</p> <p>For a description of the panes, refer to the topic describing template editor panes, page 1028.</p>
	Document Outline	
	Dependencies Tree	
	Used By	
	External References	
	Locations	
	Parameters	
	Interface Rules	
	Element Rules⁽¹⁾	
	Toolbox	
	Browser	
	Properties	
Save		<p>Save button.</p> <p>Opens the Save dialog box.</p> <p>For more information, refer to the topic describing how to save changes in templates, page 1035.</p>
Save As		<p>Save as button.</p> <p>Opens the Save As dialog box.</p> <p>For more information, refer to the topic describing how to save changes in templates, page 1035.</p>

Command		Description
Export	Standard Backup	Opens the Export window, which allows you to export the template definition to file (.sbk). You can only export a template if you have saved changes. For more information on export feature, refer to the topic describing template export, page 1039.
	Various image file formats	Each command opens a Save dialog box, which allows you to create an image file with the corresponding file extension. The image captures the entire layout of the template independently of the zoom level. Panels and toolbars are not captured.
Print	Only Visible Content	Opens the Print dialog box, which lets you print on one page, the portion of the template layout that is visible in the present view.
	Editor Content On Single Page	Opens the Print dialog box, which lets you print the entire layout of the template on one page.
Close		Closes the template editor. If you have made changes, opens the Save dialog box. For more information, refer to the topic describing how to save changes in templates, page 1035.
(1) The command appears only in the menu of the Composite Editor		

Workspace Context Menu

Right-click an empty area of the workspace of a template editor to open a context menu with the following commands.

Command	Description
Inspect	The menu entry has been removed. Use the left-hand toolbar buttons instead to open the Used By , page 1009 and Dependencies Tree , page 1009 panes.
Align Left Edges	Refer to the description of the corresponding commands in the workspace toolbar , page 1023. The commands are available only if two or more elements are selected in the workspace.
Align Top Edges	
Align Right Edges	
Align Bottom Edges	
Copy	Copies the selected element for pasting in this template or another template. You can copy multiple elements by selecting them first.
Paste	Pastes copied elements. When a naming rule applies to elements, the _n suffix (where <i>n</i> is an incremental number starting at 1) is used to create unique names.
Show Changes Log	Opens the Changes Log pane, page 1028, which lets you view the history of changes that were made to the template and saved.
Properties	Brings the Properties pane to the front to display properties of the template that is open in the editor. If the pane is closed, the command opens it.

Binding Context Menu

Right-click a binding or its connector (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) in the workspace of a template editor to open a context menu with the following commands.

Command	Description
Convert To Line	Lets you toggle between binding styles (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).
Convert to Connector	
Go To	<p>For bindings in line style, opens a submenu that contains the name and path of objects and/or their properties that are linked to this object through bindings so you can locate them (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>). When you select an entry, the object and the binding are highlighted in the workspace.</p> <p>For bindings in connector style, highlights the object that is connected to the other end of the binding.</p>
Update Binding	<p>Opens the Update Binding dialog box, which lets you move the destination (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) of the binding to another property or element.</p> <p>The command is not available for source connectors of bindings.</p>
Delete	Deletes the binding after you confirm the command.

Common Template Editor Panes

Introduction

The **Facet Editor** and **Composite Editor** use various panes to group properties and data that are related.

This topic describes panes that are common to both. Panes that are specific to either editor are pointed out.

Certain panes may be available only in editing mode.

For information on working with panes, refer to the topic describing the engineering client workspace, page 113.

NOTE: This topic does not describe how to define or modify the contents of the panes.

Editor Panes

The table describes the purpose of the panes that you can access in the **Facet Editor** and **Composite Editor**.

Pane	Description
Document Outline	<p>Allows you to view and/or edit:</p> <ul style="list-style-type: none"> • Parameters • Elements • Interfaces <p>Click a parameter, element, or interface to display its properties in the Properties pane. The corresponding item is selected in the workspace.</p> <p>NOTE: To view and/or edit the properties of the template, click Parameters.</p>
Interface Rules	Allows you to view and define the rules for the interfaces (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) referenced by the template.
Element Rules⁽¹⁾	Allows you to view and define the rules for the elements (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) referenced by the template.
Properties	Allows you to view and/or edit the properties of the selected element of the template.
Toolbox	<p>Allows you to browse and select binding functions (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) to be used in the template.</p> <p>For facet templates of the Supervision category, the Toolbox pane also contains the following elements (see EcoStruxure™ Process Expert, Global Templates, Reference Manual):</p> <ul style="list-style-type: none"> • For facet templates of the Data subtype: <ul style="list-style-type: none"> ◦ Advanced Alarm ◦ Calculated Variable Tag ◦ Disk Variable Tag ◦ Digital Alarm ◦ Equipment ◦ Equipment Parameter ◦ Equipment Group of Messages ◦ Local Variable Tag ◦ Message ◦ Time Stamped Digital Alarm ◦ Trend Tag ◦ Variable Tag

Pane	Description
	<ul style="list-style-type: none"> For facet templates of the Genie subtype: <ul style="list-style-type: none"> Equipment Equipment Parameter Equipment Group of Messages For facet templates of the Client Event subtype: Event. For facet templates of the Server Event subtype: Report. <p>These elements contain the same properties as the corresponding elements of the Supervision Participant and allow populating the respective databases.</p>
Browser	<p>In the Facet Editor, allows you to browse and select interfaces to be referenced by the template.</p> <p>In the Composite Editor, allows you to browse and select facet and composite templates, and interfaces to be referenced in the template.</p>
Dependencies Tree	<p>In the Facet Editor, allows you to view the interfaces that the template references.</p> <p>In the Composite Editor, allows you to view the interfaces, facet and composite templates that the template references.</p> <p>You can right-click an item to view and/or edit it in a new editor. You can also view the dependencies of the item or which other templates reference it.</p>
Used By	Allows you to view the templates that reference the template, which is open in the editor. You can right-click a template in the Used By pane to open it in a new editor. You can also view its dependencies or which other templates reference it.
External References	<p>Allows you to view:</p> <ul style="list-style-type: none"> In which systems an instance of the template exists. For each system, the identifier of the instances using the template. <p>You can right-click an application instance to:</p> <ul style="list-style-type: none"> Open the folder containing the instance in the Application Explorer. Open the Inspect Instance window, page 211.
Parameters (see EcoStruxure™ Process Expert, Global Templates, Reference Manual)	<p>The Parameters pane allows you to:</p> <ul style="list-style-type: none"> View and manage parameter data that is used during instantiation to customize elements of an instance. Customize the layout of the parameters that are visible when you edit an instance by using the Instance Editor: <ul style="list-style-type: none"> Change the position of parameters within their category. Change the position of categories. Assign parameters to another category.
Locations	<p>Allows you to view the path to the template and its copies, page 1038.</p> <p>Right-click an entry and select Open reference container to open the corresponding folder in the Global Templates explorer.</p>
Changes Log	<p>Keeps track of changes that are made to the template and saved. The pane indicates:</p> <ul style="list-style-type: none"> The action that was performed. The state of the template after the change. The mandatory description of changes that is entered when the template is saved. The version of the template after the change. The user name of the user logged in to the engineering client.
(1) The pane appears only in the Composite Editor .	

Common Template Editor Filters

Overview

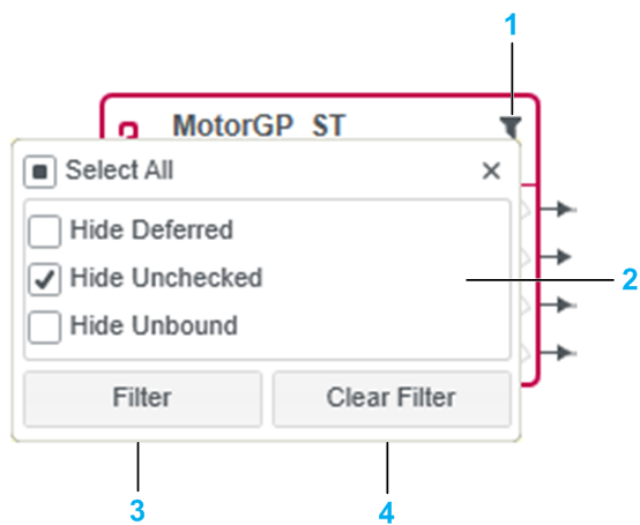
Starting with version 2021, the following context menu commands of elements, which appear in the **Facet Editor** and **Composite Editor** have been replaced by a filter menu.

- **Show/Hide Unchecked**
- **Show/Hide Unbound**

Also, the **Hide Deferred** filter criteria has been added.

Filter Menu Description

The following figure shows an example of the filter menu that is located in the header an element with its default setting.



Item	Description	
1	Click the Filter button to open the element filter menu. When at least one filter is applied, the button is shown with a black fill.	
2	Hide Deferred	When selected, hides properties and interfaces that are deferred.
	Hide Unchecked	When selected, hides properties and interfaces with an eye icon that is located on the right-hand side and that is not visible (unchecked). To uncheck a property/interface, click its icon. The icon disappears. NOTE: A property/interface that has a binding connected to it cannot be hidden even if it is unchecked.
	Hide Unbound	When selected, hides properties and interfaces that have no binding connected.
3	Click the Filter button to apply the selected filter criteria to the representation of the element.	
4	Click the Clear Filter button to clear the filter criteria selection and update the representation of the element.	

Using the Radar View

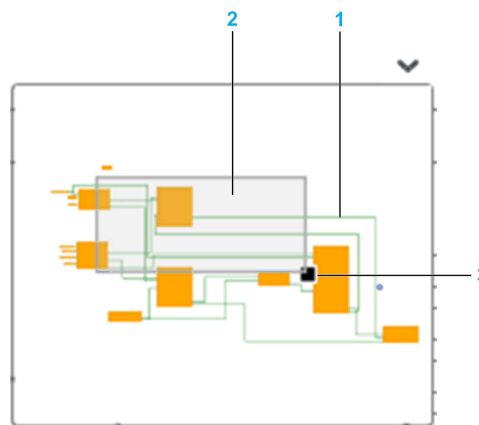
Overview

The radar view appears in the bottom right corner of certain graphical editors. It shows you which area of the layout of a template or workspace you are seeing in the editor.

You can also use it to navigate the inside the workspace and adjust the zoom level.

Description

The following figure shows an example of the **Radar View** in a template editor.



Item	Description
1	Shows all the objects that appear in the editor. The view is refreshed when you modify the layout of objects.
2	Area that is currently visible in the editor. You can move the translucent rectangle by dragging it with the pointer. The view inside the editor moves accordingly. Alternatively, click anywhere inside the Radar View to move the center of the translucent rectangle to where you have clicked. When you change the zoom level, the size of the translucent rectangle adjusts accordingly.
3	Zoom level adjustment. Dragging the corner inward or outward lets you zoom in or out respectively. NOTE: While the pointer is over the Radar View , you can also use the mouse wheel to zoom.

When the **Radar View** is collapsed, it appears as a button 

Using the Simplified View in Template Editors

Overview

Zooming out to the lowest level in the **Composite Editor** or **Facet Editor** switches the display to the simplified view.

This view mode lets you see, at a glance, the elements of a template and the relation between them without showing the details of bindings and element properties.

Some template engineering functionality are not available.

To exit the simplified view, zoom in.

Restrictions of the Simplified View

The table describes the restrictions in the template editor while the simplified view is active.

Action or object	Restrictions
Actions related to bindings (such as creating, moving, deleting, switching styles).	Actions are not allowed
Creating parameters or platform inputs.	
Element properties and deferred inputs and outputs.	Are not shown.
Multiple bindings to and from an element.	Are shown as a single line.
Binding connectors.	Are not shown. Bindings are shown in line style instead.

Description

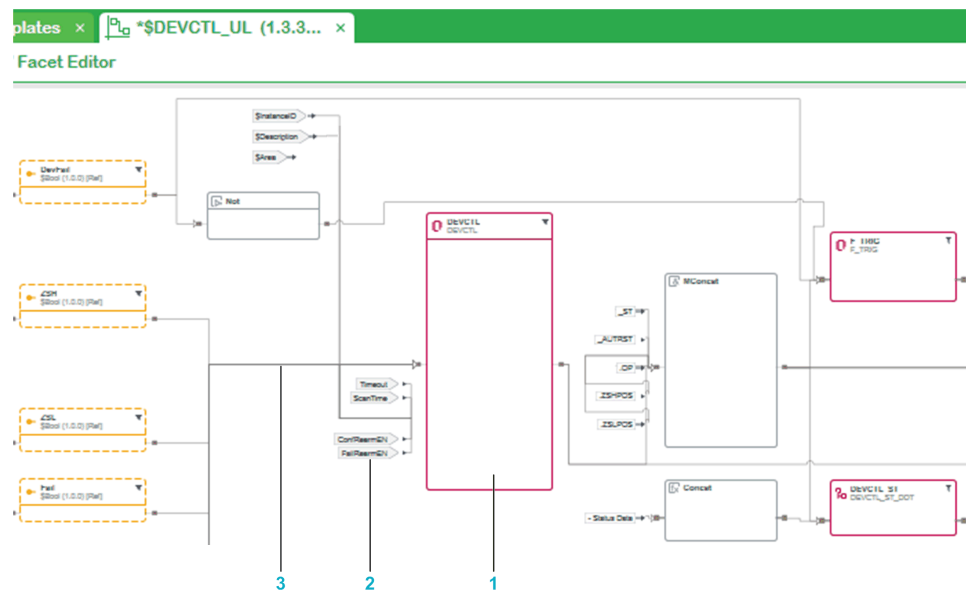
The following figure shows an example of template as it normally appears in the **Facet Editor**.

States x [P] *\$DEVCTL_UL (1.3.3... x
Facet Editor

The diagram illustrates a complex template structure. On the left, there are four sub-templates: DevFail, ZSH, ZSL, and Fail, each with its own set of inputs and outputs. These are connected to a central component labeled DEVCTL, which has numerous inputs and outputs. To the right of DEVCTL, there is an MConcat component, followed by F_TRIG and DEVCTL_ST. The connections between these components are represented by lines, some of which are grouped together, indicating multiple bindings to a single element. The layout is organized to show the flow of data or control from left to right.

Item	Description
1	Elements keep their relative size and position to preserve the aspect of the layout.
2	Parameters and platform inputs are shown in their actual position.
3	Bindings to and from each element are grouped into a single line.

The following figure shows the same template when simplified view is active (the view is enlarged for better visibility).



Managing Global Templates

What’s in This Chapter

Managing Global Templates 1035

Updating, Replacing, and Duplicating Global Templates 1051

Managing Global Templates

Saving Changes In Global Templates

Overview

When Global Templates are open in their respective editor, page 1007, you can save changes by using the following commands.

Save	Allows you to save changes to templates that you have created by using various save options. Use the command to rename a template or change its usability state. The command is available only if the template is not referenced by another template and not used by an application instance; otherwise, the Save as dialog box opens.
Save as	Allows you to save changes to Schneider Electric Global Templates, create a new template, or a new version of a template.

When you select the **Save as** command, you can select one or more locations where to store the template.

If you have edited the template from within its parent, you have the possibility to update the parent template after closing.

NOTE: To save a template with a different identifier and/or version without editing it, use the **Duplicate** command, page 1065.

Restrictions When Saving Schneider Electric Templates

Restrictions (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) related to the template identifier apply when you save a Schneider Electric template.

Tracking Saved Changes

When you save a template, a record of the save operation is kept. You can view details in the **Changes log** pane, which is available in each template editor, page 1028.

Save Options

The table describes the items of the **Save** and **Save as** dialog boxes that is displayed when you select the respective command.

Item	Description
Keep Version⁽¹⁾	Select this versioning scheme to save changes in a template without changing its version number. Selected by default.
New Build	For information on the usage of the different version components, refer to Version (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).
New Minor	
New Major	
Other	By default, New Build is selected when you open the Save As dialog box. Selecting this versioning scheme allows you to edit the following parameters in the dialog box before saving the template: <ul style="list-style-type: none"> The identifier. The three version components (major, minor, build number). Select it to rename the template or create a new template based on the one you are editing.

Item	Description
	NOTE: By default, the version that is selected is the same as for New Build .
Version	For a description of the version format, refer to Version (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).
Identifier	Within the Global Templates Library, the combination of identifier (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) and version must be unique for each template.
Locations⁽³⁾	<p>Allows you to select one or more existing folders of the Global Templates Library to store the template.</p> <p>Default value: The present location of the template. If linked copies, page 1038 exist, their location is indicated also.</p> <p>Click the browse button to open the Select Template Location dialog box, which lets you change locations and create linked copies by selecting additional locations.</p>
Usability State⁽¹⁾	<p>For information on the parameter, refer to Usability State (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).</p> <p>Default value: The present state of the template.</p>
New Version Usability State⁽²⁾	<p>Allows you to define the usability state of the new template that you are creating.</p> <p>Default value: Not Approved</p> <p>For information on the parameter, refer to Usability State (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).</p>
Old Version Usability State⁽²⁾	<p>Allows you to define what will be the usability state of the template that you are currently editing after you create the new template.</p> <p>Default value: Approved</p> <p>For information on the parameter, refer to Usability State (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).</p>
Changes Description	<p>You must enter a description by using free-form text to be able to save changes.</p> <p>The description that you enter is visible in the Changes log of the template editor. Entering a detailed description of changes allows you to keep track of the version history of the template.</p>
<p>(1) The item is displayed only in the Save dialog box.</p> <p>(2) The item is displayed only in the Save As dialog box for templates that are already referenced or instantiated.</p> <p>(3) The item is displayed only in the Save As dialog box.</p>	

Saving Changes In Global Templates

Modifying Schneider Electric Global Templates or creating templates may affect the function of these templates and/or systems, and must be performed by qualified personnel. Before proceeding, refer to the [Overview of this part](#), page 990.

To save changes in a Global Template, proceed as follows.

Step	Action
1	<p>In the template editor, click the Save or Save As button.</p> <p>Result: The corresponding dialog box opens.</p>
2	Select the versioning scheme that you want to use.
3	Edit the Identifier , Change Description , and/or Usability State as needed.
4	<p>Click the Save button.</p> <p>Result:</p>

Step	Action
	<ul style="list-style-type: none">For the Save dialog box:<ul style="list-style-type: none">The dialog box closes.Changes are saved in the template.The template remains open inside the editor.For the Save as dialog box:<ul style="list-style-type: none">A new template, which includes your changes is created. It is opened in the editor.The source template is closed without saving. <p>NOTE: Click the Cancel button to close the Save or Save as dialog box without saving changes.</p>
5	<p>Close the template.</p> <p>Result: If you have edited the template from its parent and the parent is open in edit mode, the Update/Replace References dialog box opens. Otherwise, the template is closed.</p>
6	<p>Click either button:</p> <ul style="list-style-type: none">Yes: Closes the template and updates/replaces, page 1052 all the references of it in the parent template. The parent template is shown. For each reference for which at least one binding cannot be recreated, a dialog box opens. It contains detailed information about the conflicts and lets you skip the update/replacement.<p>NOTE: Other templates that also reference the template that you have edited are not updated.</p>No: Closes the template.

Copying and Pasting Global Templates and Folders

Creating Linked Copies of Templates

The **Copy** and **Paste** commands let you place copies of a template in other folders of the Global Templates Library. The copies are linked to the source template. Any change that you make to the source template is also made to each copy that you have created.

NOTE: To create an independent copy of a template, in the same or a different folder, use the **Save as** or **Duplicate** command from the corresponding template editor or **Global Templates Explorer**.

Locating Copies

To locate the linked copies of a template, use the **Locations** context menu command, [page 1001](#).

Deleting Copies

To delete a linked copy of a template, use the **Remove** context menu command, [page 1001](#).

Copying Folders

You cannot paste an empty Library folder.

When you copy a folder containing templates and you paste it on another folder, a subfolder with the same identifier is created, which contains a linked copy of each template of the source folders. Subfolders and their contents are created in the same way.

Exporting Templates

Overview

The **Export** command of the **Global Templates Explorer** lets you export templates of the Global Templates Library.

The process creates an export file in **.sbk** format that contains:

- The selected templates.
- Referenced composite and facet templates, interfaces, and their parameter configuration.
- Local constituents.
- Supervision content.

User contents such as [content containers](#), [page 1121](#) containing documents and URLs that are linked to a template are not exported.

The export functionality allows you to copy these templates to another system or platform, or to create a backup.

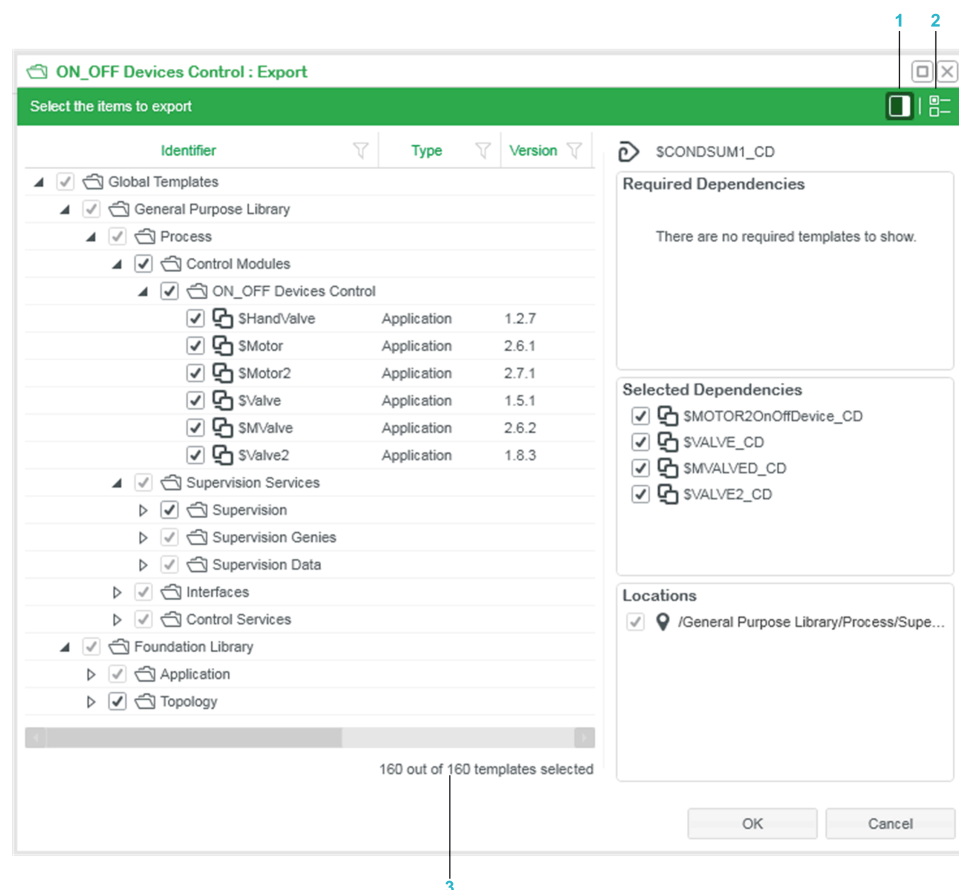
Aborting Export Tasks

After confirming the export by clicking **Save** in the **Save** window, the Abort icon is displayed in the [notification panel](#), [page 85](#). Click the icon to cancel the task. The export file is not created.

Only the user who selected the command is allowed to cancel it.

Template Export Window

The following figure shows an example of the template window that opens after selecting the **Export** command from the *ON_OFF Devices Control* sample folder.



Item	Description
1	Button to show or hide the details pane (shown on the right-hand side of the window). Select a template by clicking its row in the tree or grid view to view details.
2	Button to toggle between grid view and tree view. Tree view is the default view (shown here).
3	Indicates the number of selected templates and the total number of templates contained in the location from which you have selected the Export command. <i>Linked copies, page 1038</i> are not counted even if they are selected.

NOTE: To expand/collapse a node in the **Export** window, select the node and press **Enter**.

The table describes the information that appears in the columns of the **Export** window.

Column Header	Description
Identifier	Displays: <ul style="list-style-type: none"> The folder hierarchy in the Global Templates Library (tree view mode only). The identifier of templates contained in the location from which you have selected the Export command. Check boxes that let you select the templates you want to export. In grid view mode, only templates are shown, not linked copies. NOTE: By default, all templates are selected.
Type	Indicates the category of the template. Refer to the definition header of each template category for a description of the <i>type</i> , <i>page 36</i> .
Version	Indicates the version of the template.
State	Usability state (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) of the template (for example, Approved or Deprecated).
Description	Description of the template.

The table describes the details pane of the **Export** window.

Section Header	Description
–	Identifier of the selected template. NOTE: You can view details even if the check box next to the template identifier is not selected.
Required Dependencies	Identifier of templates that are required to export the selected template. The check box of required dependencies is selected and disabled. To enable it, you need to clear the check box of the selected template.
Selected Dependencies	Identifier of templates that reference the selected template. When you clear the check box of a selected dependency, the template is removed from this section. Once you have cleared the check boxes of all the listed templates of this section, the check box of the selected template is enabled and you can clear it to exclude it from the export file. As long as selected dependencies are listed, the check box of the selected template remains disabled.
Locations	Path of the selected template and any of its linked copies in the Global Templates Library. For each template and linked copy, the status of the check box in the Identifier column is duplicated.

Creating the Export File

To export templates and create the export file, proceed as follows.

Step	Action
1	<p>From the tree view of the Global Templates Explorer, right-click and select Export on:</p> <ul style="list-style-type: none">• A template if you want to export the template only• A template folder if you want to export any templates of the folder, subfolders, and any templates contained in these subfolders <p>Result: The Export window opens.</p>
2	Select the templates that you want to export.
3	<p>Click OK.</p> <p>Result: The Save dialog box opens.</p>
4	<p>Enter a name for the export file and select a location, and click Save.</p> <p>Result: The selected items are exported and an export file in <i>.sbk</i> format is created at the location that you have selected.</p>

NOTE: To export the templates of an entire Library, create separate export files by selecting various subfolders in the **Export** window. This creates smaller export files that take less time to import.

Importing Templates

Overview

The **Import** command of the **Global Templates Explorer** allows you to do the following in the Global Templates Library.

- Import new template.
- Import new versions of existing templates.
- Modify or update templates or parts of them.
- Import new and/or updated **Included** projects (.ctz) that are required by the templates you are importing.

Importing templates adds them to the database that is mounted.

When you select an export file (.sbk) containing templates, EcoStruxure Process Expert compares the information in the export file with the templates contained in the Global Templates Library and applies rules to keep templates unique.

The **Import** window allows you to select which valid templates of the export file you can import.

NOTE: Importing many templates may take time.

Export File Location

The export file containing the template information must be located in a folder that you can write to.

Import Rules

The following table describes the rules that are applied before import when comparing templates contained in the export file with those in the Global Templates Library.

Template identifier	Template version	Template signature	Result
Does not exist in the Global Templates Library.	Indifferent	Indifferent	The template can be imported.
Exists in the Global Templates Library.	Exists in the Global Templates Library.	Exists in the Global Templates Library.	The template cannot be imported.
		The signature is different.	The template cannot be imported and a message about the detected conflict is displayed.
	The version is unique.	The signature is unique.	The template can be imported.

Importing Supervision Included Projects

An **Included** project (.ctz) that is used by a template is imported only if the template is imported.

For example, if you modified the **Included** project that is used by the *MyMotor* template v1.0.0 but without making a change to the template, you cannot add the modified **Included** project to the [content repository](#), [page 1120](#) by importing the template if *MyMotor* v1.0.0 already exists in the Global Templates Library; because the template exists in the Library, it is not imported.

The following table describes the methods that you can use to add a new **Included** project to the content repository.

Method	Description
Modifying the template that references the new Included project so that it can be imported.	<p>You must modify at least one Supervision genie facet template (_CG) of the template.</p> <p>For example, you can create a new version, page 1035 of the genie facet template without changing its content by using the Save As command.</p> <p>Importing the template allows you to import the new Included project and add it to the content repository.</p>
Adding the new Included project to the content repository.	<p>Refer to the procedure describing how to add project files to the content repository (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>).</p> <p>When following the procedure, edit either facet template:</p> <ul style="list-style-type: none"> A new Supervision genie facet template that you create by using the Genie template from the Toolbox of the Global Templates Explorer. An existing Supervision genie facet template (_CG) of the Global Templates Library. <p>In both cases, you do not need to save the template to add the Included project to the content repository.</p>

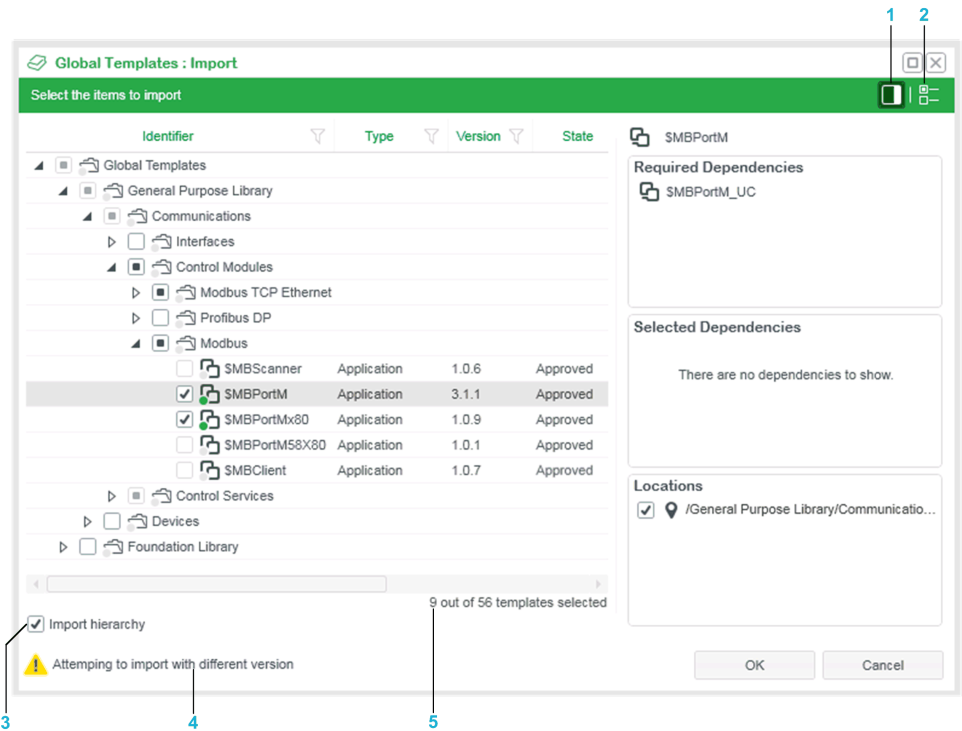
Opening the Template Import Window

To open the **Import** window, proceed as follows.

Step	Action
1	<p>From the client toolbar, click the Global Templates explorer icon.</p> <p>Result: The Global Templates Explorer window opens.</p>
2	<p>Right-click the Global Templates root folder and select Import.</p> <p>Result: The Import dialog box opens.</p> <p>NOTE: If you know the location where you want to import the templates, right-click the corresponding folder in the tree view of the Global Templates Explorer and select Import.</p>
3	<p>Select the export file (.sbk) containing the templates that you want to import and click Open.</p> <p>Result: The Import window opens and displays the content of the export file.</p>

Template Import Window

The following figure shows an example of the **Import** window.



Item	Description
1	Button to show or hide the details pane (shown on the right-hand side of the window). Select a template by clicking its row in the tree or grid view to view details.
2	Button to toggle between grid view and tree view (default view).
3	Check box yo import templates with the hierarchy of folders that existed at the time of export. Clear the check box to import templates in the same folder without hierarchy. The check box is selected by default. In grid view, selecting or clearing the check box switches to tree view.
4	Information to indicate that the templates were exported from a version that is different from the version you are using.
5	Information to Indicate the number of selected templates and the total number of templates contained in the export file. Linked copies (templates marked with a blue dot) are not counted even if they are selected.

NOTE: To expand/collapse a node in the **Import** window, select the node and press **Enter**.

The following table indicates the meaning of the colored dots that appear in the **Import** window next to template identifiers.

Color code	Description
Green	The template does not exist in the Global Templates Library and can be imported. For a node, it indicates that the elements of the node can be imported.
Gray	The template already exists in the Global Templates Library and cannot be imported. For a node, it indicates that at least one element of the node already exists and is identified by a gray dot.
Blue	The template already exists in the Global Templates Library but a <u>linked copy</u> , page 1038 is created if imported.
Red	The template cannot be selected for import because of a detected <u>signature conflict</u> , page 1042 .

The following table describes the information that appears in the columns of the **Import** window.

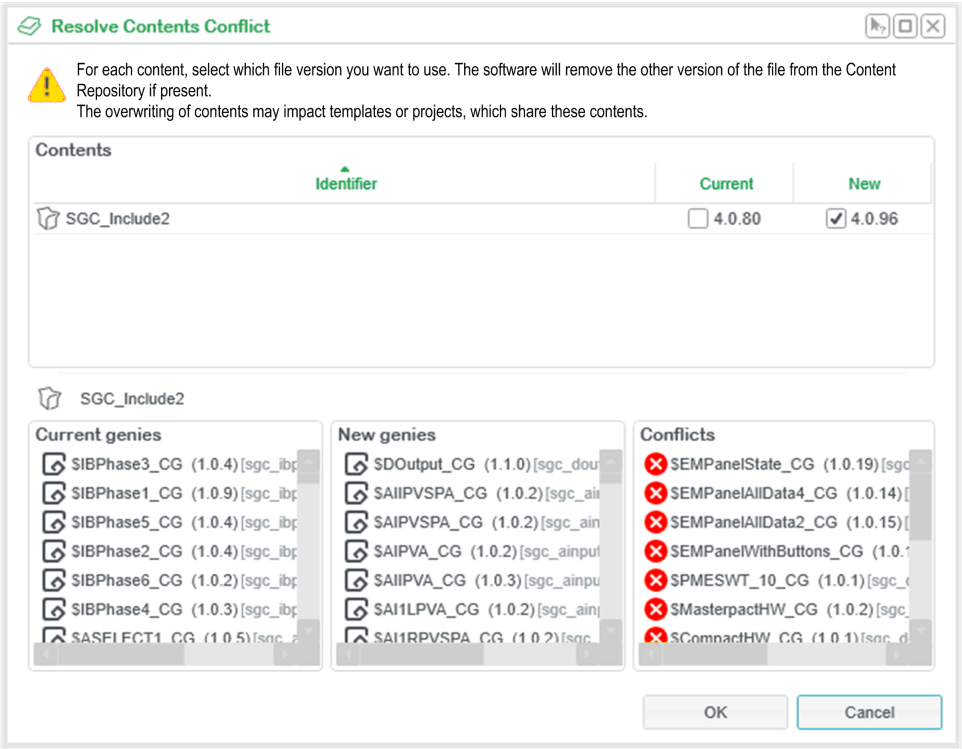
Column Header	Description
Identifier	<p>Displays:</p> <ul style="list-style-type: none"> The folder hierarchy at the time of export (when Import hierarchy is selected). The identifier of templates contained in the export file. A colored dot. Check boxes allowing you to select the templates that you want to import. <p>Selecting a template automatically selects referenced templates that can be imported. It also selects linked copies of referenced templates if the referenced template does not exist in the Global Templates Library yet.</p> <p>In tree view, if Import hierarchy is not selected, EcoStruxure Process Expert can only import one of each template and/or one linked copy of each existing template because templates and linked copies are imported to the same folder.</p> <p>In grid view, only templates are shown without linked copies. If a template exists, its check box is cleared and disabled. Templates that do not exist have their check box selected by default and enabled.</p> <p>NOTE: By default, new templates and new template versions are selected.</p>
Type	<p>Category of the templates in the export file.</p> <p>Refer to the definition header of each template category for a description of the type, page 36.</p>
Version	Version of templates in the export file.
State	Usability state (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) of the template (for example, Approved or Deprecated).
Description	Description of templates in the export file.

The following table describes the details pane of the **Import** window.

Section Header	Description
–	Identifier of the selected template.
Required Dependencies	Identifier of templates that are referenced by the selected template.
Selected Dependencies	<p>Identifier of templates that reference the selected template (only if the selected template can be imported).</p> <p>If you clear the check box of a selected dependency, it clears the check box of templates and linked copies referencing the selected template. This allows you to clear the check box of the selected template, which would be disabled otherwise.</p>
Locations	<p>Path of the template and its linked copies at the time of export.</p> <p>For each template and linked copy, the status of the check box in the Identifier column is duplicated.</p> <p>NOTE: If you clear Import hierarchy:</p> <ul style="list-style-type: none"> Linked copies are not shown. The path does not reflect the hierarchy at the time of export.

Resolving Included Project Conflicts

If an **Included** project with the same name but a different version is detected in the content repository, the **Resolve Contents Conflict** dialog box opens. The following figure shows an example of the dialog box.



Header	Description
Identifier	Name of the Included project that conflicts with the project in the content repository.
Current	Version of the Included project in the content repository.
New	Version of the Included project in the export file. NOTE: If the Current and New versions are identical, it indicates that the signatures of the existing and the imported Included projects are different.
Current genies	Animated graphics (genies) of the Included project in the content repository.
New genies	Animated graphics of the Included project in the export file.
Conflicts	Animated graphics that create a conflict if you import the Included project.

Consider the following information for conflicting **Included** projects before proceeding with the import:

Importing an Included project	<p>The animated graphics that are used by existing templates of the Global Templates Library and instances thereof may be overwritten.</p> <p>The status of Supervision project executables that use the Included project that you are replacing with the imported one may be set to Out Of Date if it is currently Built. In this case, you must build these executables to integrate in the corresponding Supervision projects the changes related to the Included project.</p>
Using the Included project in the content repository	The Included project that is used by existing templates of the Global Templates Library and instances thereof is not affected. However, templates that you are importing may be missing the animated graphics that they require to function properly.

NOTE: Refer to the **Conflicts** section of the **Resolve Contents Conflict** dialog box for a list of impacted animated graphics.

NOTE: Refer to the *Library Release Notes*. The document contains information about changes that were made to templates of the corresponding Library.

Aborting Import Tasks

After confirming the import by clicking **OK** in the **Import** window or in the **Resolve Contents Conflict** dialog box, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. No templates and **Included** projects are imported.

Only the user who selected the command is allowed to cancel it.

Importing Templates

NOTICE

LOSS OF DATA

Resolve an **Included** project conflict that arises during the import of templates by selecting the appropriate version of the **Included** project to be used.

Failure to follow these instructions can result in equipment damage.

To import templates into the Global Templates Library, proceed as follows.

Step	Action
1	In the Import window, select the templates, page 1044 that you want to import.
2	<p>Click OK.</p> <p>Result: If an Included project with the same name but of a different version is detected, the Resolve Contents Conflict dialog box opens; otherwise, the import of selected templates from the export file is performed.</p>
3	<p>If the Resolve Contents Conflict dialog box is displayed, select the version of the Included project, page 1046 that you want to use and click OK.</p> <p>Result:</p> <ul style="list-style-type: none"> The selected templates are imported. Completed is displayed in the notification panel once the import is complete. <p>NOTE: Click Cancel to revert to the Import window without importing templates.</p>

After Importing Templates

Depending on the number of templates that you imported and your hardware configuration, the import process may impact the performance of your computer. A restart may be required to restore the performance level.

NOTICE

LOSS OF COMMUNICATION

Stop the system server only when no client is running.

Failure to follow these instructions can result in equipment damage.

After importing templates, proceed as follows.

Step	Action
1	Quit clients, page 71.
2	Stop the system server.
3	Restart the computer.

Purging the Global Templates Library

Overview

The **Purge** context menu command of folders of the **Global Templates Explorer** allows you to remove unused, earlier versions of templates and interface models from the Global Templates Library. The process also removes local constituents of deleted templates and unused global constituents, page 1121.

You can select which templates to remove.

At the end of the purge process, the notification panel provides a summary containing information on the purged templates.

If the process ends abnormally, the process is rolled back.

Purge Rules

The purge deletes facets, composite templates, and interface models that fulfill the following requirements:

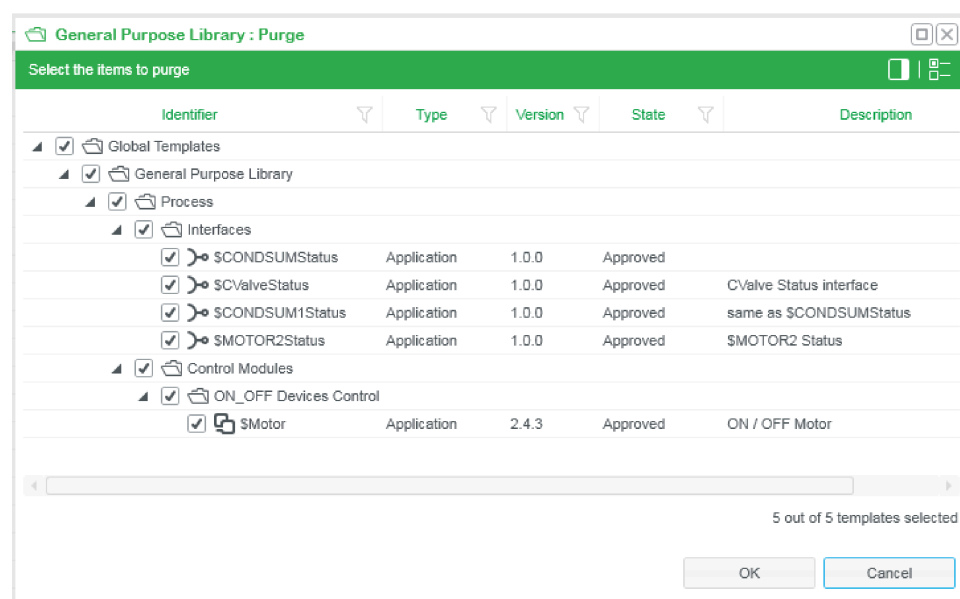
- A template with a higher version number having the same identifier and the Usability State (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) **Approved** exists in the Global Templates Library.
- The template or the interface model is not referenced (used) by another template of the Global Templates Library.
- The template is not used by an application instance.

NOTE: The purge does not delete a template if the template is used by a facet that is assigned to a project and the **Assignment** status of the facet is **Deleted**, page 425.

To delete the template, generate the project or the container to which the facet is assigned.

Purge Window

The following figure shows an example of the **Purge** window when the **Purge** command is selected from the **General Purpose Library** folder.



Aborting Purge Tasks

After confirming the **Purge** command by clicking **OK** in the **Purge** window, the Abort icon is displayed in the notification panel, page 85. Click the icon to cancel the task. No templates are purged.

Only the user who selected the command is allowed to cancel it.

Purging Global Templates

You cannot restore Global Templates that you delete from the Global Templates Library by using the **Purge** command.

<p style="text-align: center;"><i>NOTICE</i></p> <p>LOSS OF DATA</p> <p>Back up the database before using the Purge command.</p> <p>Failure to follow these instructions can result in equipment damage.</p>
--

NOTE: Refer to the topic describing how to back up the database (see *EcoStruxure Process Expert, Installation and Configuration Guide*).

To purge templates from the Global Templates Library, proceed as follows.

Step	Action
1	Back up the database.
2	Right-click the parent folder from where you want to start the purge and select Purge . Result: The Purge window opens. After a few moments, it displays the templates that satisfy the purge rules in the selected location.
3	Select the templates that you want to delete and click OK . Result: <ul style="list-style-type: none">The selected templates and interface models are deleted.Information appears in the notification panel when the purge is complete.

Updating, Replacing, and Duplicating Global Templates

Overview

This chapter describes the various commands that are available to modify Global Templates with or without editing them.

Template Modification Strategy

Overview

Various commands are available that allow you to modify Schneider Electric and/or user-created templates. Select the appropriate command depending on:

- Whether you want to edit a template or not.
- Whether the identifier of the template or one of its child templates starts with the \$ prefix (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).
- The scope of changes to be performed.

Starting with EcoStruxure Process Expert 2021, when you edit a template from within its parent, you have the possibility to [update its references](#), [page 1052](#) in the parent template after you save changes and close.

The table shows the commands that are available to modify Global Templates and their specifics.

Command	Use in	Use with	Purpose	Comment
Update	Global Templates Explorer	Templates	To update a template with the latest version of any child templates that exist in the Global Templates Library.	<ul style="list-style-type: none"> • Cannot be used to update templates whose identifier starts with the \$ prefix. • The latest versions of templates with the same identifier are found automatically. • When used at the folder level, you can select which templates to update.
		Folders	To update the templates inside a folder, page 1058 , and optionally inside its subfolders, with the latest version of any child templates that exist in the Global Templates Library.	
	Template editors	Elements	To update the template that is used by a specific element with the latest version that exists in the Global Templates Library.	<ul style="list-style-type: none"> • You can update the template that is used by a specific element (instance). • The latest version is found automatically.
Replace	Template editors	Elements	To update the template that is used by a specific element with a different version or a different template that exists in the Global Templates Library.	<ul style="list-style-type: none"> • Lets you replace the template that is used by a specific element (instance). • Can be used to undo a template update.
Duplicate	Global Templates Explorer	Templates	To create a new template from an existing one with the possibility to keep, duplicate, or replace each child template that is referenced.	<ul style="list-style-type: none"> • The replace action impacts all elements (instances) of a given template. • Can be used in place of the Update command with templates whose identifier starts with the \$ prefix.
Edit/Extend Interface	Facet and composite template editors	Interface references	To edit, add, and remove elementary elements in an interface, page 1070 from within its parent template.	<ul style="list-style-type: none"> • Provides basic modification capability. • Updating templates that use the other role of the interface is required. • The command is not available for deferred interfaces and when the interface element contains a nested interface.

Updating References in Parent After Editing Child Templates or Interfaces

When you edit a template or interface by opening it from within its parent, after saving changes and closing it, [page 1035](#), the **Update/Replace References** dialog box opens. It lets you automatically update in this parent template all the references of the template/interface that you have edited. This requires that the parent template is open in a template editor in edit mode. If it is open in read-only mode, you can switch it to edit mode.

If you rename the template while saving changes, references are replaced instead of being updated.

Once you save changes and close the parent, you are again given the choice to update the next higher level template given it is open, and so on. Refer to the example, page 1075 for details.

The impact of the template update/replacement is the same as when you use the **Update** or **Replace** command.

NOTE: Other templates that also reference the template that you have edited are not updated.

NOTE: The **Update/Replace References** dialog box opens also when you edit/extend an interface in-place, page 1070 or edit a nested interface.

Impact of Template Update or Replacement

The table describes the impact on the various template components when you update or replace a template.

Current template component	In new template	Impact
Bindings to the element selection of composite references	Are the same as in current template.	The bindings are retained.
	Are not the same as in current template.	The bindings to the element selection of the composites that are not part of the new template are removed.
Bindings and attributes of the element parameters	Are the same as in current template.	The bindings and attributes are retained.
	Are not the same as in current template.	The bindings and attributes of the element parameters that are not part of the new template are removed.
Bindings to element interfaces	Are the same as in current template.	The bindings are retained.
	Are not the same as in current template.	The bindings to the element interfaces that are not part of the new template are removed.

Applying Template Modifications to Existing Instances

Modifications to templates must be propagated to application instances to take effect. The method to apply modifications to existing instances depends on the template identifier.

Result of template modification	Method to apply
A new version of the template (same identifier)	Update the template used by instances, page 220
A template with a new identifier	Replace the template used by instances, page 220

Updating Global Templates and Templates of Elements

Overview

The **Update** command lets you perform different actions depending on where you use it.

In the **Global Templates Explorer**, it lets you create a new version of your template, in the same location, by using the latest approved version of any of the child templates that are referenced by this template.

In the **Facet Editor** and the **Composite Editor**, it lets you update the template or interface model that is used by a specific element by the latest available version.

In both cases, the update process allows you to integrate in the composition of your template the latest versions of composite and facet templates, and interface models.

NOTE: The update of the template of an element impacts templates that reference this element and requires [updating, page 1052](#) all the higher-level templates.

Restrictions When Updating Schneider Electric Templates

Restrictions (see *EcoStruxure™ Process Expert, Global Templates, Reference Manual*) related to the identifier of the parent template apply when you update a Schneider Electric template.

You cannot use the **Update** command in the **Global Templates Explorer** on a template whose identifier contains the \$ prefix. You need to use the **Duplicate** command, [page 1065](#) instead.

Good Practices

- If you know which highest level templates reference templates of which a later version is available, proceed with the update from these highest level templates. Typically, the highest level template is the control module template.
- If you do not know which highest level templates need to be updated, use the **Inspect > Used By** context-menu command on the template of which you have created a new version. This allows you to determine which are the highest level templates that you need to update with the new version child template.

Working Principle of the Update Functionality

When the **Update** command is used on a template in the **Global Templates Explorer**, EcoStruxure Process Expert proceeds as follows:

- It searches the tree of dependencies of the template for any element with the same identifier and for which a later version exists in the Global Templates Library.
- It displays in a window the latest version of templates that it has found in the dependencies and lets you select:
 - The usability state for the current version of templates after the new versions are created.
 - The versioning scheme (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) and usability state for the new version of templates that are created.
- Once you proceed, it creates the new templates, proceeds with the update, and processes bindings (see *EcoStruxure Process Expert, Global Templates, Reference Manual*).

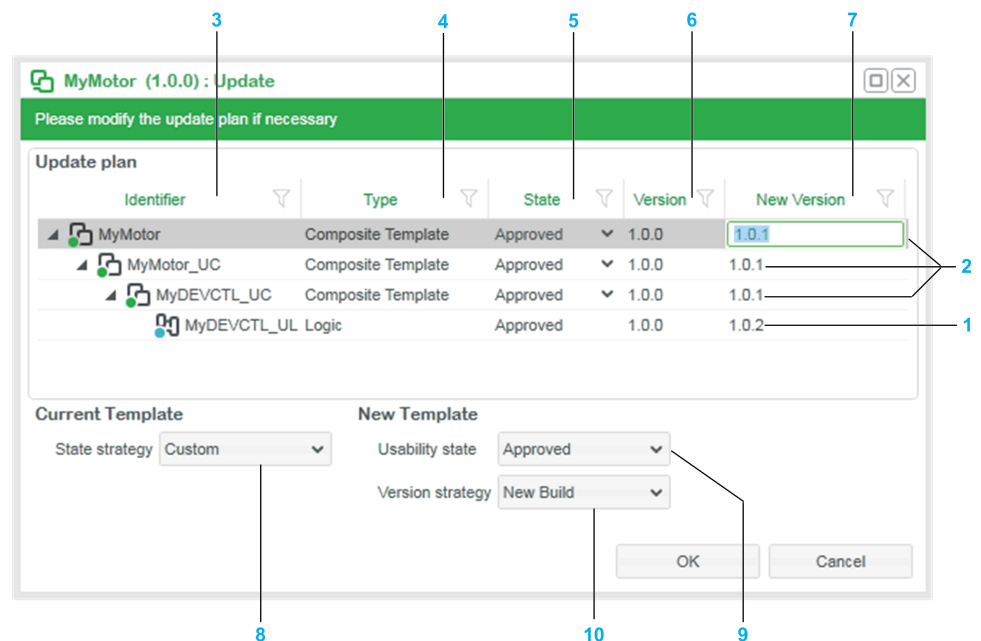
When the **Update** command is used in a template editor, EcoStruxure Process Expert proceeds as follows:

- It verifies if a template with a later version exists in the Global Templates Library.
- If so, it processes bindings and:
 - If any binding cannot be recreated, it informs you and gives you the choice to proceed or cancel the update.
 - If bindings can be recreated, it proceeds with the update of the template.

NOTE: For the template update process to complete, the latest version of the template or interface model must have the **Usability State** *Approved* or *Deprecated*.

Update Window

The following figure shows an example of the **Update** window in a scenario where the **Update** command was used on the *MyMotor* (1.0.0) control module template in the **Global Templates Explorer**. The template references *MyMotor_UC* (1.0.0). This *UC* composite template references *MyDEVCTL_UC* (1.0.0), which in turn references the *MyDEVCTL_UL* (1.0.0) facet template. *MyDEVCTL_UL* versions 1.0.1 and 1.0.2 exist in the Global Templates Library.



Item	Description
1	Template for which a later version was found in the Global Templates Library compared to the version that is currently used.
2	Templates of which a later version is automatically created because the version of their child template is being incremented.
3	Shows the hierarchy of the templates that will be updated and the templates of which a later version exists in the Global Templates Library. Other child templates whose version is unchanged are not shown. At the top level, identifier of the template from which you have selected the Update command.
4	Type of the template.
5	Usability state of the current version of the template after the new version is created based on the value of the State strategy parameter. You can select it individually for templates of which a new version is created.
6	Current version of the template from which you have selected the Update command and current version of its child templates.

Item	Description
7	<p>Version of the template after the update and latest version of child templates with the usability state <i>Approved</i> that were found in the Global Templates Library.</p> <p>For parent templates, the version is created during the update based on the value of the Version strategy parameter.</p> <p>NOTE: You can modify the version number (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) manually for each template version that is created during the update. Modifying the version number manually sets the Version strategy parameter to Custom. You cannot enter the same value as in Version.</p> <p>NOTE: If Version for a template that is updated is already 99.99.9999, New Version is blank and you need to enter a value manually.</p>
8	<p>Allows you to select the usability state of the current version (Version column) of the templates after a new version was created.</p> <p>Default value: Custom</p>
9	<p>Allows you to select the usability state of the new version of the templates after the update.</p> <p>Default value: Approved</p> <p>NOTE: The usability state applies only to new template versions that EcoStruxure Process Expert creates.</p>
10	<p>Allows you to select the versioning scheme (see <i>EcoStruxure Process Expert, Global Templates, Reference Manual</i>) for new template versions that EcoStruxure Process Expert creates.</p> <p>Default value: New Build</p> <p>NOTE: The version strategy applies only to new template versions that EcoStruxure Process Expert creates.</p>

Updating Global Templates in the Global Templates Explorer

Updating Schneider Electric or user-created templates may affect the function of these templates and must be performed by qualified personnel. Before proceeding, refer to the [Overview](#), page 990.

To update Global Templates, proceed as follows.

Step	Action
1	<p>In the tree view of the Global Templates Explorer, right-click the template that you want to update and select Update.</p> <p>Result: The Update window opens.</p>
2	<p>For the current template version and its elements, select the usability state strategy from the menu of the State strategy parameter.</p> <p>NOTE: If you select Custom, select the usability state for each template in the State column.</p>
3	<p>For the new template version and its elements, select the usability state strategy from the menu of the Usability State parameter.</p>

Step	Action
4	<p>For the new template version and its elements, select the version strategy from the menu of the Version strategy parameter.</p> <p>NOTE: You can modify the version number manually for each template that EcoStruxure Process Expert will create by double-clicking the New Version field:</p> <ul style="list-style-type: none"> • If the version number that you have entered is invalid, the default versioning scheme is applied for new templates. • To undo your version change, select New Build as the version strategy. This changes the version of new templates to the default value.
5	<p>Click OK.</p> <p>Result:</p> <ul style="list-style-type: none"> • The update is performed. • The new templates that are created are displayed in their respective folders within the Global Templates Library. • Completed is displayed in the notification panel when the update process is completed. <p>NOTE: Click Cancel to close the window without proceeding with the update.</p>

Updating Global Templates in Template Editors

Updating Schneider Electric Global Templates or user-created templates may affect the function of these templates and must be performed by qualified personnel. Before proceeding, refer to the [Overview](#), page 990.

To update the template or interface model that is used by an element of a facet or composite template, proceed as follows.

Step	Action
1	<p>In the template editor, right-click the element whose template or interface model you want to update and select Update.</p> <p>Result: If a later version of the same template or interface model with the required usability state exists in the Global Templates Library, the binding information is processed; otherwise, an information that the template is already up-to-date is displayed.</p>
2	<p>Result: If any binding cannot be recreated, the Replace Conflicts dialog box opens; otherwise, the update of the template completes and bindings are recreated.</p>
3	<p>Verify the information that is displayed in the Replace Conflicts dialog box and click:</p> <ul style="list-style-type: none"> • Yes: Proceeds with the update of the template, discards conflicting bindings, and recreates the other bindings. • No: Closes the Replace Conflicts dialog box without updating the template.
4	Recreate bindings as needed.
5	Save the change, page 1035.

NOTE: To undo a template update that you have performed in a template editor, you can use the **Replace** command, page 1060.

Updating Global Templates at the Folder Level

Overview

The **Update** command that appears in the context menu of folders of the **Global Template Explorer** lets you create a new version of templates contained in a folder, and optionally, in its subfolders.

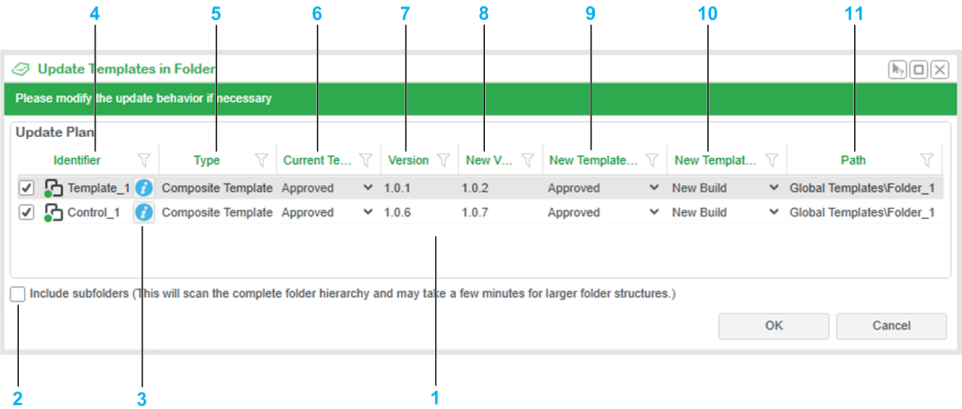
Restrictions When Updating Schneider Electric Templates

Restrictions related to the identifier (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) of the parent template and its elements apply when you update a Schneider Electric template.

When you use the **Update** command at the folder level and the folder contains a template whose identifier starts with the \$ prefix, the template is not updated. You need to edit the template and save it with a new identifier or use the **Duplicate** command, page 1065.

Update Templates in Folder Window

The following figure shows an example of the **Update Templates in Folder** window.



Item	Description
1	List of templates in the folder for which a later version of at least one child template is available in the Global Templates Library. NOTE: If a folder contains several versions of the same template and more than one can be updated, only the highest version will be shown and can be updated. This applies also If some of the versions that can be updated are in a subfolder and you have selected to include subfolders; only the highest version across all folders will be shown.
2	Lets you include templates located inside subfolders. Their location is indicated in the Path column.
3	Click the Information button to view details about which newer version child templates will be used to update the template.
4	Identifier of the templates that are candidate for the update. The operation is performed only on templates whose checkbox is selected.
5	Type of the template.
6	Usability state of the current version of the template after the new version is created.
7	Current version of the template.
8	Version of the template after the update. The version is based on your selection for New Template Version strategy .

Item	Description
	<p>NOTE: You can modify the version number (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) manually. This sets New Template Version strategy to <i>Custom</i>. You cannot enter the same value as it appears in Version.</p> <p>NOTE: If Version of a template to be updated is 99.99.9999, New Version is blank and you need to enter a value manually.</p>
9	<p>Allows you to select the usability state of the new version of the template after the update.</p> <p>Default value: <i>Approved</i></p>
10	<p>Allows you to select the versioning scheme (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) for the new template version.</p> <p>Default value: <i>New Build</i></p>
11	Location of the template in the Global Templates Library.

Updating Templates at the Folder Level

To update the templates contained in a folder, proceed as follows.

Step	Action
1	<p>In the Global Template Explorer, right-click a folder and select Update.</p> <p>Result: The Update Templates in Folder window opens and displays the templates for which a child template with a later version exists in the Global Templates Library.</p> <p>NOTE: The command is not available from the Global Templates Library root folder.</p>
2	Select to display also templates of subfolders if needed.
3	Select which templates you want to update.
4	<p>For each one, select the appropriate usability states and versioning schemes for the current and new template versions and click OK.</p> <p>Result: The selected templates are updated and appear in the tree view of the Global Templates Library. A message in the notification panel indicates the status of the operation.</p>

Replacing the Template of an Element of a Global Template

Overview

In the **Facet Editor** and **Composite Editor**, the **Replace** context-menu command allows you to replace the template that is used by an element with another template of the Global Templates Library.

You can replace the template of the following types of elements:

- Composite templates
- Facet templates
- Interfaces

You are informed if binding-related conflicts are detected before the replacement of the template is performed.

The replacement of the template of an element impacts templates that reference this element and requires updating, page 1052 all the higher-level templates.

Restrictions When Replacing Schneider Electric Templates

Restrictions (see *EcoStruxure Process Expert, Global Templates, Reference Manual*) related to the identifier of parent templates apply when you replace a template.

Before Starting

Depending on the situation, consider the following before starting:

- To replace the template of an element by the latest version of the same template, use the **Update** command, page 1054 instead.
- You can use the **Replace** command to undo the update of the template of an element, page 1054. Undoing a template update does not restore bindings that could not be recreated.

Replace Dialog Box

For a description of the template browser that opens, refer to the topic describing the **Replace** dialog box, page 1069.

Replacing The Template That Is Used by an Element

Modifying Schneider Electric Global Templates or user-created templates may affect the function of these templates and must be performed by qualified personnel. Before proceeding, refer to the *Overview*, page 990.

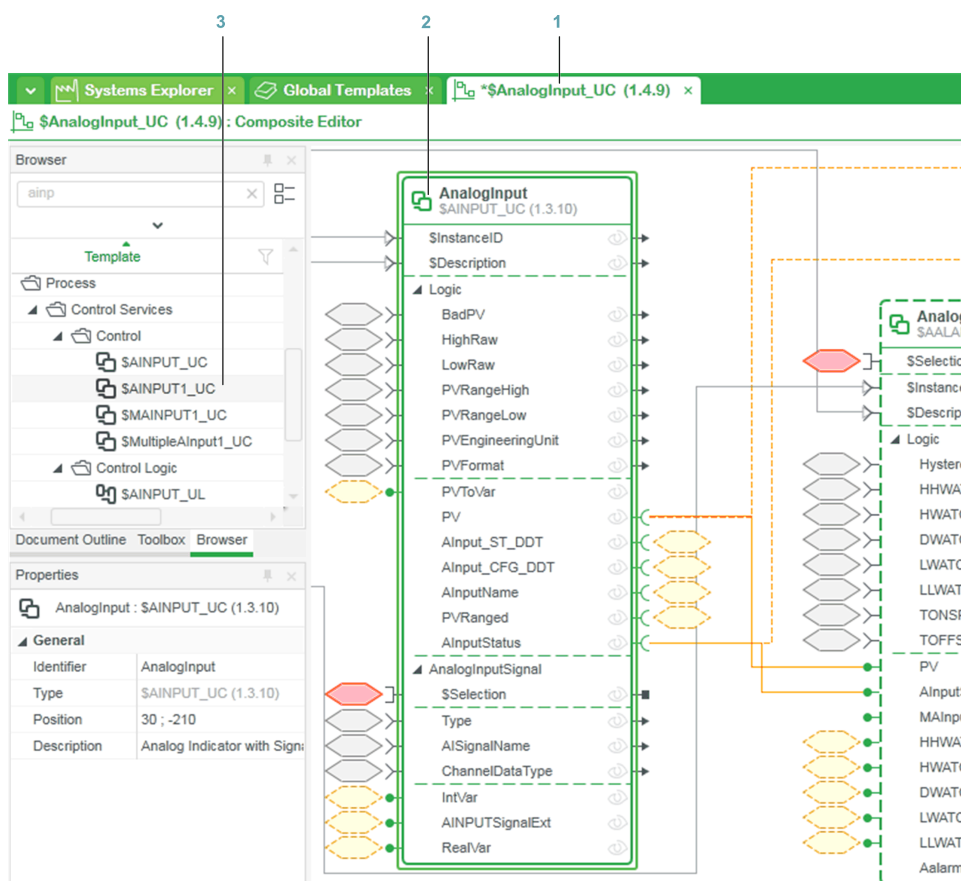
To replace the template that is used by an element of a Global Template, proceed as follows.

Step	Action
1	<p>In a template editor, right-click the element whose template you want to replace and select Replace.</p> <p>Result: The Replace dialog box opens.</p> <p>NOTE: Alternatively, you can select the template in the Browser pane and drag it onto the header of the element until Drop to update the reference with x is displayed in a tooltip (where x represents the name and version of the template that you want the element to use).</p>
2	<p>Select the template that you want the element to use instead and click OK.</p> <p>Result: The Replace confirmation dialog box opens.</p>

Step	Action
3	Verify the information that is displayed and click Yes . Result: If any binding cannot be recreated, the Replace Conflicts dialog box opens; otherwise, the replacement of the template completes and bindings are recreated.
4	Verify the information that is displayed in the Replace Conflicts dialog box and click: <ul style="list-style-type: none"> Yes: Proceeds with the replacement of the template, discards conflicting bindings, and recreates the other bindings. No: Closes the Replace Conflicts dialog box without replacing the template.
5	Recreate bindings as needed.
6	Save changes, page 1035.

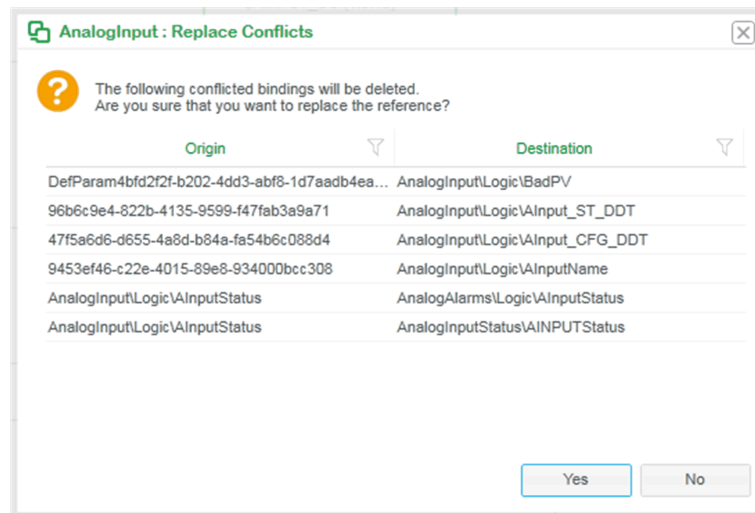
Example

The following example illustrates the replacement of the template of a composite element, which is referenced by a parent composite reference. The highest level composite template being *\$AnalogInput* (control module).

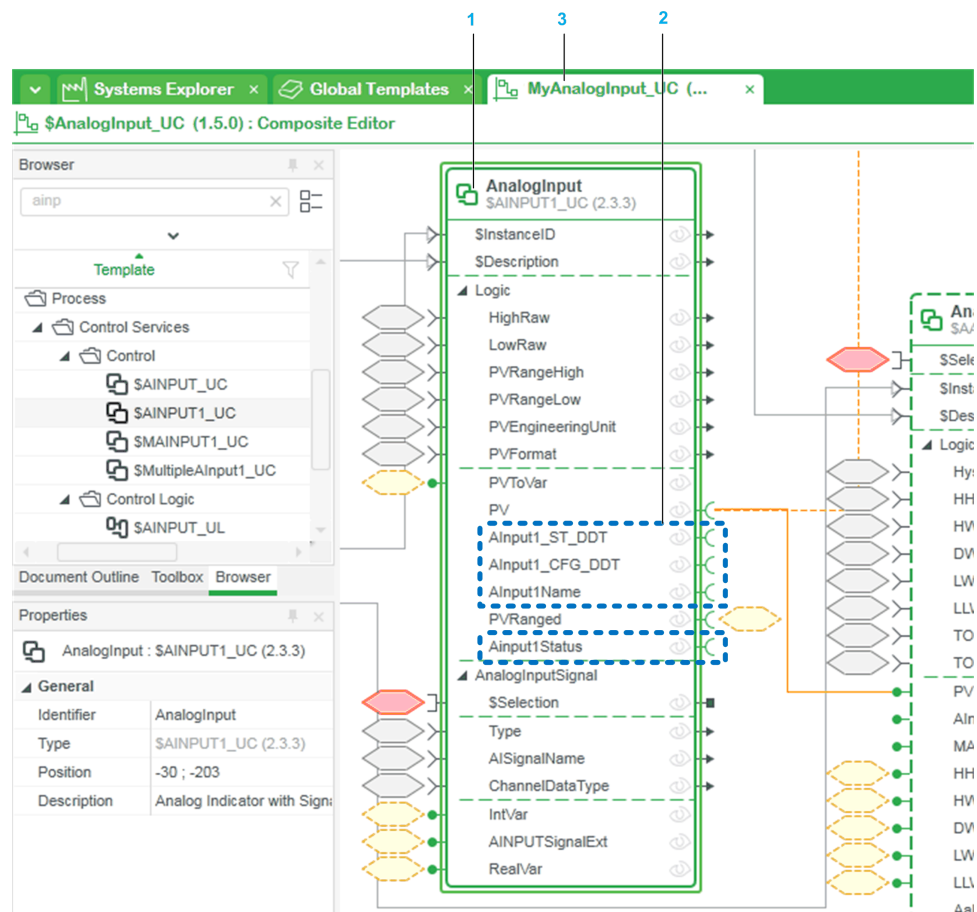


Item	Description
1	<i>\$AnalogInput_UC</i> [1.4.9], which references element <i>AnalogInput</i> is opened in edit mode in the Composite Editor .
2	Element <i>AnalogInput</i> uses template <i>\$AINPUT_UC</i> , which is going to be replaced.
3	Browser pane that allowed locating template <i>\$AINPUT1_UC</i> , which is going to replace <i>\$AINPUT_UC</i> .

The list of bindings that cannot be recreated after the template replacement is displayed. Write down these bindings so that you can recreate them later on if required.

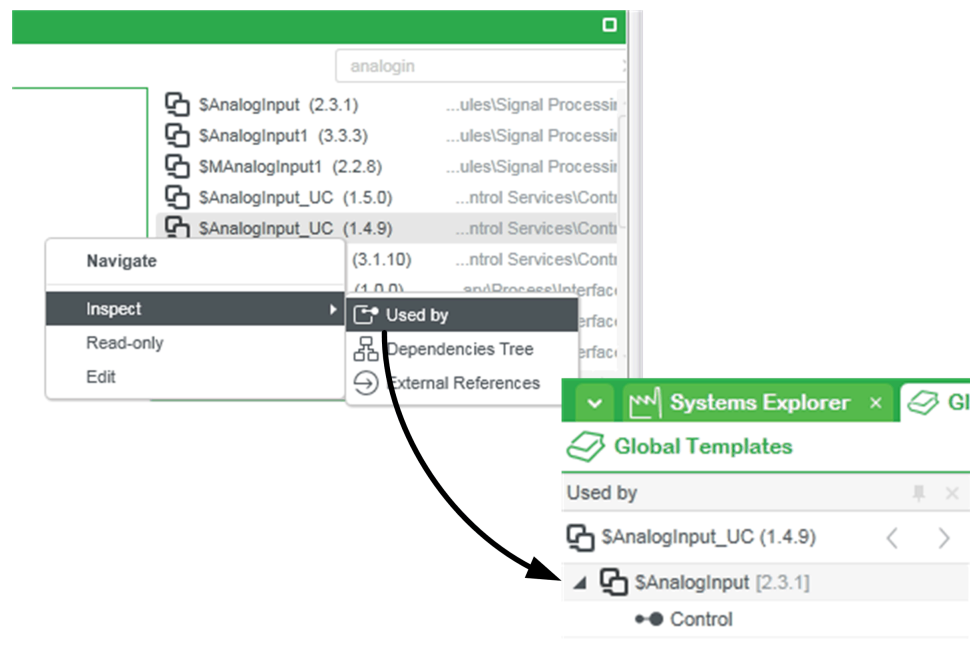


The figure shows the *AnalogInput* element after template replacement is complete and the template referencing it has been saved.

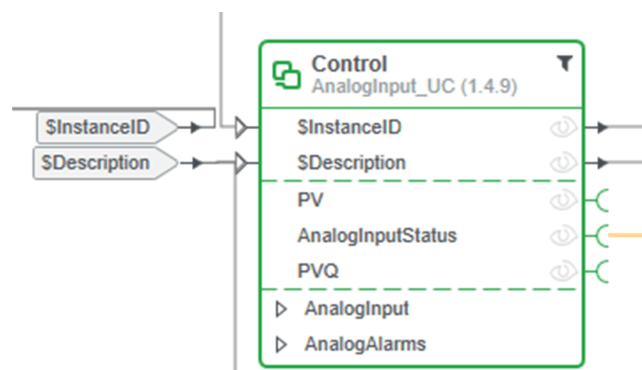


Item	Description
1	Element <i>AnalogInput</i> now uses template <i>\$AINPUT1_UC</i> . Its identifier is unchanged as shown in the element header and the Properties pane.
2	The bindings that were listed in the Replace Conflicts dialog box have not been recreated.
3	<i>\$AnalogInput_UC</i> [1.4.9] has to be saved with a different identifier (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) (for example, <i>MyAnalogInput_UC</i> [1.2.0]) with Usability state approved .

Once template replacement is complete, you need to update templates that reference *\$AnalogInput_UC* [1.4.9] so that they use *MyAnalogInput_UC* [1.2.0] instead. To identify these templates, use the **Used by** command on *\$AnalogInput_UC* [1.4.9]. In this example, the command returns *\$AnalogInput* [2.3.1], which uses *\$AnalogInput_UC* [1.4.9] for element *Control*.



The following figure shows in a partial view, element *Control* referenced by *\$AnalogInput* [2.3.1].



To update *\$AnalogInput* [2.3.1] with *MyAnalogInput_UC* [1.2.0], navigate to *\$AnalogInput* [2.3.1] and use the **Duplicate** command, page 1065. This process creates a copy with a different identifier, which includes *MyAnalogInput_UC* [1.2.0]. In this example, duplicating *\$AnalogInput* [2.3.1] creates *MyAnalogInput* [1.0.0]. (The template that is replaced is highlighted for the example.)

\$AnalogInput (2.3.1) : Duplicate

Please specify the details for the templates to be duplicated

New Identifier
☒ Prefix ☐ Suffix Duplicate All

Identifier and Version	Type	Action	New Identifier	New Version
\$AnalogInput (2.3.1)	Composite Template	Duplicate	MyAnalogInput	1.0.0
\$AnalogInput_CS (1.1.7)	Composite Template	None		
\$AnalogInput_CD (1.1.8)	Composite Template	None		
\$AALARM_CD (2.2.1)	Data	None		
\$AINPUT_CD (2.1.2)	Data	None		
\$AnalogInput_CG (1.0.4)	Composite Template	None		
\$AIIPVA_CG (1.0.3)	Genie	None		
\$AIIPVSPA_CG (1.0.2)	Genie	None		
\$AIPVA_CG (1.0.2)	Genie	None		
\$AIPVSPA_CG (1.0.2)	Genie	None		
\$AnalogInput_UC (1.4.9)	Composite Template	Replace	MyAnalogInput_UC	1.2.0
\$AALARM_UC (2.2.5)	Composite Template	None		
\$AALARM_UL (2.2.1)	Logic	None		
\$AINPUT_UC (2.0.10)	Composite Template	None		

☐ Display interfaces

Description of the change
 Template duplicated from the template AnalogInput (2.3.1)

Usability State
 Approved

OK Cancel

NOTE: You may need to update application instances that currently use `$AnalogInput` [2.3.1] and/or `$AnalogInput_UC` [1.4.9] to use the new templates `MyAnalogInput` [1.0.0] and/or `MyAnalogInput_UC` [1.2.0] respectively. Use the **Replace** command, page 220 to perform the update.

Duplicating Global Templates

Overview

The **Duplicate** command that is available in the context-menu of templates in the **Global Templates Explorer**, lets you create a copy or a new version of any template with the original and/or modified child templates.

For each child template, you can select to keep the original, duplicate (rename or change the version), or replace it by another existing template.

You are informed if binding-related conflicts are detected because of a template replacement before the template is duplicated.

NOTE: This command can be used instead of the **Update** command for templates whose identifier contains the \$ prefix. However, you need to know for which child templates a new version already exists and select it manually.

Restrictions When Duplicating Schneider Electric Templates

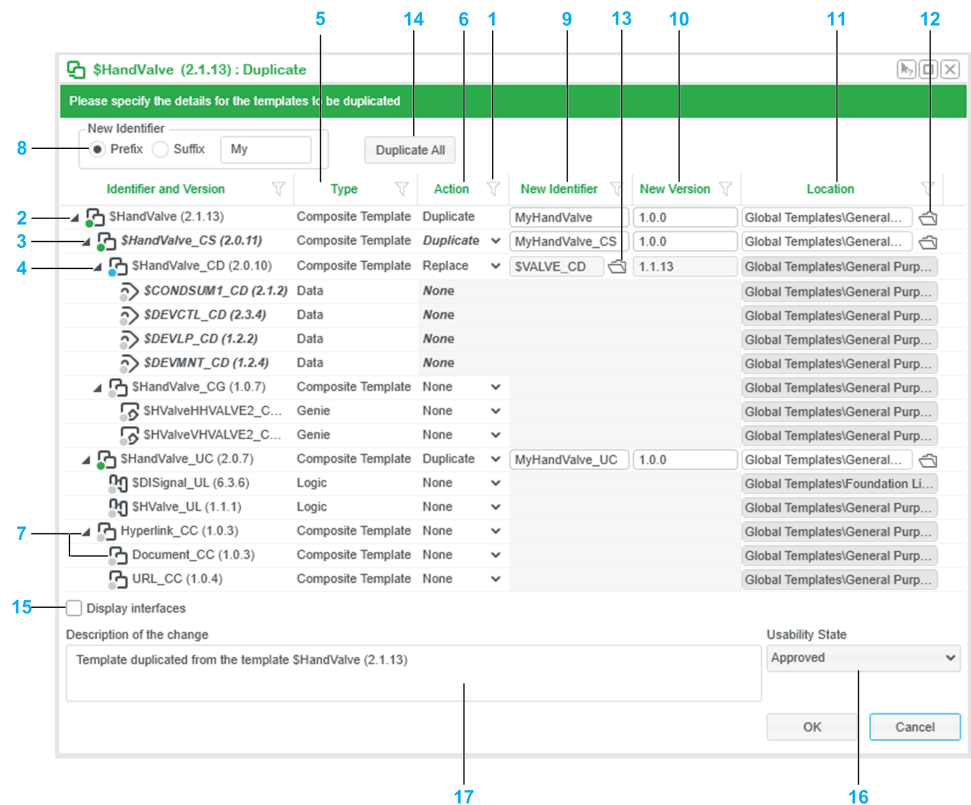
Restrictions (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) related to the template identifiers apply when you duplicate a Schneider Electric template .

Good Practices

- Use the **Duplicate** command as a one-step solution to create a copy of a Schneider Electric template whose child templates you can edit and save without using the \$ prefix (see EcoStruxure™ Process Expert, Global Templates, Reference Manual).
- Avoid duplicating templates of the Foundation Library and interface models because these are low-level resources that are common to all Schneider Electric templates to make them widely compatible. You can identify foundation templates by their location.

Duplicate Window

The following figure shows an example of the **Duplicate** window for a *\$HandValve* sample template in which duplication and replacement actions for child templates have been selected.



Item	Description
1	You can sort and filter, page 120 data.
2	Identifier and version of the template from which you have selected the Duplicate command.
3	Identifier and version of child templates that the template references. At each level of the template composition, each template is shown only once even if several instances of it exist. By default, interface models are not shown.
4	When you replace a template, the child templates of the replacement are shown under the template to be replaced in read-only mode.
5	Type, page 38 of the template.
6	Action to be performed for each referenced child template. Each type of action is identified by a colored dot: <ul style="list-style-type: none"> Duplicate (green): Creates a new template. If the template references child templates, they are reused without change unless you duplicate all or some of them as well. None (gray): No action is performed. The template is reused in the duplicated parent template. For replacement templates and their children, it is the only setting that is allowed. Replace (blue): Opens the Replace dialog box, page 1069, which lets you select an existing template in the Global Templates Library to be used in place of the current template and its child templates. The replacement template is used for all instances of the current template. The template must have the usability state (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) Approved.
7	Duplicating or replacing a template automatically sets the action of all parent templates to Duplicate . Entries in italic indicate a change that is the result of an action that you have selected.

Item	Description
8	Free form text fields to enter either a prefix or suffix that is added to the identifier of each duplicated template.
9	Identifier of the template that is proposed for duplicated templates (editable) or identifier of the replacement template that you have selected.
10	Version that is proposed for duplicated templates (editable) or version of the replacement template.
11	Location in the Global Templates Library of: <ul style="list-style-type: none"> An original template (None action). The location is read-only. The duplicated template. You can select a different location and/or add additional ones. The location that you select must exist. When you select more than one location, linked copies, page 1038 of the template are created. In the location selection dialog box, a check box lets you set the same location for all duplicated templates. The replacement template. The location is read-only. Child templates of a replacement template (None action). The location is read-only
12	Button to modify the location of duplicated templates within the Global Templates Library.
13	Button to open the Replace dialog box, page 1069 to select a different template.
14	Sets the action of all child templates to Duplicate . Applies to interfaces only if Display interfaces was selected when the Duplicate All button was clicked. This action overrides changes that you have made for child templates for which you have selected the Replace action. NOTE: You cannot revert this action for all templates at once unless you cancel and discard changes.
15	When selected, shows the interfaces that are referenced by child templates. You can select an action for each interface. The changes that you have made to interfaces are retained when you clear the check box. When you select the Duplicate command for an interface, select it to show nested interfaces.
16	Lets you select the usability state (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) that is set for all duplicated templates.
17	Free-form text field to enter a description of the changes.

Duplicating Global Templates

Duplicating Schneider Electric Global Templates or user-created templates may affect the function of these templates and must be performed by qualified personnel. Before proceeding, refer to the [Overview](#), page 990.

To duplicate a global template, proceed as follows.

Step	Action
1	In the tree view of the Global Templates Explorer , right-click the template that you want to duplicate and select Duplicate . Result: The Duplicate window opens.
2	Select an action for referenced child templates as needed.
3	Configure the parameters for duplicated templates as needed.
4	Select a usability state for duplicated templates.
5	Click OK . Result: If no binding-related conflicts are detected, the template is duplicated based on your selections; otherwise, the Replace Conflicts dialog box opens.

Step	Action
6	Verify the information that is displayed in the Replace Conflicts dialog box and click: <ul style="list-style-type: none">• Yes: Proceeds with the duplication of the template and discards conflicting bindings.• No: Closes the Replace Conflicts dialog box and reverts to the Duplicate window.
7	Recreate bindings as needed.

Replace Dialog Box

Overview

The **Replace** dialog box opens when you select the corresponding command in the **Duplicate** window, page 1065 or in the context menu of an element in a template editor, page 1060.

It lets you browse the Global Templates Library to select a replacement template.

Description

The browser of the **Replace** dialog box operates like the templates browser, page 993 of the **Global Templates Explorer** with the following exceptions:

- The scope of the search is context-sensitive.
- Facet templates are shown by default.
- Only templates with the usability state **Approved** are shown.

Editing and Extending Interfaces In-Place

Overview

The **Edit/Extend Interface** command lets you edit, add, and remove elementary elements of an interface that is referenced in a template without the need to edit the interface by using the **Interface Editor**.

You can add elements in direct mode only.

The operation creates a new version of the interface and updates the reference. It also allows you to update other references of this interface (any role) that exist in this template.

The command is not available for deferred interfaces and when the interface element contains a nested interface.

NOTE: The update of an interface reference impacts the other role of this interface and requires updating, page 1052 the templates that reference the previous version of the other role. Refer to the example, which describes the workflow, page 1072.

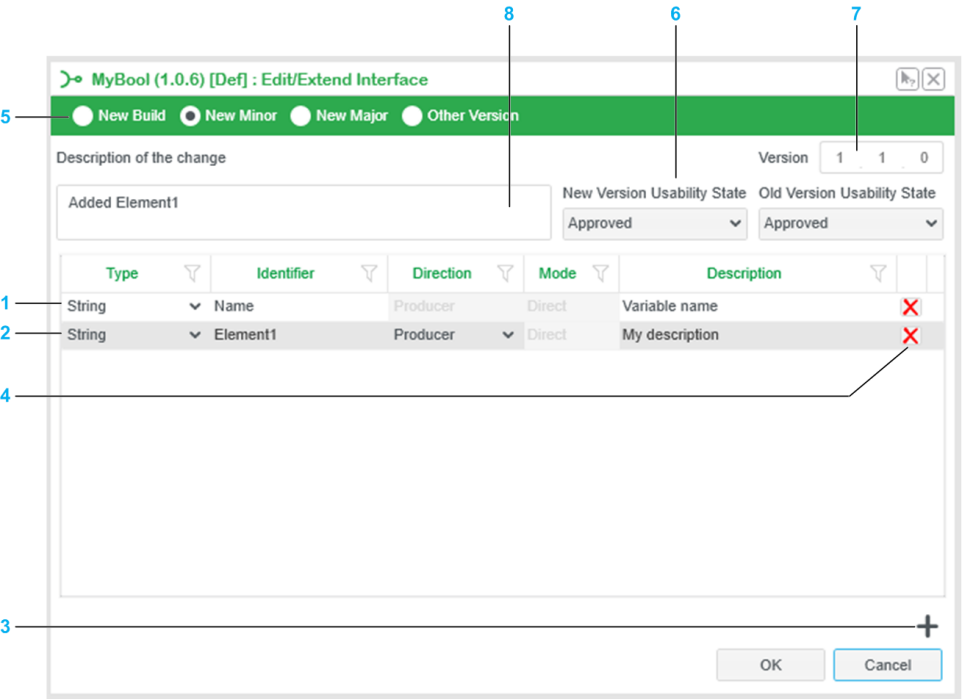
Restrictions When Updating Schneider Electric Templates

Restrictions (see EcoStruxure™ Process Expert, Global Templates, Reference Manual) related to the identifier of the parent template and its elements apply when you update a Schneider Electric template.

You cannot use the **Edit/Extend Interface** command on an interface whose identifier starts with the \$ prefix. You need to edit the interface from within the template and save it with a new identifier or use the **Duplicate** command, page 1065.

Edit/Extend Interface Window

The following figure shows an example of the **Edit/Extend Interface** window.



Item	Description
1	<p>Element that was originally part of the interface.</p> <p>Once an element has been added and the interface saved, you cannot change its Direction and Mode properties anymore.</p> <p>NOTE: Modifying Type does not discard existing bindings. However, modifying Identifier does.</p>
2	<p>Element that is being added to the interface by using the Edit/Extend Interface command.</p> <p>The Mode interface property (see EcoStruxure™ Process Expert, Global Templates Reference Manual) is read-only; only direct data propagation is supported when you use the command.</p>
3	Button to add a new element.
4	<p>Button to remove the element.</p> <p>NOTE: If you remove a bound element and recreate it with identical identifier and configuration, the binding with this element is maintained when you update the interface.</p>
5	<p>Lets you select the versioning scheme that you want to use for the new version of the interface.</p> <p>Select Other as version scheme to edit the value.</p>
6	<p>Lets you select the following:</p> <ul style="list-style-type: none"> Usability state that the new version of the interface will have. <p>NOTE: Only <i>Approved</i> or <i>Deprecated</i> allow using the new version of the interface right away in the template.</p> <ul style="list-style-type: none"> Usability state of the interface (old) that you are editing/extending after you create the new version of interface.
7	Version of the interface that you are creating.
8	<p>Mandatory change description.</p> <p>The description is visible in the changes log of the interface.</p>

Editing/Extending Interfaces

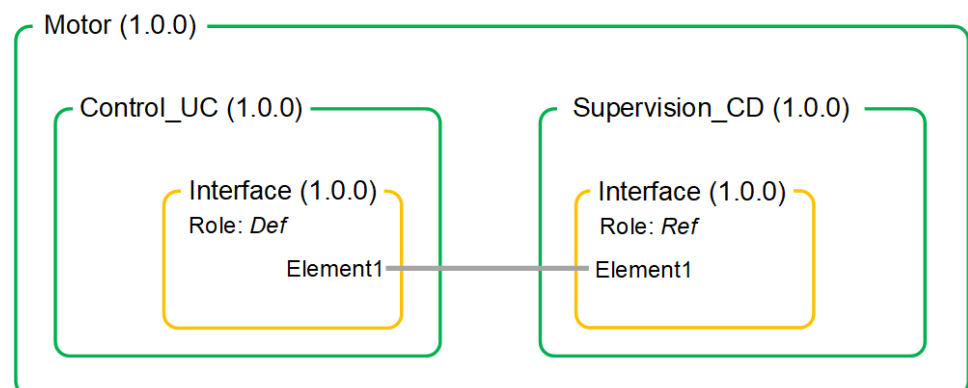
To edit/extend an interface from within its parent template, proceed as follows.

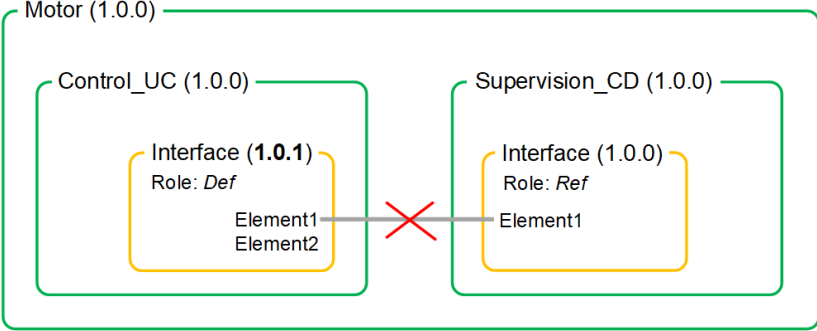
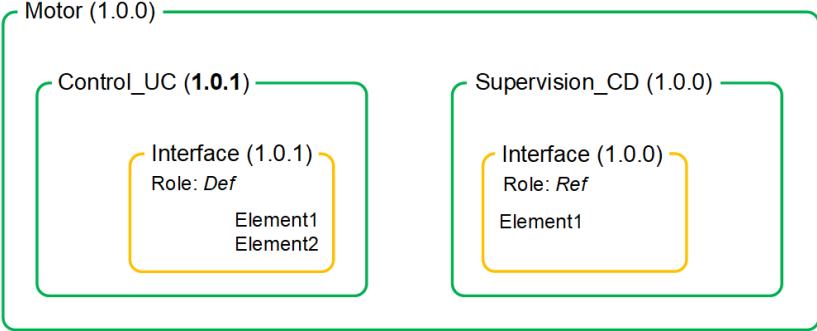
Step	Action
1	in the Facet Editor or Composite Editor , right-click the header of an interface element and select Edit/Extend Interface .
2	In the Edit/Extend Interface window, perform the required changes, enter a change description, and click OK . Result: The window closes and a new version of the interface with your configuration is created. NOTE: If New Version Usability State is <i>Not Approved</i> or <i>Obsolete</i> , a new version of the interface is created but the interface role element is not replaced in the template.
3	If other references (any role) of this interface exist in this template, the Update References dialog box, page 1052 opens. Click either button: <ul style="list-style-type: none"> Yes: Updates all references of the interface in the template if existing bindings can be recreated. No: Only updates the interface that you have edited if existing bindings can be recreated.
4	If in the new interface version at least one element that was bound has been removed, a Replace Conflicts dialog box opens in sequence for each reference that is updated. In each dialog box, verify the information that is displayed about bindings that cannot be recreated and click either button: <ul style="list-style-type: none"> Yes: Proceeds with the replacement of the reference without recreating bindings with elements that are removed. No: Does not replace the reference.
5	Save changes in the template that you are editing.
6	Update the other templates that reference other role of each interface reference. You can identify these templates by using the Inspect > Used By context menu command on the previous version of the interface in the Global Templates Explorer or using the Update command at the folder level.

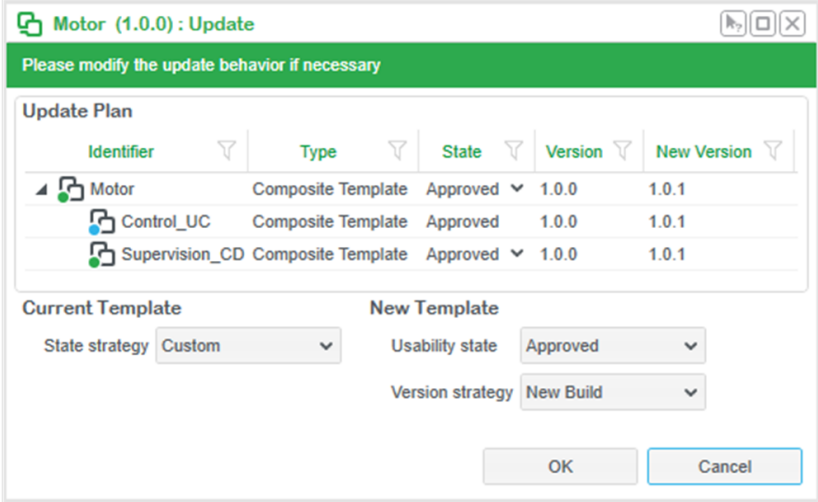
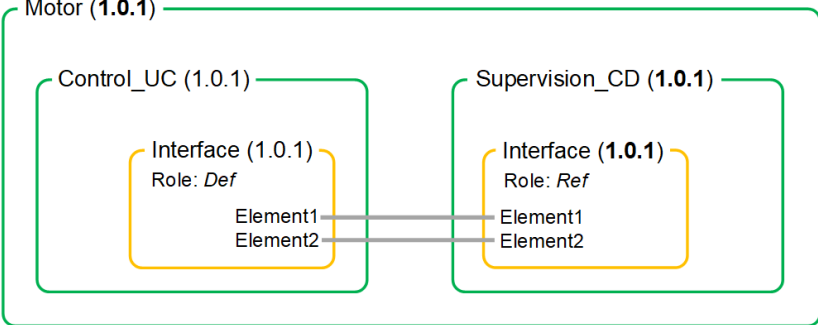
Workflow When Editing/Extending an Interface in a Control Module Template

This following table describes the workflow to propagate changes that were made to an interface by using the **Edit/Extend Interface** command to the entire composition of the template referencing this interface. A simple template is used as example.

The control module template *Motor (1.0.0)* contains two child templates. Each of them references a role of *Interface (1.0.0)*. The interface consists of one element to propagate data.



Step	Action
1	<p>From the Global Templates Explorer, edit the template that contains either role of the interface.</p> <p>(In this example, <i>Control_UC</i>. However, you can also edit <i>Supervision_CD</i> instead. The overall workflow remains the same.)</p>
2	<p>Select the Edit/Extend Interface command on the interface and modify it.</p> <p>(In this example, <i>Interface</i> and adding <i>Element2</i>.)</p>
3	<p>Save changes to this role of the interface with a new version and usability state <i>Approved</i>.</p> <p>(In this example, <i>Interface (1.0.1)</i>, role <i>Def</i>.)</p> <p>NOTE: Do not rename the interface because this will prevent the propagation of the changes throughout the template. If the identifier of the interface starts with the \$ prefix, use the Duplicate command, page 1065 to remove it first.</p>  <p>NOTE: At this point, data is not propagated anymore to other templates referencing the original version of other role of the interface (In this example, to <i>Supervision_CD (1.0.0)</i> referencing role <i>Ref</i> but of <i>Interface (1.0.0)</i>).</p>
4	<p>Save the template (Save As) that contains the modified interface role with a new version, usability state <i>Approved</i>, and close it.</p> <p>(In this example, <i>Control_UC (1.0.1)</i>.)</p> 

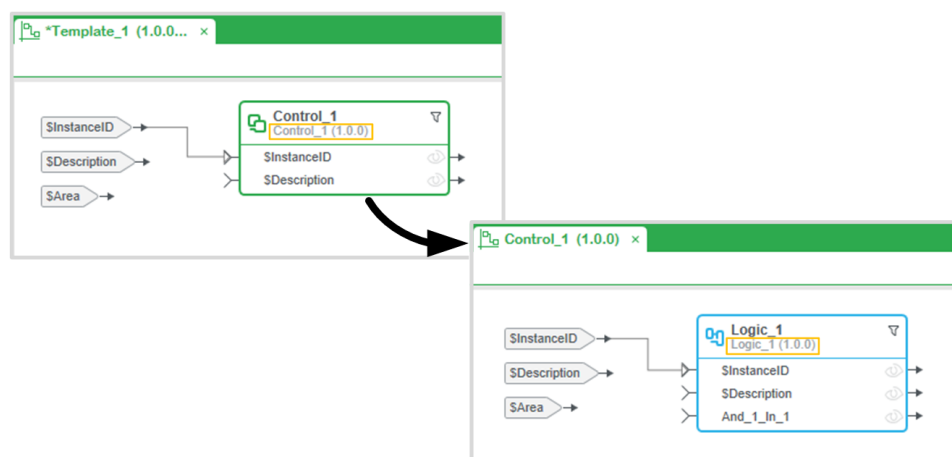
Step	Action
5	<p>In the Global Templates Explorer, right-click the highest level template that contains the interface and its parent and select Update.</p> <p>(In this example, <i>Motor (1.0.0)</i>, which contains <i>Interface</i> and its parent <i>Control_UC (1.0.0)</i>. However, in your template, it could be several levels up.)</p> <p>Result: The Update window opens and lists, among other templates, all the child templates that contain a reference of either role of the original interface version, which will be updated to the new version containing your changes.</p> <div></div> <p>NOTE: The interface that you have modified does not appear.</p>
6	<p>In this example, clicking OK creates <i>Motor (1.0.1)</i>, which is updated with the following:</p> <ul style="list-style-type: none">Existing <i>Control_UC (1.0.1)</i> (which already contains <i>Interface (1.0.1)</i>, <i>Def</i>)<i>Supervision_CD (1.0.1)</i> created by EcoStruxure Process Expert (in which <i>Interface (1.0.0)</i>, <i>Ref</i> is updated to <i>Interface (1.0.1)</i>, <i>Ref</i>) <p>As a result, the binding between <i>Element1</i> is automatically re-established and a new binding between <i>Element2</i> created.</p> <div></div>

Updating References in Parent After Editing Child Element – Example

This topic describes the update of references inside the parent template by using a simple three-level template as an example. It shows how to update the parent after the lowest-level template has been modified.

Starting Point – Parent Template

To illustrate the automatic update process, [page 1052](#), this example uses parent *Template_1* version 1.0.0, which references composite *Control_1* version 1.0.0, which references facet *Logic_1* version 1.0.0.



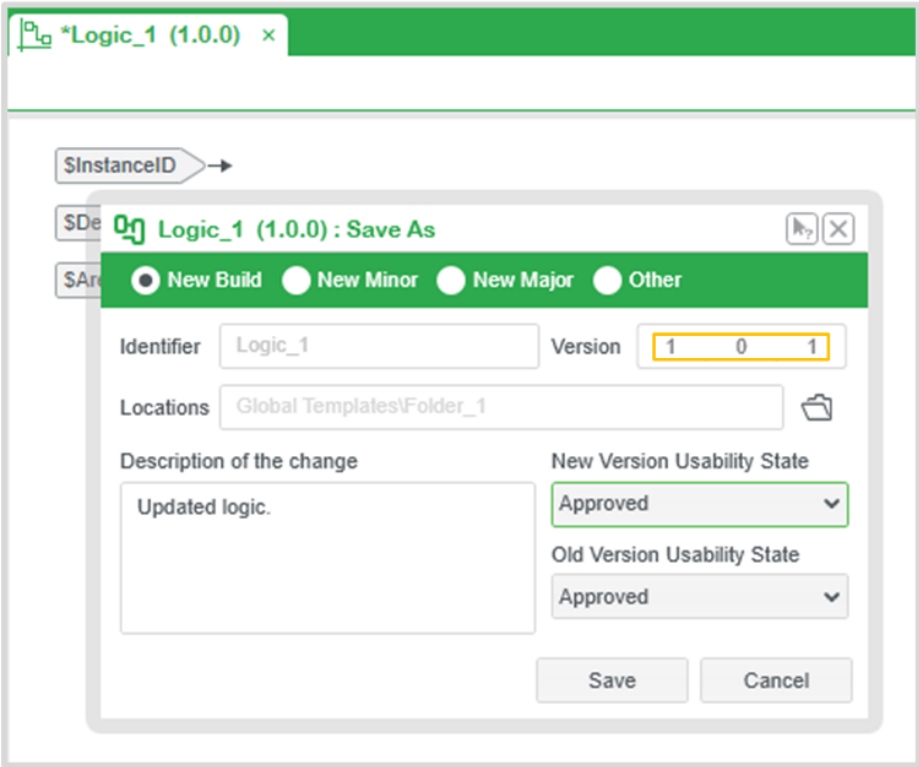
NOTE: Template identifiers and versions are highlighted with an orange outline for illustration purposes only.

Modifying the Lowest-Level Template and Saving Changes

Template_1 is edited.

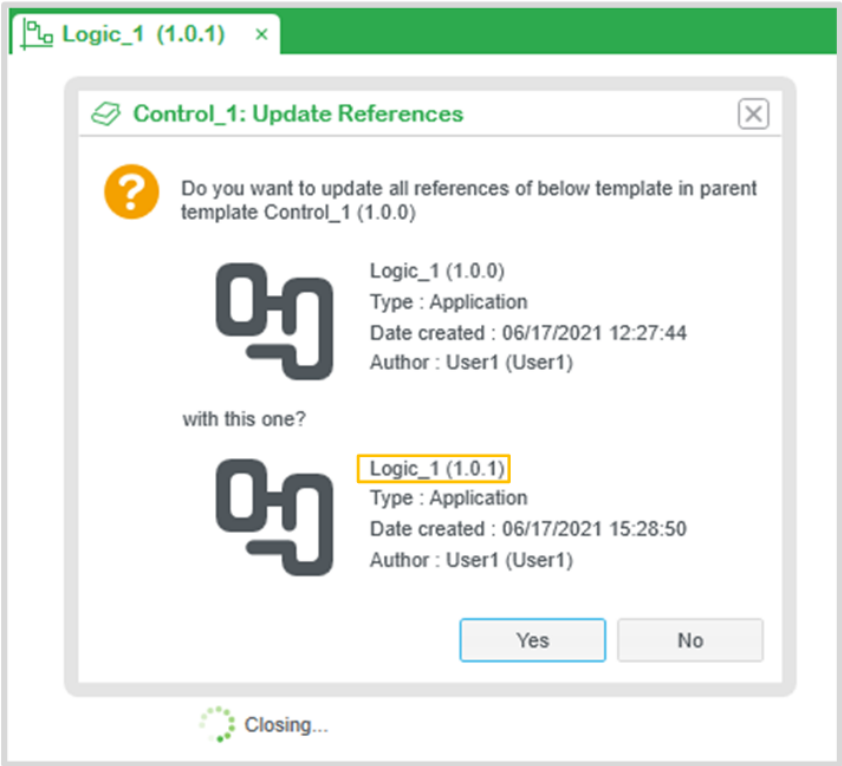
From within *Template_1*, *Control_1* is edited (by using the context menu command). All templates must remain open.

Then, from within *Control_1*, *Logic_1* is edited, modified, and saved with a new version (1.0.1) and usability state *Approved*.



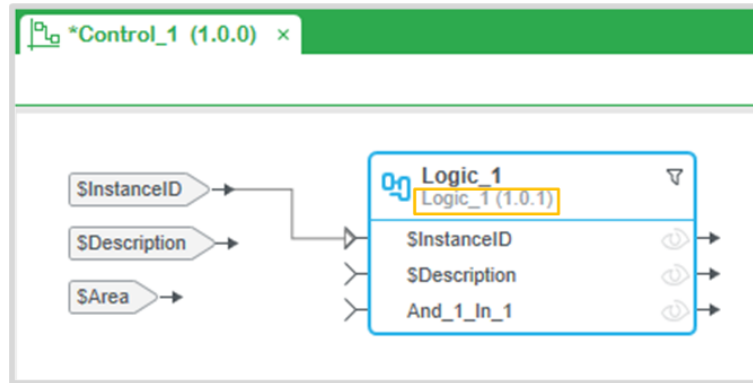
Closing the Lowest-Level Template and Updating the Reference in Parent (Mid-Level Template)

After closing *Logic_1*, the **Update References** dialog box opens, which lets you update reference *Logic_1* v1.0.0 with version 1.0.1 inside *Control_1*.



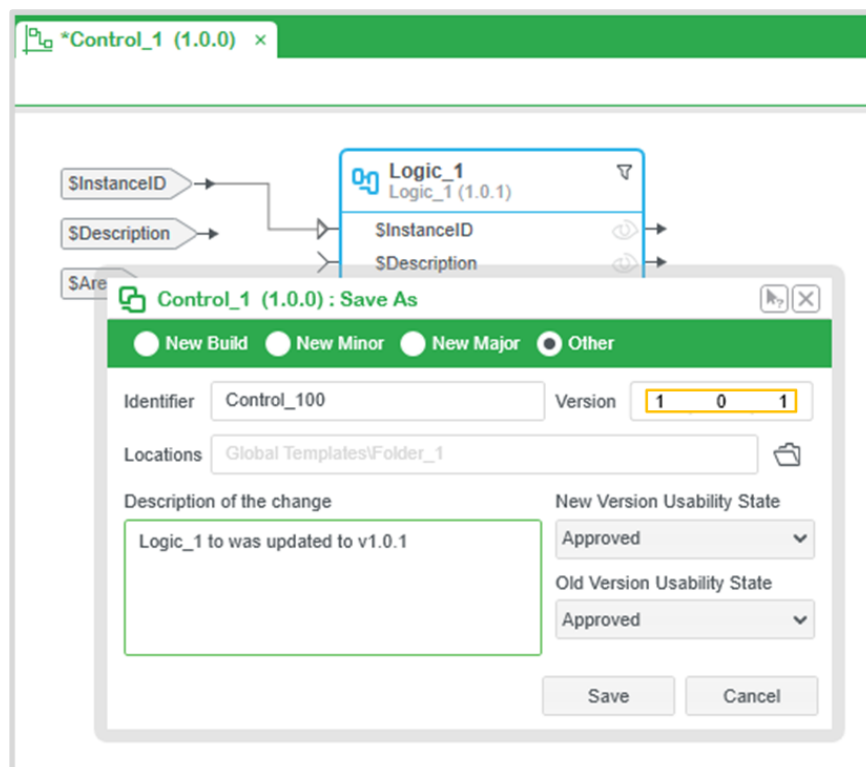
NOTE: If *Control_1* contains several references of *Logic_1* v1.0.0, clicking **Yes** replaces them all. Bindings are managed by EcoStruxure Process Expert and a **Replace Conflicts** dialog box opens for each reference for which at least one binding cannot be recreated. You can select not to proceed with the update.

The following figure shows the result in *Control_1* after clicking **Yes** in the dialog box.



Saving Changes in Mid-Level Template

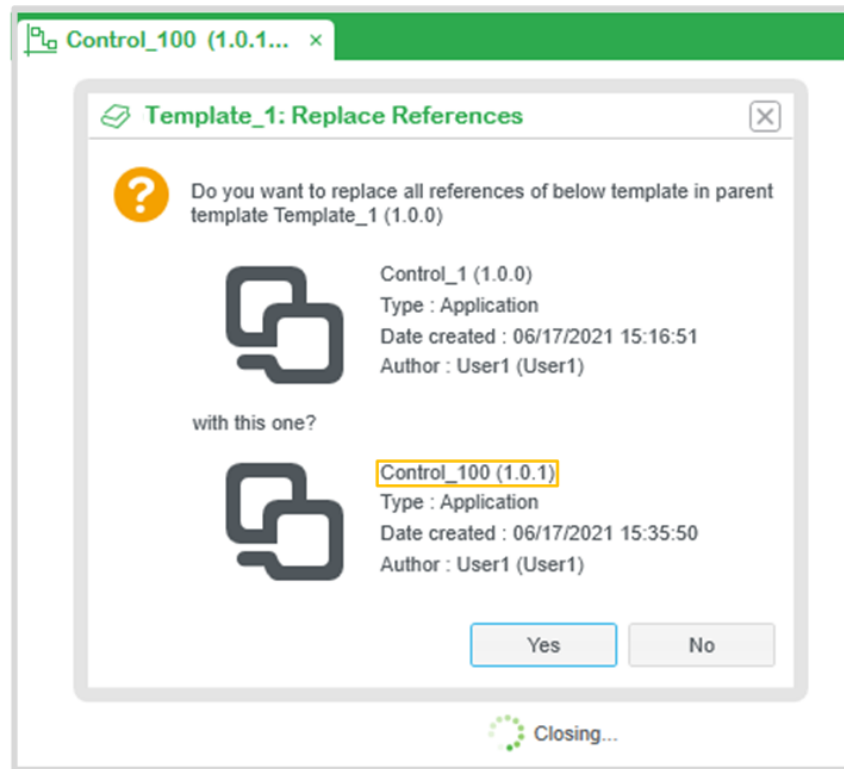
To apply the update of *Logic_1*, *Control_1* needs to be saved with a new version (1.0.1) and usability state *Approved*. In this example, *Control_1* is renamed *Control_100* to show the impact of renaming a template when saving it.



NOTE: You can perform additional modifications in *Control_1* before saving it.

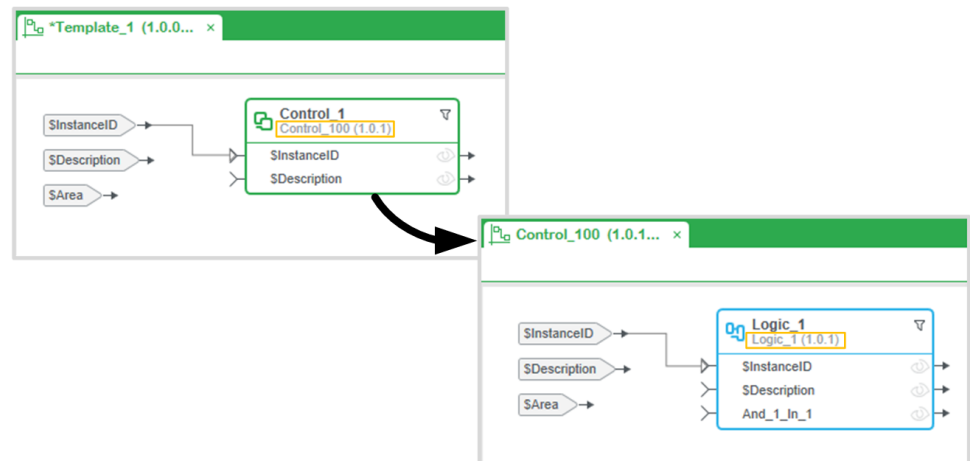
Closing the Mid-Level Template and Updating the Reference in Parent

After closing *Control_100*, the **Replace References** dialog box opens, which lets you *replace* reference *Control_1* v1.0.0 with *Control_100* version 1.0.1 inside *Template_1*.



Result of the Update Process

The following figure shows the result in *Template_1* after clicking **Yes** in the dialog box. The modifications in *Logic_1* and *Control_100* have been propagated to the highest level. To finalize the update process, *Template_1* needs to be saved.



EcoStruxure Process Expert Services

What's in This Part

System Documentation.....	1080
Managing the Consistency of the Topology.....	1090
System Monitoring	1107
Basic Diagnostics.....	1118
Content Repository.....	1119

Overview

This part describes the system-wide and domain-specific services that are provided to facilitate the engineering of systems.

System Documentation

What's in This Chapter

About System Documentation 1081

System Documentation Content 1083

Generating System Documentation 1089

About System Documentation

Scope of System Documentation

You can generate, page 1089 system documentation that contains information on either one:

- A system (all domains)
- One or more domains of a system. The domains are:
 - **Application**
 - **Projects** (Control and Supervision)
 - **Topology**
 - **Global Templates**

The documentation encompasses the detailed engineering data of a system or its domains including:

- Instance custom configuration.
- Application hierarchy.
- Control and Supervision project content (includes Participant-generated documentation), mappings, properties, and settings.
- Topological entity configuration and physical connections.
- All templates of the Global Templates Library or only templates that are used by objects of the system.

Details on how to configure the documentation to obtain the desired information are given in the following pages.

Documentation Layout

You can customize the information that appears in the documentation as well as the layout in the first **Generate Report** window. You can also preview the documentation cover page.

The following are always included in a system documentation .

Information	Comment
Product name	Name of the software that is used to engineer the system. Appears on each page of the documentation. For example, <i>EcoStruxure Process Expert Report</i> .
Identifier of the system	For example, <i>System_1</i> . Appears on the first page of the documentation.
Schneider Electric logo	Appears in the header and footer of each page.
Creation date and time	Present computer date and time.
Blank pages	Five blank pages are added at the end of the documentation for your own use.

The following information and documentation properties can be customized.

Information	
Header and footer	Customizable. Free-form text. Appear on each page of the documentation.
Page orientation and size	Applies to all pages of the documentation.
Documentation format	PDF or Microsoft Word.
Documentation file name and location	Default values: <ul style="list-style-type: none"> Name: <i><System identifier></i> Location: The Windows Documents folder of the user logged in to the computer.
Enduser details	<ul style="list-style-type: none"> Customer name and site name: Free-form text. Appear on the cover page only. Company logo: Appears in the header of each page. Accepted formats: JPG, PNG, BMP. The size of images is automatically adjusted to fit the documentation (approximately 350 x 175 pixels).
Documentation description and author	Free-form text. Appear on the cover page only.

System Documentation Content

System Information

System properties are not included in the documentation.

Domain Information

For each domain of a system that you select, a corresponding section appears in the documentation. The following pages describe the content that is included in each section by default and on-demand.

Application Domain Data

Object Selection

The **Application** tab displays the instances and the structure of the applications that exist in the system. You can exclude objects from the documentation by clearing their checkbox.

NOTE: Clearing the checkbox of the root node excludes the entire domain from the documentation.

Default Information

The *Application Domain* section of the documentation contains the following:

- *Folders* table
- *Instances* table

NOTE: Element selections and default parameter values are not shown.

Optional Information

The following table describes the information that is added to the *Application Domain* section when you select the corresponding option.

Optional selection	Description
Include Modified Parameters	Adds the <i>Modified Parameters</i> table. For each instance, lists the parameters with a value that is different from the default value of the template. <i>Parameters</i> column: <ul style="list-style-type: none">• <i>Name</i>: Name of the modified parameter.• <i>Value</i>: Modified value.
Include Interface Links	Adds the <i>Interface Links</i> table. For each instance that is the source of an interface link, it lists the following: <ul style="list-style-type: none">• Source interface• Destination instance• Destination interface
Include Workspaces	Adds the <i>Workspaces</i> table that lists the workspaces that exist in the Application Explorer . For each workspace, it lists the following: <ul style="list-style-type: none">• The instances it contains and the template that the instances use.• Information about the interfaces of instances.

Project Domain Data

Object Selection

The **Projects** tab with its **Control Project** and **Supervision Project** child tabs reproduces the projects that exist in the system.

You can exclude projects from the documentation by clearing their checkbox. Clearing the checkbox of the root node in the Control or Supervision child tab excludes the respective sub-domain from the documentation.

Default Control Project Information

The *Project Domain* section consists of a sub-section for each Control project, which contains the following:

- *Properties* table: Indicates the controller platform of the project.
- *Settings* table: Indicates the settings that are configured for the project.
- *Control Executable* sub-section: A subsection exists for each Control executable of a Control project. It contains the following information:
 - *Service Mappings* and *Communication Mappings* tables: A table for each type of mapping that exists.
 - *Hardware Mappings* table: A table lists the hardware mapping interfaces of the mapped controller and, if they have been mapped, the interfaces of instances whose facets are assigned to the project.

Optional Control Project Information

The following table describes the information that is added to each Control project sub-section when you select the corresponding option.

Optional selection	Description
Include Assignments	<p>Adds the <i>Assignments</i> table to each Control project sub-section.</p> <p>For each FBD project section, lists the instances and their facets that are assigned to it. Also shows the facet validity status and the assignment and generation statuses.</p>
Include Section Content and Controller Configuration	<p>Adds documentation that is generated by the Control Participant in a technical documentation folder.</p> <p>Documentation is generated only if a Control project contains at least one section and has been generated.</p> <ul style="list-style-type: none"> • Documentation file names: <Control project identifier (time stamp)>.pdf (for example, ControlProject_1 (20221025143510).pdf) • Technical documentation folder name: <System identifier (time stamp)>_ControlProject Reports. • Location of technical documentation folder: Same as the system documentation. <p>For details on the content of the Control Participant documentation, refer to <i>Topics in Documentation</i> in <i>EcoStruxure™ EcoStruxure Control Expert, Operating Modes</i>.</p> <p>NOTE: The information about the configuration does not correspond to the configuration (topology) of the controller that is mapped to executable of the Control project but to the default configuration that exists for each Control Participant project. To view information about the configuration part of the Control project, refer to the <i>Engine</i> column of the <i>Service Mappings</i> table of its executables and to the <i>Topology Domain</i> section of the system documentation.</p>

Default Supervision Project Information

The *Project Domain* section consists of a sub-section for each Supervision project, which contains the following:

- *Cluster* sub-section: A subsection exists for each cluster of a Supervision project. It contains the following information:
 - *Tag Container Properties* table: Lists the tag containers that exist.
 - *Server* tables: A table for each type of server that exists (such as alarm, report, or I/O).
 - *Boards* and *Ports* tables: A table for each, which lists the boards and ports that exist.
 - *IODevice*s table: Lists the I/O devices that exist.
- *Supervision Executable* sub-section: A subsection exists for each Supervision executable of a Supervision project. It contains the following information:
 - *Service Mapping* table: Lists the service mappings that exist.
 - *Communication Mapping* table: Lists the communication mappings that exist.

Optional Supervision Project Information

The following table describes the information that is added to each Supervision project sub-section when you select the corresponding option.

Optional selection	Description
Include Assignments	<p>Adds the <i>Assignments</i> table to each Supervision project sub-section.</p> <p>For each tag container, lists the instances and their facets that are assigned to it. Also, shows the facet validity status and the assignment and generation statuses.</p> <p>If the Supervision project contains more than one cluster, a table exists for each cluster.</p>
Include Genie Assignments	<p>Adds the <i>Genies</i> table.</p> <p>For each page, lists the instances and their facet that are assigned to it. Also, shows the facet validity status and the assignment and generation statuses.</p>
Include Pages	<p>Adds a graphical output of each page as a separate file in a technical documentation folder.</p> <ul style="list-style-type: none"> • Page file names: <Supervision project identifier>_<Page identifier>.pdf (for example, Supervision_1_Page_1.pdf) • Technical documentation folder name: <System identifier (time stamp)>_SupervisionProject Page Reports. • Location of technical documentation folder: Same as the system documentation. <p>NOTE: If a dialog box opens when you select the check box, Microsoft Print to PDF is not set as the default printer on the computer. In this case, you can generate the report but Supervision pages are not included. To include pages, close the dialog box, set the default printer, and select the check box.</p>
Include Tag Data	<p>Adds the content of the various DBF files of the Supervision Participant project as a separate files in CSV format in a technical documentation folder.</p> <ul style="list-style-type: none"> • File names: <Supervision project identifier>_<Table>.csv (for example, Supervision_1_variable.csv) • Technical documentation folder name: <System identifier (time stamp)>_SupervisionProject Tag Reports. • Location of technical documentation folder: Same as the system documentation.

Topology Domain Data

Object Selection

The **Topology** tab reproduces the hierarchy of the topology and the entities that exist. You can exclude objects from the documentation by clearing their checkbox.

NOTE: Clearing the checkbox of the root node excludes the entire domain from the documentation.

Default Information

The *Topology Domain* section contains a pair of tables for each type of topological entity that exists and has been configured (for example, controllers, workstations, I/O devices):

- *<Topological entity type>* table: For each type of topological entity, lists the entities that exist and the devices/services they consist of.
- *<Topological entity type> Properties* table: Lists some of the properties of each topological entity. For workstations, the *Workstation Services* table is shown instead.

Control Participant Documentation for Controllers, STB Islands, and PRM Devices

For each controller, STB island, and PRM device that is configured, additional documentation generated by the Control Participant (project documentation) is included in a technical documentation folder with a name that refers to the system (for example, *<System identifier (time stamp)>_Controller Reports*, *<System identifier (time stamp)>_Advantys Reports*).

Documentation is generated only if the topological entity is included in the system documentation.

- Documentation file name: *<Topological entity identifier (time stamp)>.pdf* (for example, *Controller_1 (20221025143510).pdf*)
- Location of technical documentation folders: Same as the system documentation.

For details on the content of the Control Participant documentation, refer to *Topics in Documentation* in *EcoStruxure™ EcoStruxure Control Expert, Operating Modes*.

Optional Information

The following table describes the information that is added to the *Topology Domain* section when you select the corresponding option.

When an entity that corresponds to an option (for example, a controller, a workstation) exists in the topology, the option is automatically selected. Conversely, clearing an option excludes entities of the corresponding type from the documentation.

Optional selection	Description
Include Controller Information	Adds the <i>Controllers</i> and <i>Controller Properties</i> tables as well the Control Participant documentation.
Include Workstation Information	Adds the <i>Workstation</i> and <i>Workstation Services</i> tables.
Include Ethernet Network Information	Adds the <i>Logical Networks</i> , <i>Logical Network Properties</i> , and <i>Physical Connections</i> tables.
Include STB Island Information	Adds the <i>STB Islands</i> and <i>STB Island Properties</i> tables as well as the Control Participant documentation.
Include Device Information	Adds the <i>DeviceIOs</i> and <i>DeviceIO Properties</i> tables.
Include PRM Profibus DP Information	Adds the <i>PRM Devices</i> and <i>PRM Device Properties</i> tables as well as the Control Participant documentation.

NOTE: When you select an option, a table is added only if a corresponding entity exists in the topology.

Global Templates Domain Data

Object Selection

The **Global Templates** tab reproduces the hierarchy of the entire Global Templates Library and the templates it contains. You can exclude templates from the documentation by clearing their checkbox.

NOTE: Clearing the checkbox of the root node excludes the entire domain from the documentation.

Default Information

The *Global Templates Domain* section contains the following:

- *Global Templates* table: Lists the templates of the Global Templates Library grouped by Library and location/type. It also indicates the template version, usability state, description, and whether it is used in the system.

Optional Information

The following table describes the information that is added to the *Global Templates Domain* section when you select the corresponding option.

Optional selection	Description
Include Change History	Adds the <i>Change History</i> table. For each template, lists the information that appears in the changes log. NOTE: The <i>Change History</i> table contains only templates that are used in the system if the Include Only Used Templates option is selected.
Include Only Used Templates	Limits the content of the <i>Global Templates</i> table to templates that are used in the system. Otherwise, includes the data of the templates that exist in the Global Templates Library. The option is selected by default. Clearing the option may generate a large documentation and take time.

Generating System Documentation

Procedure

To generate documentation containing information of a complete system or some of its domains, proceed as follows.

Step	Actions
1	In the Systems Explorer , select a system and verify that no other window is open in the engineering client.
2	Click Generate Report in the toolbar or context menu of the system. Result: The first Generate Report window opens.
3	In the first window, fill out the information as required, select a format, name, and location for the documentation, and click Next .
4	In the second window, select the scope of the documentation by selecting one or more domains and click Next . NOTE: At any time while the Generate Report window is open, you can go back to a previous window and modify your selection. Your selection in the current window or tab is retained.
5	In the third window, wait until the objects that exist in the various domains appear. Then, for each of the selected domains (tabs), refine the scope of the documentation and click Generate . Result: The Generate Report window closes and the documentation is generated. A dialog box informs you of the completion of the process and the report location that you have specified opens. NOTE: Do not interact with the Save Print Output As dialog box and Participant windows, which may open several times during the generation if the documentation contains Participant information. The dialog box and windows close automatically. NOTE: Once the Generate Report window closes, your selections are reset.

Managing the Consistency of the Topology

What's in This Chapter

Analyzing the Topology	1091
Analysis Rules	1094

Overview

This chapter describes how you can verify the consistency of the system topology and the rules that are used. It helps troubleshoot and fix configuration issues of communication attributes of entities.

Use the functionality during the initial engineering phase and after making changes.

Analyzing the Topology

Analyzing the Topology

Overview

You can perform an analysis of the topology by using the **Analyze** button in the toolbar of the **Topology Explorer** or analyze controllers by using the **Analyze** command in their context menu.

An analysis may be performed automatically when you use certain commands, for example, to configure the controller.

The analysis verifies the consistency of communication and/or hardware configuration data that is managed at the **Topology Explorer** level (such as IP addresses, ports, communication-related attributes, logical networks, and properties) against a set of predefined rules, page 1094. The scope of the analysis depends on the object that is analyzed.

The global result of the analysis is shown in the notification panel and if incorrect configurations are detected and/or advisories are generated, details are also available.

While an analysis is in progress, you and other users working on the same system can modify the attributes of entities or logical networks but changes may not be taken into account in this analysis. A new analysis must be started after changes are made.

NOTE: Depending on the number of entities, their type, and the number of logical networks that the topology contains, the analysis may take several minutes to complete.

Scope of the Analysis

The following table presents the scope of the analysis depending on the analyzed objects.

Analyzed object	Scope of the analysis
The topology.	The communication configuration of the entire topology and the entities it contains. Includes hardware configuration aspects.
One or more controllers.	Analyzes the configuration of the controller and its dependencies, such as scanned devices. Includes hardware and communication aspects.

NOTE: The analysis applies to data and settings configured in the **Topology Explorer**. Data and settings configured in the Control Participant are not analyzed.

Analysis Principle

Rules, page 1094 are grouped in categories and each rule is assigned a priority. The analysis is performed in an iterative way. In each iteration, the configuration of the selected objects is verified against one rule of each category.

The analysis ends when the objects have been verified against each rule of each category.




Rules that are not satisfied generate a notification in the notification pane. The ID of each rule is indicated.

NOTE: The setting of the **Under Construction** attribute of an entity can limit the extent of the analysis or exclude the entity completely.

Result of the Analysis

When an analysis is performed, it creates a main entry in the notification panel with an icon and *Analyze* in the **Action** column. Sub-entries give you a detailed message for each detection. An ID, page 1094 indicates the rule that is not satisfied.

Once the analysis is complete, either icon appears for this entry.

Icon	Description
	No incorrect configurations were detected and there are no advisories. No details are available.
	There is at least one advisory for communication and/or hardware configurations.
	At least one incorrect configuration was detected.

Because of the iterative analysis method that is used, you may see several messages for the same object.

The table describes the types of detections that the analysis can generate.

Detection (message in the notification panel)	Description
Incorrect configuration (Error)	<p>You must correct the configuration before you can perform actions on the entity.</p> <p>Example:</p> <p>The IP addresses of a controller module are not configured. This prevents you from configuring the controller.</p>
Advisory	<p>You can perform actions on the entity but the configuration may have an impact on the execution.</p> <p>Example:</p> <p>The port of a device is disabled but connected by a physical link. Communication with this port is not possible.</p>

Correcting Incorrect Configurations

Similarly to the way the analysis is performed, fix incorrect configurations in an iterative way.

Start by fixing incorrect configurations one by one for each logical network (if applicable). Fixing these may solve other less severe incorrect configurations for this logical network and/or incorrect configurations of entities.

Once you have fixed an incorrect configuration, perform a new analysis and repeat the process.

When no more incorrect configurations for logical networks are reported, start fixing those of entities. Perform an analysis each time you modify a configuration.

NOTE: Typically, when you use a command that triggers an analysis, the command will execute only once you have fixed incorrect configurations that were detected. Advisories do not need to be fixed but are shown in the notification panel.

Performing an Analysis

To perform an analysis of the topology or controllers, proceed as follows.

Step	Action
1	Open the Topology Explorer .
2	Optionally, you can lock the topology, page 525 so that no changes can be made while the analysis is in progress.
3	<p>Depending on the scope of the analysis, use either method:</p> <ul style="list-style-type: none">• The topology: Click the analysis button in the toolbar of the Topology Explorer.• One or more controllers: Select the Analyze command in the context menu of each controller. <p>Result: The analysis starts and an entry appears in the notification panel.</p>
4	<p>Once the analysis is completed, expand the entry and review the details.</p> <p>NOTE:</p>

Analysis Rules

Overview

This section describes the rules against which EcoStruxure Process Expert verifies the configuration of IP addresses, ports, communication-related attributes, and logical networks of the topology and its entities when you perform an analysis.

Analysis Rules - 1000 ID Series - Logical Networks

Rule Description

ID	Detection	Rule
1001	Incorrect configuration	The subnet address, page 576 must be valid.
1002		The gateway address must belong to the logical network and be different from the subnet address and the broadcast address. NOTE: The value 0.0.0.0 is accepted.
1003		The subnet address: <ul style="list-style-type: none">• Must be valid in relation to the subnet mask.• Must start at 1.0.0.0.• Cannot be a class D address.• Cannot start with 227 (loop back address) or 169.254. (APIPA). The subnet mask: <ul style="list-style-type: none">• Cannot be a class D subnet.• Must comply with the required format when addressing classful devices.
1004	Advisory	Each logical network must have unique attribute values, page 576.

Analysis Rules - 2000 ID Series - IP Addresses

Rule Description

The rules that appear in the following table apply to Internet Protocol version 4 (IPv4) addresses unless mentioned otherwise.

ID	Detection	Rule
2001	Incorrect configuration	An IP address must be configured except if the value of the Under Construction attribute is TRUE (check box selected). For BMENOC**** communication modules, IP address A is optional.
2002	Advisory	IP addresses must be associated to a valid logical network except if the value of the Under Construction attribute is TRUE (check box selected).
2003	Incorrect configuration	IP addresses must comply with the following rules to be valid: <ul style="list-style-type: none"> • Must start at 1.0.0.0. • Cannot be a class D address (>223.255.255.255). • Cannot be a loopback address (starting with 127). • Cannot start with 169.254, which is reserved for Automatic Private IP Addressing (APIPA) in a class B network. • Cannot be equal to the subnet address, page 576. • Cannot be equal to the broadcast address. NOTE: The value 0.0.0.0 is accepted for both SNMP managers , page 701. Refer also to rule 6028.
2004	Incorrect configuration (Advisory for BMENUA0100 and BMENUA100.2 modules)	An IP address must be valid for the configured logical network.
		An IP address cannot be equal to the subnet address, page 576 or broadcast address.
		If the NTP service of a module is configured as NTP Client , the primary and secondary NTP server IP addresses must comply with the following rules: <ul style="list-style-type: none"> • They must belong to the logical network of one of the enabled interfaces of the module (for example, the embedded or the backplane interface). • They cannot be equal to the network address of this logical network. • They cannot be equal to an address that ends with the following value regardless of the value of the subnet mask: <ul style="list-style-type: none"> ◦ 255.255.255 in a class A network. ◦ 255.255 in a class B network. ◦ 255 in a class C network. NOTE: The rules do not apply to BMENOC03●1 communication modules. NOTE: The value 0.0.0.0 is accepted.
		The IP address of either SNMP manager , page 701 is invalid if it ends with the following value regardless of the value of the subnet mask: <ul style="list-style-type: none"> • 255.255.255 in a class A network. • 255.255 in a class B network. • 255 in a class C network.
2005	Incorrect configuration	An IP address must be unique in the logical network.
2006	Advisory	The same IP address must not be used in different logical networks of the topology.
2007	Incorrect configuration	For the following modules, if IPv6 addresses are enabled and/or selected, the IPv6 addresses of the embedded interface, page 679 and of the primary and secondary NTP servers, page 698 cannot be null. <ul style="list-style-type: none"> • BMENUA0100 • BMENUA100.2 NOTE: The use of IPv6 addresses is not supported and you cannot save the configuration or synchronize changes if you enable it.

ID	Detection	Rule
2008	Advisory	<p>An IPv6 address must be unique within the topology.</p> <p>The rule applies to the following modules:</p> <ul style="list-style-type: none">• BMENUA0100• BMENUA0100.2 <p>NOTE: The use of IPv6 addresses is not supported and you cannot save the configuration or synchronize changes if you enable it.</p>
2009	Incorrect configuration	<p>The IP address of each BMECRA**** adapter module in an M580 controller must be unique .</p>

Analysis Rules - 3000 ID Series - Logical Network Configuration

Rule Description

ID	Detection	Rule
3000	Incorrect configuration	IP addresses MainIP and IPA of a controller module must belong to the same logical network.
3001		In a redundant controller, IP addresses MainIP , MainIP + 1 , IPA , and IPB of the controller modules must be unique and belong to the same logical network. The rule applies also to IP addresses IPA and IPB of communication modules of the controller.
3002		If the following conditions are fulfilled for the backplane port of a communication or gateway module that is located in the local rack or drop of a controller: <ul style="list-style-type: none"> It is enabled. It is associated to the embedded interface, page 679. If the module has routing capability, its Interface attribute is disabled (no IP forwarding). Then, the IP addresses of the embedded interface of the module and the controller module must belong to the same logical network. NOTE: The rule does not apply to BMENOS0300 modules.
3003		If the Interface attribute of the backplane port of a communication module is enabled, the IP address of the backplane port (BKP_Interface) must belong to the same logical network as the IP addresses of the embedded interface of the controller module. NOTE: The rule does not apply to BMENUA0100 modules.
3004	Advisory or incorrect configuration	Typically, the IP addresses of two ports that are connected by a physical link , page 608 must belong to the same logical network. The type of detection depends on the type of objects that are connected: <ul style="list-style-type: none"> A managed switch and an entity or module: Advisory. The link may be valid depending on the configuration of the switch. Two entities or modules: Incorrect configuration. The link is not valid. Modify the configuration of either entity/module or delete the link. NOTE: If a Modbus TCP or EtherNet/IP device is scanned by a BMENOC0301.4 or BMENOC0311.4 communication module, an advisory is indicated instead of an incorrect configuration if the Scanning Device Outside Logical Network attribute of the module is set to TRUE (check box is selected). The detection is performed only for direct connections between two ports. If an entity that does not manage an IP address (for example, an unmanaged switch) is connected in between, the logical network discrepancy is not detected.
3005		The IP addresses of the embedded interfaces of the BMECRA**** adapter module and the controller module of a same controller must belong to the same logical network.
3006	Incorrect configuration	In a redundant controller, the IPv4 addresses A and B, page 679 of the control port of a BMENUA0100 module must belong to the same logical network.
3007		The interface of a scanned device must belong to the same logical network as the interface of the scanner. An exception exists for BMENOC0301.4 and BMENOC0311.4 modules. They can scan devices that communicate by using the Modbus TCP or EtherNet/IP protocol although their interface belongs to a different logical network.
3008	Advisory	ID:3008 is an information only and is displayed depending on the configuration of the Scanning Device Outside Logical Network attribute. Verify and, if necessary, correct the configuration in the Communication Mapping View , page 544 and/or LOGICAL NETWORK VIEW .
3009	Incorrect configuration	The IP addresses of the Embedded Interface of a controller module must be different.
3010	Incorrect configuration	If the control port of a BMENUA0100 module is enabled, either IPv4 or IPv6, page 679 must be enabled. NOTE: The use of IPv6 addresses is not supported and you cannot save the configuration or synchronize changes if you enable it.
3011		If the primary or secondary NTP server IP address format, page 698 of a BMENUA0100 module is set to IPv6, the control port, page 682 and the IPv6 embedded interface of the module must be enabled. NOTE: The use of IPv6 addresses is not supported and you cannot save the configuration or synchronize changes if you enable it.

ID	Detection	Rule
3012	Incorrect configuration	If the ETH1 port, page 682 of a BMENOP0300 module is set to Extended Network , its BKP port must be disabled.
3013	Advisory	Authorized addresses in the ACCESS CONTROL VIEW cannot be identical to the following IP addresses of M580 controllers and BMENOC03•1 communication modules: <ul style="list-style-type: none">• MainIP• IPA• IPB

Analysis Rules - 4000 ID Series - Routing

Rule Description

ID	Detection	Rule
4000	Advisory	<p>If dynamic routing is used and the Interface attribute of a port is enabled, the IP address of the port (attributes of Interface) must be the gateway address of the logical network.</p> <p>(The rule does not apply to BMENOC0321 communication modules in redundant controllers.)</p>
4001	Incorrect configuration	<p>For an entity or module, the Logical Network attribute of each interface, page 679 that is associated to a port (for example, ETH1_Interface) must be unique and/or different from attribute of the Embedded Interface.</p> <p>Typically, this applies when interfaces are created by enabling the Interface attribute of a port (for example, BKP). However, other settings can also create port-specific interfaces.</p>
4002		The host address of a router must not belong to a logical network that any other of its ports belong to (except for BMENOC0321 communication modules).
4006		If several BMENOC0321 communication modules are installed on the rack of a standalone controller, only one of them can have the Interface attribute of its backplane port enabled.
4007		In a redundant controller, a BMENOC03•1 communication module in local rack A must not have either of its dual ports linked to the BMENOC03•1 communication module in local rack B if their backplane port is enabled.
4009	Advisory	For a given logical network, only one device can have a port configured with the gateway address of this network (IP address of the Interface attribute of the port).
4010	Incorrect configuration	For a workstation, the Port value that is configured for each Control Service must be unique.
4011	Incorrect configuration	For BMENOC03•1 communication modules, the backplane port cannot be enabled if the IPsec service is enabled.

Analysis Rules - 5000 ID Series - Physical Links

Rule Description

ID	Detection	Rule
5000	Advisory	A port that is disabled should not be linked.
5003		If a device has redundant ports for which RSTP is enabled, a link needs to exist on both ports.
5004	Incorrect configuration	A physical link connects two port types, page 682 that support different networks (for example, RIO and DIO). If one of the ports is disabled, the message is not shown.
5007	Advisory	A physical link connects two ports that have different RSTP settings. If one of the ports is disabled, the message is not shown.
5008		A physical link connects two ports that have different connector types, page 608. If one of the ports is disabled, the message is not shown.
5009	Incorrect configuration	A physical link connects two ports that belong to different redundant networks, page 539. If one of the ports is disabled, the message is not shown.

Analysis Rules - 6000 ID Series - Services

Rule Description

ID	Detection	Rule
6001	Advisory	If a local slave, page 709 is configured for the controller or BMENOC03•1 module of an M580 controller, the EIP service of this module must be enabled in the Services tab of the PROPERTIES pane.
6002	Incorrect configuration	For the following modules, if the NTP client service, page 698 is enabled, the NTP polling value must be configured in the range from 1 to 120: <ul style="list-style-type: none"> • BMENOC0301 • BMENOC0311 • BMENOC0321 • Modicon M580 controller module
6003		Using a BMECRA**** adapter module requires that the following services, page 694 of the associated controller module be enabled: <ul style="list-style-type: none"> • TFTP • EIP • DHCP
6004	Advisory	If the DHCP server is disabled, client devices that subscribe to it in the DHCP VIEW do not receive IP addresses and network parameters. Enable the service in the Services tab, page 694 of the PROPERTIES pane.
6005	Advisory	If the DHCP server does not support the protocol, page 694 that is used by a client entity, it cannot provide IP addresses and network parameters to the entity. Assign the entity to a server that supports the protocol in the DHCP VIEW or change its protocol in the Services tab of the PROPERTIES pane, if possible.
6006	Incorrect configuration	If the DHCP client, page 694 service is enabled and a DHCP server is configured for an entity, Identifier cannot be empty. For some devices, the DHCP identifier must comply with device-specific format requirements. Default identifiers may not fulfill these requirements; for example, for generic devices.
6007	Advisory	If the DHCP client, page 694 service is enabled for an entity that is scanned, page 544 by a scanner, the DHCP server that is configured for this entity must be the scanner.
6008		If the control port, page 682 of a BMENUA0100 module is disabled, the primary and secondary NTP server IP addresses, page 698 in IPv4 format must belong to the same logical network as the BKP interface of the module. The same applies if the control port is enabled but the IPv4 interface of the module is disabled. In a redundant controller, if the control port and the IPv4 interface of the module are enabled, the NTP server IP addresses in IPv4 format of modules in local racks A and B must belong to the same logical network. NOTE: The incorrect configuration is not detected if the IP addresses are set to 0.0.0.0. An advisory may be indicated (see rule 6018).
6009		Using a BMENUA0100 module requires that the following services, page 694 of the associated controller module be enabled: <ul style="list-style-type: none"> • TFTP • DHCP
6010		Using a BMENUA0100 module requires that Data Dictionary be enabled in the project settings of the associated control project
6011		One or more of the following settings is disabled: <ul style="list-style-type: none"> • TFTP • DHCP Ensure that the configuration of security services, page 690 is appropriate for your application.

ID	Detection	Rule
6012	Incorrect configuration	If the Update CPU Time functionality is used, enable it only for one communication module of the controller and verify that the controller module is not configured as NPT client.
6013		If the IPsec services , page 690 is enabled, Secure Protocol must be set to IPsec and the Pre-Shared Key attribute cannot be left blank.
6014		If QoS is selected for the L2 Service of a physical port, page 682, the QoS service must be enabled for the module.
6015		If the DHCP client, page 694 service is enabled for an entity, the DHCP server name must be configured.
6016		For TeSys devices, the DHCP identifier, page 694 cannot exceed 16 characters.
6017		If the DHCP client, page 694 service is enabled and configured to use the MAC address to identify the entity, this address must be properly configured.
6018	Advisory	To use the NTP service, page 698 with SNTP, at least the IP address of the primary NTP server must be configured.
	Incorrect configuration	To use the NTP service, page 698 with NTP version 4, at least one NTP server IP address must be configured.
6019	Incorrect configuration	If the DHCP client, page 694 service is enabled for an entity, the DHCP service of its DHCP server must be enabled.
6021		Verify that for each DHCP client that subscribes to the same DHCP server, the DHCP identifier, page 694 is unique.
6022		For a BMENUA0100 module, if the control port is enabled, the IP address that is configured for the primary or secondary NTP server must be different from the static IP address of the control port (Embedded Interface). This applies to IPv4 and IPv6 addresses. For a redundant controller, it applies to any combination of roles A and B.
6024		If SNMPv3 is enabled, the User Name attribute, page 701 cannot be empty.
6025		If the Syslog server is enabled, page 700, NTP State must be set to NTP Client .
6026		If the Syslog server is enabled, page 700, the following IP addresses of the M580 controller module must belong to the same logical network: <ul style="list-style-type: none"> • IP Address of the Syslog server • MainIP • Primary NTP Server • Secondary NTP Server
6027	Advisory	The values configured for the Syslog Server category, page 700 of a controller are those that are used for communication with the Syslog server.
6028	Incorrect configuration	The IP addresses of the first and second SNMP managers, page 701 must be different. Applies to the following modules: <ul style="list-style-type: none"> • M580 standalone controller modules • BMECRA31210 • BMENOC0301, BMENOC0310, BMENOC0321 • BMENUA0100 • BMXCRA31200, BMXCRA31210
6029		To use the NTP service, page 698 with NTP version 4, the IP address of the preferred NTP server must be configured.
6045		If Access Control is enabled, page 691 for a module and its Main IP interface is configured, the logical network that this IP address belongs to must be added to the authorized addresses of the module in the ACCESS CONTROL VIEW .
6046		If Access Control is enabled, page 691 for a module, the following cannot be added to the authorized addresses of the module: <ul style="list-style-type: none"> • A logical network, if it is a subnet of an existing authorized logical network of this module in the ACCESS CONTROL VIEW. • An IP address, if it belongs to an existing authorized logical network of this module in the ACCESS CONTROL VIEW.

Analysis Rules - 7000 ID Series - M580 Controllers

Rule Description

ID	Detection	Rule
7001	Incorrect configuration	<p>For a given scanner, the following rules must be satisfied:</p> <ul style="list-style-type: none"> The total number of requests (Modbus TCP) and connections (EtherNet/IP) of the active I/O profiles, page 709 that appear in the Communication Mapping Views, page 544 does not exceed the capacity of the scanner. The total number of devices (Modbus TCP and EtherNet/IP) that are scanned does not exceed the capacity of the scanner. <p>For details on the capacity, refer to documentation of the module that is used for scanning.</p>
7002		If the DHCP client of a BMECRA**** adapter module is enabled, the Identifier attribute, page 694 must be correctly configured.
7003		For a given scanner, each scanned device must have a unique device number, page 709 independently of the communication protocol that is used by the device.
7007		For a given module, local slave connections, page 709 of a same type (for example, Exclusive Owner) that appear in the Local Slaves subcategory must have a unique number.
7008		The Default Gateway/Bridge Device attribute can only be used if the scanner is a BMENOC0301/11.4 communication module.
7009		For a given module, if two local slave connections, page 709 have the same number (for example, Local Slave 1) but a different connection type, the values of their Default RPI T->O , Input Size , and Default RPI O->T attributes must be identical.
7010	Advisory	For BMENOP0300 modules, the values that you enter for interfaces and ports, page 676 in the Topology Explorer must also be manually entered in the IEC 61850 Configuration Tool. You can open the tool from the Control Participant if it is installed.
7011		<p>For BMENOS0300 modules, the configuration of the Ethernet ports, page 682 needs to be performed by using the rotary switches located on the module. Configuring Service L2 in the PROPERTIES pane has no impact.</p> <p>For details, refer to the module installation and configuration guide.</p>
7012	Incorrect configuration	<p>If the topology contains an M580 controller as of version 4.00, you must change, page 588 its default Web Diagnostic/Data Storage password to configure it.</p> <p>NOTE: Other configurations may be required.</p>

Analysis Rules - 9000 ID Series - I/O Profiles

Rule Description

ID	Detection	Rule
9001	Incorrect configuration	For a given register, page 709, Read Length and Write Length cannot be null at the same time.
9002		<p>For a given communication profile, page 709:</p> <ul style="list-style-type: none">• The Read Address values configured for several registers cannot overlap.• The Write Address values configured for several registers cannot overlap. <p>A read or write address is verified only if a non-null value is entered in the corresponding Length attribute.</p>

Analysis Rules - 10000 ID Series - Hardware

Rule Description

ID	Detection	Rule
10000	Incorrect configuration	The maximum number of BME-series Ethernet communication modules that you can install in a rack with a BMEP582040 controller module is 2.

System Monitoring

What's in This Chapter

Displaying the Communication Status of Entities in Physical Views	1107
Displaying the Identification Mismatch Icon	1111
Displaying Operating States of M580 Controllers	1112
Displaying Operating States of Altivar Process Devices.....	1114
Displaying Operating States of TeSys Devices.....	1115
Displaying Operating States of Generic Automation Devices	1116
Displaying Operating States of Hirschmann Switches	1117

Displaying the Communication Status of Entities in Physical Views

Overview

System monitoring lets you know whether entities of the topology that appear in a physical view are active by using a diagnostic engine. The engine sends requests to the IP address of each entity and monitors responses.

The communication status of an entity (controllers, modules, devices, or workstations) is indicated by a diagnostic icon in the physical view. You can add, modify, and remove entities while system monitoring is active. The status is indicated or refreshed automatically.

Status Detection Method

Unless mentioned otherwise, the diagnostic engine that runs on the system server computer sends ping requests to the IP address that is configured for the **Embedded Interface**, page 679 of the entity and monitors responses.

For modules of M580 controllers, the following IP addresses are used:

- Standalone controllers: **MainIP**
- Redundant controllers: **IPA** and **IPB**

NOTE: While system monitoring is active, entities that do not support the functionality do not show diagnostic information (for example,).

NOTE: System monitoring is not supported for:

- Devices of the **EPE Managed Devices** product range.
- Entities for which an IP address cannot be configured in the **Topology Explorer**, such as Quantum controllers.
- Entities that do not manage an IP address.

Prerequisites

The following are prerequisites for using system monitoring:

- The entities are added to a physical view.
- In a distributed EcoStruxure Process Expert architecture, ensure that the engineering client can communicate with the physical devices modeled by the entities.
- For devices of the **Generic Devices** product range, diagnostics are enabled, page 717.
- The SODB service, page 1110 is running.
- The firewall is configured to allow communication between the engineering client computer and entities by using port 502.

Starting System Monitoring

To use system monitoring, proceed as follows.

Step	Action
1	Open the Topology Explorer .
2	Click the Activate System Monitoring button in the toolbar of the Explorer. Result: The button appears selected, Diagnostic engine ready is displayed in the notification panel, and diagnosis starts. The communication status is indicated on entities in physical views.

Stopping System Monitoring


When you close the **Topology Explorer**, the engineering client, or if the connection to the system server is interrupted, the diagnostic engine is stopped.

To stop using system monitoring, click the **Activate System Monitoring** button in the toolbar.

Result: The button does not appear selected anymore, diagnosis stops, and the communication status is not indicated on entities anymore.



Impact of Entity Configuration on System Monitoring

The table indicates the impact of the configuration of entities on the diagnostic information.

Configuration	Result
No IP address for the embedded interface is configured (independently of the configuration of the Logical Network attribute).	Only the <i>service not available</i> icon is shown.  (Gray dot)
An IP address for the embedded interface is configured but the logical network attribute is set to NOCONF .	Diagnostic information is available.
The Under Construction attribute, page 676 is set to TRUE (selected).	The entity does not show diagnostic information. NOTE: This also applies to modules that inherit their under construction status from the parent entity.

Statuses Indicated by Diagnostic Icons

The table describes the meaning of the icons that appear on entities displayed in a physical view. The icon description is shown in a tooltip.

Icon	Description	Device status
	Service being started or not available (gray dot).	Diagnostic information is not available. This can be caused by a non-operational diagnostic engine, page 1110. Default icon for entities and modules whose embedded interface IP address is not configured.
	Not reachable (red dot with black cross).	The entity is not reachable or did not respond to the request.
No icon appears.		The entity responded successfully to the request. If several requests were sent by using different protocols (for example, ping and SNMP), the entity responded to, at least, one of them.

NOTE: Other diagnostic icons, page 1112 may appear on certain entities and modules.

Representation of the Communication Status

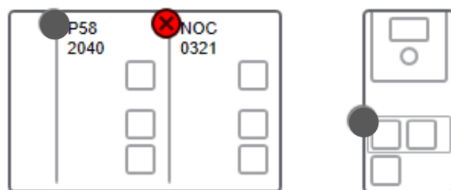
The following figure shows an example of a workstation and an M580 controller displaying diagnostic icons.



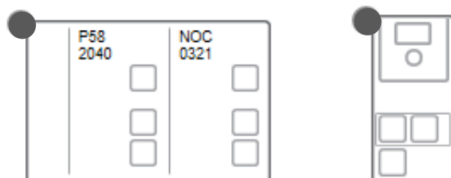
Representation of the Communication Status for Entities Containing Modules

Depending on the zoom level that is applied in a physical view, the diagnostic icons for entities that contain modules are shown as follows.



100% and higher: The icon is shown on each module.



Less than 100%: A global diagnostic icon is shown at the local rack or entity level.



The table describes the resulting global diagnostic icon based on the icon of each module.

Icon of individual modules	Resulting global icon for rack or entity
No icon for the modules of the entity.	None
<ul style="list-style-type: none">At least one module has the gray dot icon.The icon of other modules of the entity is not relevant.	 Gray dot. Service being started or not available
<ul style="list-style-type: none">At least one module has the red dot with black cross icon.No module has the gray dot icon.	 Red dot with black cross. Not reachable

NOTE: Other diagnostic icons, page 1112 may appear on certain entities and modules.

Troubleshooting

If you cannot obtain diagnostic information, ensure that the SE SODB ControlExpert Host X service (where X represents the version) of the Service Oriented Device Bus (SODB) diagnostic engine is running on the Control Expert server computer (**Services** window of the Windows operating system). If not, start the service and enable system monitoring again.

Displaying the Identification Mismatch Icon

Overview

When system monitoring, page 1107 is active, you can see whether a device or module of the topology is incompatible with or different from the physical device or module that is configured with the same IP address.

An incompatibility or identification mismatch is indicated by an icon on the device or module in the physical view and you cannot deploy to the device or module nor load data from it.

For more information on M580 controller compatibility, refer to the topics describing controller to PC transfer (see *EcoStruxure™ Control Expert, Application Versions and Controller Firmware, Compatibility Rules, User Manual*) and controller application compatibility (see *Modicon M580, Hardware, Reference Manual*).

NOTE: The incompatibility or identification mismatch is also detected when system monitoring is inactive but no icon is shown on devices or modules to indicate it.

Identification Mismatch Icon

The icon that indicates an incompatibility or identification mismatch is the same as the yellow triangle icon indicating a not operational device.



The tooltip indicates **Device different**.

Detection Method and Supported Devices

The diagnostic engine, page 1107 that runs on the system server computer sends a Modbus request with function code 43, subcode 14 to objects by using the IP address of their embedded interface.

The compatibility or identification mismatch detection functionality is supported only for modules and entities that respond with a *ProductCode* (product code) value that is identical to their **Part Number** as it appears in the Schneider Electric catalog.

For example, the functionality is supported for an M580 controller because the returned *ProductCode* value and its **Part Number** are both equal to, for example, BMEP582040.

Displaying Operating States of M580 Controllers

Overview

System monitoring lets you obtain operating state information for Modicon M580 controller modules by using the diagnostic engine, page 1107.

The engine sends requests to their IP address and monitors responses.

The state of the module is indicated by a diagnostic icon in the physical view in which the module appears.

NOTE: Use the indication of the operating state for commissioning only as it does not replace the use of proper system monitoring.




NOTE: For BMENOC03•1 communication modules, the indication of the operating state in a physical view is not available. To view their operating state, configure the controller, open the module, and connect to the controller. At the top of the screen, the state LEDs of the module are shown. Alternatively, you can use the rack viewer, page 559 in **ONLINE** view mode.

Detection Method

The diagnostic engine that runs on the system server computer sends Modbus TCP requests to the IP address of the controller module, page 1107 to obtain information about its state.

Operating State Representation

The table describes which operating states can be indicated on M580 controllers and the icons that are used. The icon description is shown in a tooltip. The indication of an operating state implies that the device is *reachable* and *responded*, page 1109.

Icon	Description	Controller state
	RUN. Green arrow.	Refers to the state of the RUN LED of the controller module (see <i>Modicon M580, Hardware, Reference Manual</i> and <i>Modicon M580, Safety Manual</i>). It indicates that the LED is either on or flashing. If another diagnostic icon is already shown, the run icon appears below.
	Not fully operational. Blue diamond.	Non-blocking condition linked to an I/O detected error and the controller is not in HALT state. NOTE: When this icon is displayed while the <i>RUN</i> icon is present, it indicates that at least one task is not in RUN state. To know the operating mode of a task, edit the control project and open the Task tab of the PLC Screen .
	Out of Order/Not operational. Yellow triangle. NOTE: It can also indicate an incompatibility or mismatch, page 1111 between the configured and the physical object.	Non-blocking condition linked to an I/O detected error and the controller is in HALT state. Blocking condition that led the controller to go in HALT state. The controller may be able to communicate.











NOTE: For more information on states of M580 controllers and blocking/non-blocking conditions, refer to the topic describing M580 diagnostics (see *Modicon M580, Hardware, Reference Manual*).

Representation of the Operating State for Controllers

Depending on the zoom level that is applied in the physical view, the diagnostic icons for controllers are shown as follows:

- 100% and higher: Icons are shown on each module of the local rack.
- Less than 100%: One or more global diagnostic icons are shown at the local rack level.

The table describes the resulting global diagnostic icon based on the icon of modules on the local rack.

Controller icons	Icon of other modules on the local rack	Resulting global icons on local rack
 or  and/or 	No module has the <i>not reachable</i> icon, page 1109  .	Same as on the controller module
 or  and 	At least one module has the <i>not reachable</i> icon  .	 (not reachable) and 

Displaying Operating States of Altivar Process Devices

Overview

System monitoring lets you obtain operating state information for devices of the **Altivar Variable Speed Drives** product range by using the **diagnostic engine**, page 1107. The engine sends requests to their IP address and monitors responses.

The state of the device is indicated by a diagnostic icon in the physical view in which the device appears.

NOTE: Use the indication of the operating state for commissioning only as it does not replace the use of proper system monitoring.

Detection Method

The diagnostic engine that runs on the system server computer sends Modbus TCP requests by using the following IP address to obtain information about the state of the device:



- For **ATV6xx** devices, the IP address of the embedded interface of the device or of the communication card.

NOTE: If both IP addresses are configured, only the communication card IP address is used.

- For **ATV9xx** devices, the IP address of the embedded interface of the device.

States Indicated by Diagnostic Icons

The table describes which operating states can be indicated on devices of the **Altivar Variable Speed Drives** product range and the icons that are used. The icon description is shown in a tooltip. The indication of an operating state implies that the device is **reachable and responded**, page 1109.

Icon	Description	Device state description
	Not fully operational. Blue diamond.	Bit 7 of the [CIA402 State Reg] EtA status word = 1.
	Out of Order/Not operational. Yellow triangle. NOTE: It can also indicate an incompatibility or mismatch, page 1111 between the configured and the physical object.	Bit 3 of the [CIA402 State Reg] EtA status word = 1. The operating state of the device is <i>Fault</i> .

NOTE: The representation of the diagnostic icon varies depending on the zoom level, page 1109.

NOTE: For more information on the status word bits, refer to the *Programming Manual* of the corresponding Altivar Process variable speed drive.

Displaying Operating States of TeSys Devices

Overview

System monitoring lets you obtain operating state information for devices of the **TeSys Motor Controllers** product range by using the [diagnostic engine](#), page 1107. The engine sends requests to their IP address and monitors responses.

The state of the device is indicated by a diagnostic icon in the physical view in which the device appears.



NOTE: Use the indication of the operating state for commissioning only as it does not replace the use of proper system monitoring.

Detection Method

The diagnostic engine that runs on the system server computer sends Modbus TCP requests to devices by using the IP address of their embedded interface to obtain information about their state.


States Indicated by Diagnostic Icons

The table describes which operating states can be indicated on devices of the **TeSys Motor Controllers** product range and the icons that are used. The icon description is shown in a tooltip. The indication of an operating state implies that the device is [reachable](#) and [responded](#), page 1109.

Icon	Description	Device state description
	Not fully operational. Blue diamond.	Bit 3 of the register 455 status word = 1 (system status register 1).
	Out of Order/Not operational. Yellow triangle. NOTE: It can also indicate an incompatibility or mismatch, page 1111 between the configured and the physical object.	Bit 2 of the register 455 status word = 1 (system status register 1). The operating state of the device is <i>Fault</i> .

NOTE: For more information on the status word bits, refer to the *User Manual* of the corresponding TeSys motor controller.

The table describes which operating state can be indicated on TeSys islands of the **TeSys Motor Controllers** product range and the icon that is used. The icon description is shown in a tooltip. The indication of an operating state implies that the device is [reachable](#) and [responded](#), page 1109.

Icon	Description	Device state description
	Not fully operational. Blue diamond.	The device is in a pre-operational state or encountered a minor event. Bit 4 of the register 3201 = 1 (pre-operational). Bit 3 of the register 3201 = 1 (minor event).

NOTE: For more information on the *SystemAvatar* function block, refer to the *Third Party Function Block Guide* of the TeSys island.

Displaying Operating States of Generic Automation Devices

Overview

System monitoring lets you obtain operating state information for devices of the **Generic Devices** product range by using the [diagnostic engine](#), page 1107. The engine sends requests to their IP address and monitors responses. The functionality is available for devices that support it.

The state of the device is indicated by a diagnostic icon in the physical view in which the device appears.

To use this functionality, [configure the diagnostic attributes](#), page 717 of the device first.




NOTE: Use the indication of the operating state for commissioning only as it does not replace the use of proper system monitoring.

Detection Method

The diagnostic engine that runs on the system server computer sends requests to the embedded interface IP address of the device to obtain information about the state of the device based on the configuration of its diagnostic attributes.

States Indicated by Diagnostic Icons

The table describes which operating states can be indicated on devices of the **Generic Devices** product range and the icons that are used. The indication of an operating state implies that the device is [reachable](#) and [responded](#), page 1109.

Operating state	Icon
Not Fully Operational	 (blue diamond)
Out Of Order	 (yellow triangle)
Run/Stop Status	 (Run state. Green arrow.) No icon is shown for the stop state.

Displaying Operating States of Hirschmann Switches

Overview

System monitoring lets you obtain operating state information for devices of the **Network Infrastructure Device** product type, **Hirschmann Switches & Hubs** product range by using the **diagnostic engine**, page 1107. The engine sends requests to their IP address and monitors responses.

The state of the device is indicated by a diagnostic icon in the physical view in which the device appears.

NOTE: Use the indication of the operating state for commissioning only as it does not replace the use of proper system monitoring.


Detection Method

The diagnostic engine that runs on the system server computer sends Modbus TCP requests to devices by using the IP address of their embedded interface to obtain information about their state.

NOTE: The detection of **device incompatibility or mismatch**, page 1111 is not supported for devices of the **Hirschmann switches & Hubs** product range.

States Indicated by Diagnostic Icons

The table describes which operating state can be indicated on devices of the **Hirschmann switches & Hubs** product range and the icon that is used. The icon description is shown in a tooltip. The indication of an operating state implies that the device is **reachable and responded**, page 1109.

Icon	Description	Device state description
	Not fully operational. Blue diamond.	The value of at least one of the <i>Port [1..X] Link Status</i> registers = 0 (where X is the number of ports that the switch is equipped with).

NOTE: For more information on operating states, refer to the documentation of the device.

Basic Diagnostics

What's in This Chapter

Basic Diagnostics for Altivar Process Devices1118

Basic Diagnostics for Altivar Process Devices

Overview

You can view various diagnostic codes for devices of the **Altivar Variable Speed Drives** product range in the **basic diagnostic window**, page 558. The diagnostic engine, page 1107 sends requests to the IP address that is configured for the device and monitors responses.

NOTE: Devices that are configured by using a DTM, may also feature the **Diagnosis** context menu command, which lets you use the diagnostic functionality of their DTM.

Detection Method

The detection method, page 1114 that is used is the same as that used to detect the operating state of Altivar Process devices.

Available Diagnostic Data

The table describes the diagnostic codes that are available for each basic diagnostic category. For a description of the codes and the corresponding register, refer to the *Programming Manual* of the corresponding Altivar Process variable speed drive.

Status [Drive State]	Last Detected Error [Last Error]	Warning/Alarms [Last Warning]
<i>HMIS</i> register number 3240	<i>LFT</i> register number 7121	<i>LALR</i> register number 12890
<i>RDY</i>	<i>NOF</i>	<i>NOA</i>
<i>NST</i>	<i>CFF</i>	<i>FRF</i>
<i>RUN</i>	<i>CFI</i>	<i>RLS</i>
<i>ACC</i>	<i>CNF</i>	<i>STT</i>
<i>DEC</i>	<i>EPF1</i>	<i>SRA</i>
<i>FLT</i>	<i>OPF1</i>	
<i>NLP</i>	<i>PHF</i>	
	<i>EPF2</i>	
	<i>ETHF</i>	
	<i>FDR1</i>	
	<i>FDR2</i>	

Content Repository

What's in This Chapter

Content Repository Explorer	1120
Temporary Contents Repository Files	1124

Content Repository Explorer

Overview

The content repository is the place where EcoStruxure Process Expert stores:

- Global files used by Global Templates and that are required by the platform to provide various services (for example, global constituents referenced by templates). These files are associated to the Library that is installed during the installation of EcoStruxure Process Expert and to optional application Libraries.
- System-related files (for example, logical Participant projects or user files attached to a Supervision project or to instances). These files are created during the different stages of the system engineering life cycle.

You cannot add to or remove files directly from the content repository. The system-related files, such as Participant project files are removed from the content repository when they are not required anymore (for example, when you delete a Participant project). An exception exists for documents that you can manage yourself (see *EcoStruxure Process Expert, Runtime Navigation Services, User Guide*).

Each system server possesses one content repository in which the global and system-related files are stored.

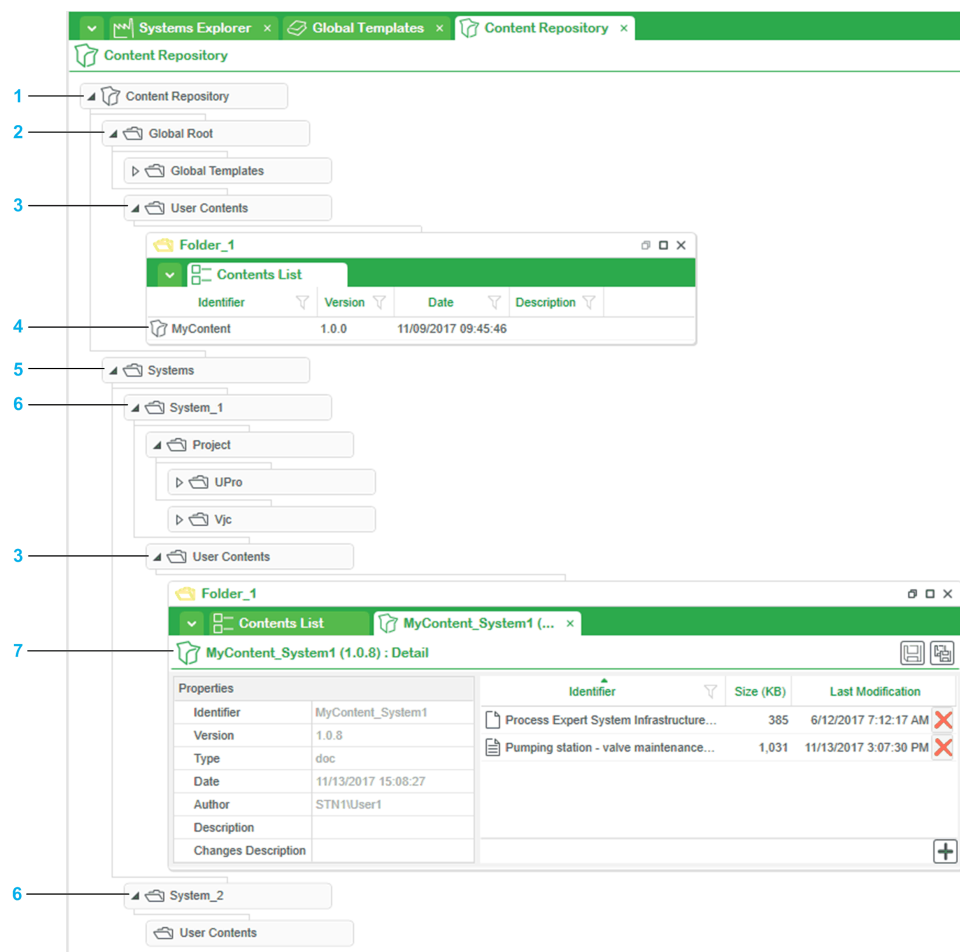
The **Content Repository** explorer is the tool, which lets you view contents in the form of a folder structure and work with the files that are stored in the content repository.

Opening the Content Repository Explorer

To open the **Content Repository** explorer click the **Content Repository** icon in the **toolbar**, page 104.

Content Repository Explorer Window

The following figure shows an example of the **Content Repository** explorer window.



Item	Description
1	Content repository root folder.
2	Root folder for global content repository items that are available at the platform level (system independent).
3	User Contents folders that are the parent folder of the folder structure and content containers that you can add depending on your role.
4	User-created content container.
5	Root folder for content repository items that are available to systems.
6	Root folders for content repository items that are available only to one given system.
7	User-created content container open in the content editor.

Description of the Contents of the Repository

The following data is stored in the content repository under the **Global Root** \Global Templates parent folder.

Node	Contains
Global Constituents	Global Control and Supervision Participant constituents (such as variables, DFBs, included projects).
Local Constituents	Control facet templates of the Global Templates Library and for each facet, the referenced local constituent sources and the referenced global constituents
User Contents	<p>Documents (files) that can be managed by users having the appropriate role (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>). They are delivered with EcoStruxure Process Expert and organized in a predefined folder structure and content containers (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>).</p> <p>These documents can be opened when viewing instance information by using runtime navigation services.</p> <p>They can be associated either to templates or instances.</p>

The following data is stored in the content repository under the **Systems** parent folder.

Node	Contains
<system identifier>	<p>For each system that exists in the database of the system server, the complete set of files defining the various existing Participant projects, their components, and topological entities.</p> <p>Data is grouped by Participant.</p>
Project	<p>The logical Participant projects under the folder with the name of the project. If you update the logical Participant project by using the Update Project command, the project file is updated, reflecting the changes.</p> <p>For each Participant project, data of the various components of the logical Participant project such as:</p> <ul style="list-style-type: none"> • Section export files under the folder with the name of the section. • The built Participant projects under the folder with the name of the executable.
Topology	<p>Configuration files and profiles of the entities that exist in the topology of the system (under the folder with the name of the entity).</p> <p>For entities that you configure by using the Control Participant, the corresponding project files.</p>
Upro	For each entity of the system, the deployed Participant project. If you do online refinement, the file is updated, reflecting the changes.
VJC	For each Supervision Participant project of the system, the various project files such as database and configuration files.
User Contents	<p>User documents (files) that can be added and managed by users having the appropriate role (see <i>EcoStruxure Process Expert, Installation and Configuration Guide</i>). They are organized in a user-created folder structure and content containers (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>) at the system level.</p> <p>These documents can be opened when viewing instance information by using runtime navigation services.</p> <p>They can be associated either to templates or instances.</p>

Content Repository Root Folder Actions

Right-click the **Content Repository** root folder to open a context menu with the following commands.

Command	Description
Export User Contents	Opens the Export User Contents window, which lets you export content containers (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>) with the documents they contain as a single file (.cbk).
Import User Contents	Opens the import window, which lets you select an export file (.cbk) to import (see <i>EcoStruxure Process Expert, Runtime Navigation Services, User Guide</i>) the folders and content containers it contains.

Content Repository File Actions

Starting out from one of the previously mentioned parent folders:

1. Drill down to the lowest level folder of a node until you reach the **Contents List** tab (this requires opening the folder).
2. Right-click a content and select **View Content Detail**.

In the **Detail** tab, right-click a file to open a context menu with the following commands.

Command	Description
View	Opens the file by using the program that is associated to it. The program must be installed on the computer. NOTE: You cannot save changes in the content repository when you open a file by using this command.
Save As	Opens a dialog box, which lets you save a copy of the file by selecting a location and a file name.

NOTE: To be able to open a Control Participant project file (.stu) in a standalone installation of Control Expert, the version of Control Expert and of the installed EcoStruxure Control Expert DTM Library need to be the same as the Control Participant. For information on the installed version, refer to the platform release notes.

NOTE: For Control Participant project files (.stu), an application password is set and file encryption is enabled if a system access password, page 141 for the corresponding system or a Control constituent application password, page 1002 is set.

Temporary Contents Repository Files

Overview

The system server stores temporary files from the content repository inside a folder structure to access the files it needs faster. If needed temporary storage folders are created when you start the system server.

Temporary File Folder Location

The folder is named *Process Expert x*, where *x* corresponds to the major version of EcoStruxure Process Expert installation (the *R•* version suffix does not appear). It is located on the computer on which the system server is installed, at the path C:\Users\<UserName>\AppData\Local\Temp\Schneider Electric.

Reclaiming Disk Space

With time, the size of this temporary file folder increases.

You can reclaim disk space by deleting the subfolders and files inside the *Process Expert x* folder when EcoStruxure Process Expert is not running.

Appendices

What's in This Part

Linking Instances by Using the Links Editor	1126
---	------

Linking Instances by Using the Links Editor

What's in This Chapter

Links Editor 1127

Linking Instances by Using the Links Editor 1130

Links Editor

Overview

The **Links Editor** lets you link the exposed application interfaces of an instance to the interfaces of other instances that are compatible and respect the cardinality of the roles (for example, 1 to 1, or 1 to n). The editor also allows you to change or delete existing application links between instances.

You can use the **Links Editor**, for example, to map a signal to an application object such as a limit switch signal to the corresponding input of a valve motor.

Some instances require that you link them to other instances depending on the interface rules of their templates (for example, you need to link the instance of a motor starter template *\$TesySTMB* to a communication instance such as a port *\$MBPortQ*); otherwise the status of the **Link**, page 177 of the device instance remains **Invalid**.

Link any instance with an **Invalid Link**, page 204 status with the **Links Editor** to change the status to **Valid** since it is prerequisite to proceed further in the system engineering life cycle.

NOTE:

- Linking instances by using the **Links Editor** helps avoid having to create such links later by refining the Control Participant project and provides a level of consistency in terms of Control resource names.
- Linking an instance to itself is not allowed. If this is required, edit the template used by the instance or refine the Control project to which facets of the instance are assigned.

Naming Rules for Variable Names

The name of the variable that is created by linking two instances needs to satisfy the naming rules, page 32 for the subsequent generation process to complete successfully.

Instance and Links Appearance

In the **Links Editor**, instances appear as follows:

- **With a green outline:** Applies to the instance for which you have selected the **Edit Links** command and, which is referred to as *active*. Only one instance can be active in the same **Links Editor** tab.
- **With a gray outline:** Applies to instances that you can link, which are referred to as *inactive*.

NOTE: Active and inactive instances can be selected in which case a green border appears around them.

The following colors are used for application interface links:

- Orange: Default color.
- Blue: When an instance is selected to highlight links that are connected it.

Opening the Links Editor

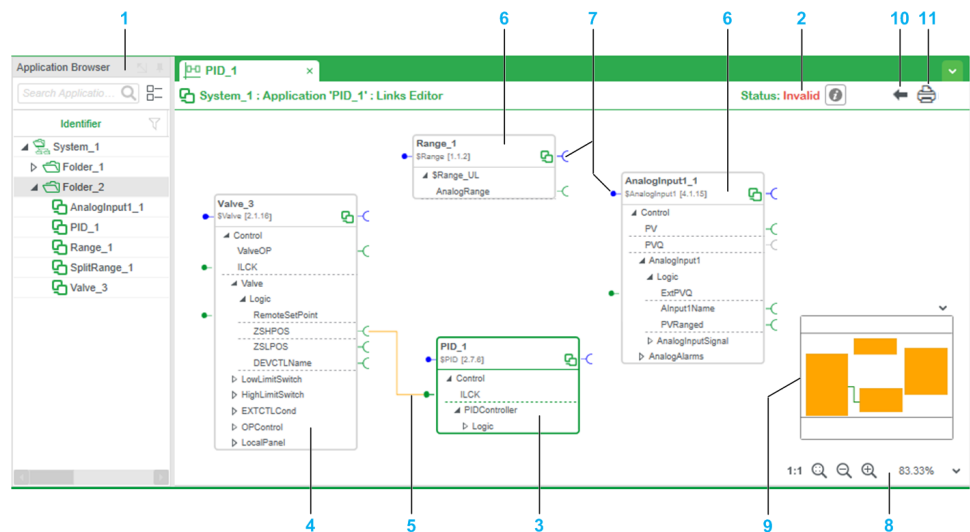
To open the **Links Editor**, right-click the instance that you want to link to one or more other instances, and select **Edit Links**.

Result: The **Links Editor** tab opens inside the work area, page 107 of the **Application Explorer** and displays the selected instance in the center of the editor. The instance appears as active, page 1127.

NOTE: If the active instance is already linked to other instances, these are shown and appear as inactive.

Links Editor

The following figure shows an example of the **Links Editor**, which is open in the work area of the **Application Explorer**.



Item	Description
1	Application Browser pane of the Application Explorer from which you can drag instances that you want to link.
2	Indication of the Link status of the active instance and button to display the Validity Diagnosis window if the status is Invalid .
3	Active, page 1127 instance from which you have selected the Edit Links command.
4	Inactive instance, which was already linked with the active instance or which you have linked.
5	Application interface link.
6	Inactive instances, which you have dragged from the Application Browser pane to the editor and which you can link with the active instance.
7	RTNS interfaces, page 208 (runtime navigation services). Appear in blue and are located in the header section of the instance.
8	Zoom tools, page 1023.
9	Radar View, page 1031
10	Reverts the view to the section in the Assignment Editor from where you have opened the Links Editor . The command is available only when you open the Links Editor from a Control facet. NOTE: The command does not close the Links Editor , which is required to perform the automatic generation of the section, page 407.
11	Lets you print the visible portion of the workspace.

NOTE: You can also move the editor background, page 108.

Impact of Modification of Instance Properties

When you edit an instance by using the **Instance Editor** and saving the change disables an interface that was linked, the link is automatically removed.

This may set the **Link** status, page 204 of either linked instances to **Invalid**.

Viewing Interfaces and Interface Description

Not all interfaces of instances that are displayed in the **Links Editor** are visible by default. You may need to expand nodes of the interface to show enabled (green) or disabled (gray) interfaces.

To enabled an interface, edit the instance and select the element that contains the interface, and/or configure the element parameter that enables it.

Move the pointer over any interface of an active or inactive instance in the **Links Editor** to display a description of the interface in a tooltip.

NOTE: An interface may be compatible, unbound, and enabled but cannot be connected, page 1130.

Viewing Linked Instances

Right-click a link between two instances and select **Go to**. A submenu opens and displays:


- The identifiers of the two instances that are linked.
- For each instance, the name of the element that exposes the interface and its path.

Click an entry to view the instance and highlight the element in the editor.

Linking Instances by Using the Links Editor

Creating Interface Links

To create an interface link between two instances, proceed as follows.

Step	Action
1	From the Application Browser pane, drag the instance that you want to link next to the active instance, page 1127, which is displayed in the Links Editor . Result: The instance is shown next to the active instance as inactive. NOTE: You can drag additional instances to the editor.
2	From the active or inactive instance, position the pointer over an interface and when the pointer turns into a hand, click and move the pointer away from it while holding down the left mouse button. Result: A green line appears and follows the pointer. You can release the mouse button. NOTE: Press ESC to return to the normal pointer function.
3	Move the pointer over the other instance. Result: The compatible and enabled, page 1130 interfaces of the instance are automatically highlighted and the green line snaps to the nearest matching connector.
4	Click to confirm the connection or move the pointer to another highlighted interface until the line snaps to it. Result: When you click: <ul style="list-style-type: none"> The interfaces of both instances are linked with a line. The status of the active instance changes to Valid if the interface rules are satisfied (if applicable). NOTE: To delete the link, right-click the line and select Delete .
5	Verify that the status of the active instance is Valid ; otherwise click the validity status information button  to display the Validity Diagnosis window and make the required links until the instance status becomes Valid . NOTE: You can only make links between an active and an inactive instance. To link the inactive instance with another inactive instance, right-click the header of the inactive instance and select Edit Links . The inactive instance becomes active.
6	Repeat steps 1...5 until you have made the required links.
7	Close the Links Editor tab.

Removing Instances from the Links Editor

You can remove inactive instances from the **Links Editor**.

To do so, right-click the header of the inactive instance that you want to remove and select **Remove Instance**. This also removes links that may exist with the active instance after you confirm the command.

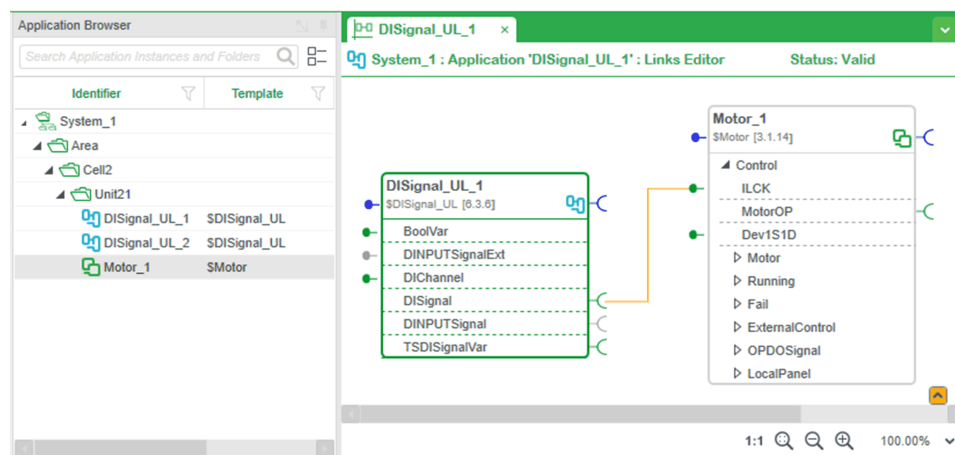
NOTE: Interface links that the instance that you are removing may have with other instances and that are not displayed in the this **Links Editor** tab are not removed.

Enabled Interfaces That Cannot Be Connected

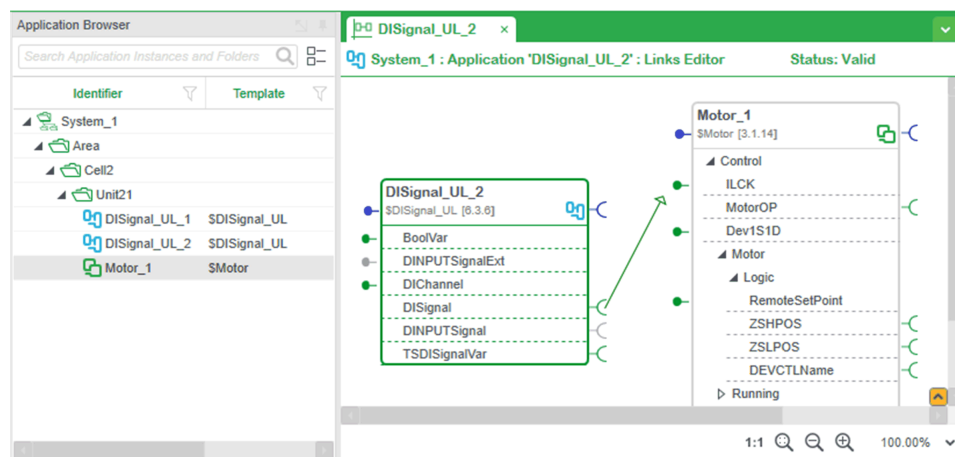
The interface of an inactive, page 1127 instance may appear as enabled, unbound, and be compatible but the link does not snap to it. This can be the case if a link with the interface exists but is not visible and the interface accepts only one connection at a time.

To view the links of an instance, open it in the **Links Editor** so that it is the active instance or open the **Inspect Instance** window, page 211.

In the following example, interface *ILCK* of *Motor_1* is connected to *DISignal_UL_1* and the **Links Editor** is closed.



When you open the **Links Editor** for *DISignal_UL_2* and drag *Motor_1* to the editor, its *ILCK* interface appears enabled. Yet, the link being created from *DISignal_UL_2* does not snap to *ILCK*. This is because *Motor_1* is an inactive instance and its links to other instances (*DISignal_UL_1*) are not shown in this view.



A

application interface:

Mechanism that allows sharing data and manage dependencies between 2 instances/references.

The *application interface* allows you to link:

- Application instances/references to each other
- Topological instances/references to each other

application link:

Links describing connections between instances of the application that are made by using application interfaces.

application template:

Global Template contained in the Global Templates library that models an object of the application of a system and its associated functions. The template encapsulates the necessary Participant capabilities.

Application templates are instantiated and can be configured to create the application of a system.

application:

Models the process to be controlled and monitored. It consists of instances of application templates and a application folder structure. A system can contain several applications.

area:

The *area*, when referring to access control, defines a topological, functional, or another user-based rule to restrict access in the scope of an automation system.

B

binding:

The *binding* is the key mechanism that establishes relations between the following items of the facet and composite templates:

- Parameters
- Interfaces
- Elements

built Participant project:

The *built Participant project* is a generated, optionally refined, and built project, which is created by the corresponding Participant and that exists at the platform level in the form of an executable.

For example, files with *.stu* and *.ctz* extensions that are represented by Control and Supervision executables respectively are built Participant projects.

C

cardinality:

The *cardinality*, in the context of the interface model, is the number of connections that are supported by the objects playing the other role of the interface model.

communication channel:

A *communication channel* is the logical representation, at the platform level, of communication between controllers or between a controller and devices.

For example, a controller I/O scanner row is modeled by a *communication channel* for Modbus TCP communication in the executable of the Control project.

communication interface:

Mechanism that allows sharing data and manage dependencies between 2 instances or references.

Communication interfaces allow the platform to link topological instances/ references at the logical level (for example, I/O scanner, OPC Factory Server, Supervision I/O devices).

communication link:

Links describing logical connections between topological instances that are made by using communication interfaces.

For example, the communication link between 2 controllers exchanging data.

communication mapping:

The *communication mapping* process defines the communication aspects of a logical Participant project after being mapped to the topology.

composite template:

The *composite template* combines the capabilities of 1 or more facet templates, each encapsulating functionalities provided by one software Participant, and/or of other *composite templates*.

composite:

The *composite* is an instance of a composite template.

constituent encapsulation:

Process that is performed by using the **Facet Editor** and that allows you to:

- Select constituents that are created with a software Participant.
- Optionally, modify constituents inside the corresponding Participant.
- Include the selected constituents inside a facet template.

constituent:

Set of data provided by a software Participant, which can be global or local.

contents repository:

The *contents repository* is a global storage provided by the platform to manage global constituents and other files, such as Participant projects, used by the libraries and the systems.

D

deferring mechanism:

Mechanism that allows you to make available in the composite template that is at the highest level optional element selections, parameters, and/or interfaces of its references.

deployed Participant project:

A *deployed Participant project* is a built project that has been transferred to a controller or other entity of the topology acting as an engine, and that you can execute.

DFB type:

The *derived function block type* is a programming language element that consists of:

- A data structure definition partitioned into input, output, public, and internal variables.
- A set of operations written in IEC 61131 programming languages to be performed upon the elements of the data structure when an instance of the function block type is started.

E**EcoStruxure Process Expert:**

Third generation name of the software platform. Formerly, StruxureWare Process Expert then, EcoStruxure Hybrid DCS.

element:

Element is the term that is used to describe the contents of templates at the next lowest level as well as the contents of interface models:

- For facet templates, elements are the constituents that the facet encapsulates.
- For composite templates, elements are:
 - Facet references
 - Composite references
- For interface models, elements are the data that is shared. See *interface elements*.

You can define properties and rules for elements during the Global Templates definition stage.

engine:

An *engine* is the projection of the hardware and software defined in the topology that is able to execute the deployed Participant project.

engineering client:

EcoStruxure Process Expert client connecting to the system server that you use to engineer and maintain systems. It can run on the system server and/or an engineering station.

Engineering clients use software Participants.

engineering station:

The *engineering station* is a computer running an EcoStruxure Process Expert engineering client to design, and maintain systems that exist on the server to which the engineering client connects.

executable:

The *executable* is a project component that represents the built Participant project. It contains the mapping information and can be deployed a topological entity.

execution domain:

Property of the executable of a Participant project, which serves as a filter for selecting the scope of applicable services.

For example, it is used to define the boundaries for runtime navigation services or to restrict the exchange of data through peer to peer communication between projects that have executables with the same execution domain.

F

faceplate:

The *faceplate* is a component of a Supervision genie that provides a user interface to interact with the object that it represents.

facet element:

The *facet element* is the contents that can be accessed at the system level. Depending on the type of facet template, such elements can be either:

- The visible part of the constituents stored in the facet templates.
- A set of data defined by the facet template.

facet template:

The *facet template* is the smallest available template that encapsulates a specific capability provided by 1 software Participant only.

facet:

The *facet* is an instance of a facet template.

G

genie:

Animated graphic that is assigned to Supervision pages and that represents functionalities of instances of the application during operation.

Genies are encapsulated in Supervision facets.

global constituent:

Set of data provided by a software Participant that is a common resource referenced by several Global Templates. Global constituents are stored in the contents repository.

Global Templates:

Global Templates encapsulate one or more functions that can be customized and reused in any system created with EcoStruxure Process Expert. They are stored in the Global Templates library.

H

hardware mapping:

The *hardware mapping* process links the logical projection of the hardware that is defined in the application and assigned to projects to the hardware defined in the topology of the system.

For example, linking Control variables representing I/O signals to I/O channels of an I/O module.

HMI:

Abbreviation for *human machine interface*. It is a graphical operator interface for industrial equipment.

I

IEC:

The *International Electrotechnical Commission* is a non-profit and non-governmental international standard organization that prepares and publishes international standards for electrical, electronic, and related technologies.

instance element selection:

The *instance element selection* is a mechanism of the instantiation stage that allows you to customize an instance by selecting services that are provided by the template that the instance uses.

instance parameter:

Instance parameters are properties of the elements of an instance that you may be able to customize.

instance:

Abbreviation for object instance. It is the result of the instantiation of a template.

instantiation naming convention:

The *instantiation naming convention* defines the naming strategy that the platform applies when you create instances.

interface element:

Represents the data that will be shared through an interface. Interface elements are defined during in the interface definition. Interface elements can be transformed by the interface by using expressions.

interface link:

Links describing logical connections between instances that are made by using interfaces.

The following types of links exist:

- Physical links
- Communication links
- Application links

Each type of link is made by using the corresponding interface (physical, communication, or application interface).

interface model:

The *interface Model* is a type of template that is available in the Global Templates library and that you can configure.

interface:

The *interface* is a reference, inside a composite or facet template, of an interface model. Interfaces are exposed by the facets of an instance, allowing you to make different types of links with other instances to share data.

In the context of the template definition, interfaces are a mechanism to define the links between references by declaring compatibility and/or requirement rules.

I/O:

Abbreviation for *Inputs/Outputs*.

L

local constituent:

Set of data that is provided by a software Participant, encapsulated inside a facet template but not used at the system level. The local constituent is used to generate the contents of the logical Participant project. An example is Control Participant code that is encapsulated in a Control facet template.

logical Participant project:

The *logical Participant project* is a generated and refined project, which is created by the Participant but, which is not associated to the topology.

NOTE: The refinement of the project is optional.

M

mapping interface:

Mechanism that allows sharing data between two facets.

Mapping interfaces allow you to perform the hardware mapping. It is the process whereby you link facets assigned to projects to facets representing the hardware defined in the topology through matching mapping interfaces that these facets expose.

MES:

The *Manufacturing Execution Systems* is a control system for managing and monitoring work-in-process in a factory plant.

N

network variable:

The *network variable* is a peer to peer communication mechanism allowing you to share data between 2 or more Control projects.

O

object template:

An *object template* is a generic term that covers several reusable templates such as facet or composite templates.

OFS:

Abbreviation for OPC Factory Server.

operation client:

An EcoStruxure Process Expert client connecting to the system server that you can use during runtime to monitor and troubleshoot a system.

operator station:

The *operator station* is a computer running a Supervision client software.

P

Participant services:

Participant services are the functions provided by a software Participant when interacting with EcoStruxure Process Expert.

Participant:

An external tool that is embedded in EcoStruxure Process Expert and installed on the system server and client computers. It allows refining projects and/or configuring topological entities. For example, the Control Participant.

peer to peer communication:

Peer to peer communication is a data exchange mechanism between 2 or more Control projects, which uses the I/O scanner function of the controller acting as client.

physical (interface) link:

Links describing logical connections between topological instances that are made by using physical interfaces.

For example, the connection of a controller to an Ethernet network.

physical connection:

Link between topological entities representing the physical connections between controllers, station nodes, devices, and communication networks.

physical interface:

Physical interfaces allow the platform to link, at a physical level, topological instances to model the topology of the system.

platform:

Abbreviation for system platform. Represents the services that are provided by EcoStruxure Process Expert apart from the software Participants.

privilege:

Defines groups of functions that are provided by an application and granted to users through access control.

project container:

The *project container* is an organizational structure of a project to organize the facets that are assigned to it. Such a structure models the ones managed by the corresponding software Participant, and that are visible at the system level.

project facet:

Facet that is assigned to a project.

project:

Component of a system associated to a Participant. It contains elements that are managed at the platform level and executables that you can deploy to topological entities.

R

redundant controller:

Generic term that is used to refer a Quantum Hot Standby controller and/or an M580 redundant controller.

reference:

Defines facet and composite templates, which are contained inside other facet or composite templates in order to distinguish templates, which are used in the composition of other templates from the highest level templates, such as control modules.

role:

The *role* can have 2 different meanings,

- For access control:

The *role* groups functionalities to grant different levels of user rights, which combine areas and privileges to fulfill a set of services.

- For interface models:

The *role* defines the 2 sides of an interface, role A and role B.

runtime navigation services:

Describes the complete set of functionalities that are provided by EcoStruxure Process Expert operation client in runtime, such as process monitoring, viewing of instance information, diagnostics.

S

service mapping:

The *service mapping* links the execution capabilities of a project represented by the executable to an engine of the topology.

For example, it can link:

- The I/O server of a Supervision project to a workstation modeling the operation server.
- The executable of a Control project to a controller.

service:

In the context of projects, the *project service* is an organizational structure of a project to organize the execution capabilities. It models the structures that are managed by the corresponding software Participant, and that are visible and configured at the system level.

In the context of templates, a *service* represents a functionality provided by a Participant under the form of a facet referenced by the template.

software:

Refers to EcoStruxure Process Expert.

station node:

A *station node (workstation)* models a computer that can act as an engine to execute a Control project by using the Controller Simulator or a Supervision project by using Supervision software.

subnet mask:

A 32-bit value that indicates the number of available host addresses on a subnet, which uses TCP/IP knowing that the first and last addresses are reserved (for example, subnet mask 255.255.192.0 allows 317 addresses for classless devices). It also indicates whether addressing is classful or classless. For classful addressing, it indicates the class of the network (for example, 255.255.255.0 is the subnet mask for class C networks).

Supervision client:

Client connecting to the Supervision server that provides runtime services to operate and monitor automation systems. The client runs on an operator station and/or operation server.

Supervision server:

Server running on the operation server or on the system server, and that provides runtime data to Supervision clients.

system engineering life cycle:

Represents the stages that must be completed to create and commission a system with EcoStruxure Process Expert.

system server:

EcoStruxure Process Expert server that hosts:

- Software Participants
- The database containing template libraries and system data

The system server manages requests from the EcoStruxure Process Expert clients.

system:

Models a physical automation system. It consists of the following components:

- Applications
- Control and Supervision projects
- The topology

T

topological entity:

A *topological entity* is the representation of piece of hardware infrastructure.

For example, a controller.

topology:

Models the hardware, software, and communication network infrastructure of a system by using topological folders and entities.

Index

\$EGenericDeviceHW template	
generic device mapping interfaces	748
\$GenericDevice	
implementing communication.....	937
\$GenericNOC_UL	
implementing communication.....	937
\$GenericPBSlave template	
generic device mapping interfaces	748
\$GenNOCDevice	
implementing communication.....	937
\$PRMMgt templates	
managing Modbus TCP communication (I/O	
scanning).....	941
\$System	
system parameters of instances	191
\$UnknownModule	
configuring controllers	591
1000 ID series	
analysis rule description	1095
10000 ID series	
analysis rule description	1106
2000 ID series	
analysis rule description	1096
3000 ID series	
analysis rule description	1098
4000 ID series	
analysis rule description	1100
5000 ID series	
analysis rule description	1101
6000 ID series	
analysis rule description	1102
7000 ID series	
analysis rule description	1104
9000 ID series	
analysis rule description	1105

A

aborting	
abort task icon	85
application import command	256
back up command.....	148
build command (control)	779
build command (supervision)	790
Deploy Built Project command (supervision	
workstation)	848
Deploy Built Project command (supervision)	843
Deploy Changes command (supervision)	851
device export command	661
device import command.....	668
export command	229
export command (control)	279
export command (supervision)	321
export command (Topology).....	649
folder delete command	171
Generate and Build command for projects	399
Generate command for project containers.....	428
Generate command for projects	427
import command (control).....	292
import command (supervision)	331
import command (Topology).....	657
instance delete command	179
paste command	214
Purge command	1050
ReGenerate command for projects	435
restore command	153

template export command	1039
template import command	1047
Update Project command	905
About Process Expert	
About dialog box	106
opening the about dialog box	104
Access Control View description.....	556
actions	
entity context menu commands	570
on assigned facets	390
on generated facets	442
activity	
log files	92
notification panel.....	85
addresses	
generating network addresses for supervision	
components	789
identifying engines	796
advanced settings	
advanced communication attribute	
descriptions	705
Advantys	
selecting templates	584
Advantys islands	
creating STB islands	599
alarm server redundancy	
implementing	315
Altivar	
basic diagnostic	1118
diagnostic status	1114
analysis	
analyzing configuration changes (M580 safety) ..	924
configuring controllers	1091
Analyze command.....	570
analysis rules.....	1094
analysis rules - 1000 series.....	1095
analysis rules - 10000 series.....	1106
analysis rules - 2000 series.....	1096
analysis rules - 3000 series.....	1098
analysis rules - 4000 series.....	1100
analysis rules - 5000 series.....	1101
analysis rules - 6000 series.....	1102
analysis rules - 7000 series.....	1104
analysis rules - 9000 series.....	1105
analyzing communication configurations and	
hardware	1091
analyze issues	
troubleshooting control Participant projects	449
validity indicators	777
Analyze Project	
automatic generation	
automatic generation setting	272
control project settings.....	272
generation	
automatic generation setting	272
analyzing	
control projects during generation.....	401
Application Browser pane	
presentation	166
using and customizing	166
Application Explorer	
accessing	160
opening	104
presentation	160
application export files	
using the invariant culture	236
application folders	
actions	171
changing folder structure	216

changing the application hierarchy	213	backing up systems.....	145
copying and pasting	214	scheduling system backups	150
creating	170	backing up data	
managing	210	backing up controller data	880
moving	216	backup	
properties	172	password information and settings in system	
application objects		backups.....	147
exporting and importing to XML or CSV	227	backup files	
application passwords		managing data backup files (control)	296
changing the application password of control		basic diagnostics	
projects	809	basic diagnostic for Altivar Process devices	1118
clearing the application password of control		basic diagnostic for devices	1118
projects	810	basic diagnostics window description	558
managing passwords of control projects	804	batch	
resetting a forgotten application password	810	modeling the equipment hierarchy	159
setting the application password of control		BKP	
projects	808	attribute description	682
application structure		BMENOC03•1	
tree view description in Containers pane	384	operating states	1112
tree view when refining Control projects	448	parameter correspondence with Topology	
tree view when refining Control projects online ...	875	Explorer.....	522
application templates		BMENOC0321	
description.....	42	configuring IP forwarding in Topology Explorer ...	682
applications		BMENOP0300	
generating system documentation reports.....	1083	configuring ETH1_Interface in Topology	
import rules for control project-related objects	282	Explorer.....	679
import rules for supervision project-related		configuring ports in Topology Explorer	682
objects	326	BMENOS0300	
modeling processes in systems.....	157	configuring ports in Topology Explorer	682
Asset Workspace Editor	200	BMENUA0100	
using the radar view	1031	configuring Control Port	684
using the simplified view	203	configuring interfaces	679
Asset Workspace pane	193	configuring NTP service.....	698
asset workspaces		configuring OPC UA in Topology Manager	707
exporting	227	parameter correspondence with Topology	
assigned facets		Explorer.....	522
generating automatically assigned facets	399	variable tag addresses	788
managing assigned facets	389	boards	
unassigning generated facets	443	creating communication boards (supervision).....	312
assigning		bold formatting	
Assignment Viewer description	372	editing instance properties	188
defining container names.....	363	browsers	
facets	362	Templates Browser presentation	164
facets (control).....	375	build command	
facets (supervision)	377	aborting (control).....	779
Assignment Editor		aborting (supervision).....	790
presentation	343	build stage	
assignment stage		completing.....	773
completing	340	concept	65
concept	56	control projects	774
Assignment Viewer		supervision project	787
description.....	372	build states	
View Assignments window.....	392	when deploying (supervision).....	843
assignments		build status	
automatic assignment of remaining facets.....	381	actions changing the build status (control).....	781
automatic assignment of Supervision facets.....	383	building the logical supervision project	790
viewing by instance	392	changes affecting the build status (control).....	779
asterisk		verifying before deploying (control).....	814
Synchronize Changes command	590	when deploying from workstation (supervision)...	846
audit trail		building	
functionality description	88	building safety control projects	930
authentication		creating the built Control Participant project	775
logging into server and clients	72	network addresses of supervision components...	789
B		building changes (control)	
back up command		building with device DDT variables	782
aborting.....	148	building control projects	
backing up		building with device DDT variables	777
		Firmware and Web Diagnostic/Data Storage	
		passwords	775

Generate and Build command	777	color coding	
project settings	777	importing application objects	258
building supervision projects		importing devices from CSV	669
Generate and Build command	790	color indicators	
built Participant projects		instance color icons	195
changing (control)	781	project color icons	333
changing (supervision)	792	colors	
C		representation of port types	539
calculated variable tags		commands	
I/O device properties	310	entity context menu commands	570
canceling		communicating	
canceling generation for version conflict		with devices through Ethernet explicit	
(control)	402	messaging	944
capabilities		with devices through EtherNet/IP explicit	
hardware	24	messaging	949
programming	24	with devices through Modbus explicit	
catalog		messaging	953
supported equipment	516	communication	
CCOTF	585	changing EtherNet/IP device properties	605
refining projects offline	446	changing Modbus TCP device properties	605
using CCOTF with S908 architectures	585	changing STB islands properties	600
certificate validity		communicating with the computer for deployment	
About dialog box	106	(control)	803
changes		displaying device communication status	1107
actions changing the build status (control)	781	EtherNet/IP explicit messaging	949
deploying changes in hardware mappings	834	implementing I/O scanning	936
deploying changes to controllers (control)	830	implementing I/O scanning with devices	937
managing hardware mappings after deploying ...	753	implementing I/O scanning with PRMs	941
saving changes in Manage Network Variables dialog		implementing I/O scanning with STB islands	940
box	473	implementing peer to peer communication	943
changes log		local slave communication workflow	645
accessing	1028	managing	935
interfaces	1009	mapping (control)	725
saving changes in Global Templates	1035	Modbus explicit messaging	952
CHM help		Modbus TCP explicit messaging	944
HTML help for user-created templates	98	task workflow	126
using the HTML help for several versions of		Variable Mapping Editor window	738
templates	96	communication boards	
circular links		creating (supervision)	312
generating	402	driver selection	312
citect.ini files		properties (supervision)	312
deploying	841	communication channels	
deploying from workstation	846	actions	733
deploying to workstations from executable	843	adjusting properties	730
Client Memory Length		creating	730
attribute descriptions	685	mapping stage (control)	729
Client Memory Start		moving	732
attribute descriptions	685	properties	733
clients		properties server side	736
connection to the server	76	unmapping	731
creating (supervision)	308	communication mapping	
engineering client UI description	103	Communication Mapping Editor	726
logging in and out	72	I/O Mapping Editor description	547
notification panel	85	prerequisites (control)	725
quitting	80	with Modbus serial devices	769
restarting after closing unexpectedly	78	with Modbus TCP devices	769
supervision client properties	308	communication mapping views	
working with several clients	82	description	544
closing		communication module	
closing the software	80	removing from the configuration	593
clusters		communication ports	
actions	305	creating (supervision)	312
creating	305	creating additional ports (supervision)	313
properties	305	properties (supervision)	313
collapsing		communication profiles	
nodes in the Application Browser pane	168	configuring I/O profiles	638
nodes in the instance pane	185	configuring I/O scanning	641
		creating and configuring	709
		Compare command	570

components		owner and consumer projects463, 471
creating supervision project components 304		consumer project
Composite Editor		description..... 463
changes log 1028		Container Naming Rules
menus and toolbars..... 1022		window description..... 363
panes..... 1028		containers
parameter pane 1028		creating and managing 341
presentation 1018		Containers pane
using the radar view 1031		displaying sections in tree view 345
using the simplified view 1032		content
composite templates		resolving contents conflicts on import 1046
changes log 1028		content repository
description..... 38		actions 1123
modification strategy 1052		content description..... 1121
configuration		deleting temporary files 1124
changing EtherNet/IP device properties 605		explorer..... 1120
changing Modbus TCP device properties 605		opening 104
changing STB islands properties 600		root folder actions 1122
changing the controller configuration 593		User Contents folder 1120
changing the PRM configuration..... 602		Content Repository explorer
configuring M580 safety controllers 923		opening..... 1120
Configuring modules with Topology Explorer 519		context menus
deploying hardware changes 833		entity context menu commands..... 570
Rack Viewer description 559		Control Constituent Password
updating configuration after deploying changes		managing Control facet template passwords 1002
online 875		Control Expert Classic
updating controller configurations..... 897		parameter correspondence with Topology
updating hardware changes..... 900		Explorer..... 522
viewing instance configuration 211		Control Expert projects
configuration change on the fly 585		importing a Control Expert project 294
configuration file		Control facet templates
OFS/OPCUA configuration file description 788		managing Control facet template passwords 1002
Configuration File command..... 570		Control Facet Templates
configuration files		Global Template settings 1002
managing entity configuration files..... 676		control modules
configuration stage		description..... 38
completing..... 512		template modification strategy..... 1052
concept..... 61		Control Project Browser
Configuration tab		description..... 266
attributes of the Configuration tab 676		control project sections
configuring		actions 348
configuring HART modules 606		control projects
instance parameters..... 188		actions 269
connecting		build stage..... 774
connecting entities by physical links 608		changing built Participant projects 781
connection		communication mapping import rules..... 284
client-server connection..... 76		consistency management 460
redundant configuration connection scenarios.... 892		creating 269
connections		creating FBD and non-FBD sections 273
deleting physical links..... 613		creating FBD sections 348
EtherNet/IP connection attribute descriptions 714		creating M580 safety-related control projects 912
logical network connection import rules..... 654		creating the built Participant project 775
physical connection import rules..... 654		data backup file import rules 288
physical link properties 614		deleting 271
consistency		deploying 794
managing consistency of system topology 1090		deploying changes to controllers 830
consistency checks		description of tree view mode 384
after import..... 280		executable import rules 284
before updating control projects 903		executing 855
code lock feature..... 452		export settings 278
fixing inconsistencies 462		exporting 276
performing (control)..... 462		facet import rules 283
rules 460		generating after changes 430
consistency management		generation specifics 401
generating inconsistent facets 432		hardware mapping import rules 284
refining control projects 460		import settings 281
consumer		importing 280
about network variables..... 463		importing a Control Expert project 294
managing network variables 471		managing 274

managing passwords of control projects	804	deploying data	824
mapping stage	720	deploying to redundant controllers	812
parameter correspondence with Topology Explorer	522	disabling the I/O scanner	593
peer to peer communication export settings	279	disconnecting from logical networks	618
peer to peer communication variable import rules	287	execution with redundant controllers	859
prerequisites	269	exporting	647
Project Explorer	268	import rules	653
Project Explorer and Browser	263	importing	651
project import window	289	managing data backup files	296
properties	270	managing network variables	471
refining	446	mapping (control)	723
refining locked code	452	memory size configuration	592
regenerating	434	properties	592
related project data import rules	282	redeploying built control projects	819
section import rules	282	representation	595
selecting type version changes	409	representation of devices	578
service mapping import rules	284	selecting templates	584
starting the simulator with your control project	800	setting the application password of control projects	808
starting the simulator without password protection	801	starting and stopping	861
task workflow	126	supported platforms	24
tasks	130	copying	
tree view mode when refining	448	copy-paste application folders	215
tree view mode when refining online	875	copy-paste instances	215
updating projects after deploying changes online	875	copying entities in a physical view	539
Control projects		copying HART field devices	555
managing system access passwords	141	copying PROFIBUS field devices	555
Control service		Global Templates and folders	1038
creating in workstation	597	copying instances in Control projects	390
ControlExecutables		counting	
actions	721	counting instances	180
assigning execution domains	722	CRA modules	
creating	721	connection with device I/Os	936
properties	722	Create Instance dialog box	351
controller actions	570	creating	
controller module		Global Template editors	1007
replacing the controller module	594	creating M580 safety-related control projects	912
controller modules		CSV	
operating states of M580 controllers	1112	application export file description	231
controller passwords		editing the application export file	235
resetting a forgotten controller password	810	exporting and importing application objects to XML or CSV	227
controller stop		exporting and importing devices	660
when deploying changes	840	importing CSV files in the topology	667
ControllerExist		importing CSV files into the application	254
attribute descriptions	689	csv device export files	
controllers		using the invariant culture	664
about network variables	463	CSV export files	
adding entities to systems	565	using the invariant culture	236
analyzing configurations	1091	ctz files	
assigning IP addresses	592	changing (supervision)	792
backing up controller data	880	creating (supervision)	787
changing the application password of control projects	809	deploying	841
changing the configuration	592	deploying changes to workstations	851
clearing the application password of control projects	810	deploying from workstation	846
communicating with the computer	803	deploying to workstations from executable	843
configuration limits	591	redeploying to engines from executable	844
configuring	592	redeploying to engines from workstation	850
configuring the I/O scanner	595	cutting	
connecting to logical networks	617	cutting entities in a physical view	539
creating	591		
creating the system topology	562	D	
deleting	596	data	
deploying built projects	814	backing up controller data	880
deploying changes (control)	830	deploying controller data to engines	824
		instance validity status	186
		managing data backup files (control)	296
		sorting data	120

using filters	120	to redundant controllers	812, 814
data backup		to workstations from executable (supervision)	843
with redundant controllers	880	deploying changes	
data backup files		changes requiring to stop the controller	840
import rules for data backup files (control)	288	deploying device DDT variable changes	836
managing data backup files (control)	296	hardware mappings	834
data deployment		main redundant configuration scenarios	889
to redundant controllers	824	selecting IP addresses	831
Data Storage password		to redundant controllers	813, 831
building control projects	775	to workstations (supervision)	851
refining control projects	447	deploying control projects	
regenerating control projects	434	supported engines	796
data storage passwords		deployment	
changing the Webuser password	588	communicating with the computer (control)	803
Data Storage passwords		process flowchart	795
recovering passwords	589	deployment stage	
data types		completing	793
conversion to Participant types	28	concept	67
data types used by the software	28	control projects	794
validating instance parameters	188	supervision project	841
DDDT		Description tab	
building changes in control projects	782	attributes of the Description tab	675
building control projects	777	device DDT	
creating during build	776	building changes in control projects	782
deploying changes to control projects	836	building control projects	777
for peer to peer communication	776	creating during build	776
managing device DDT variables	455	deploying changes to control projects	836
naming convention	776	for peer to peer communication	776
naming convention for P2P communication	776	naming convention	776
DDTs		naming convention for P2P communication	776
device DDT type and instance names	710	device DDT variables	
device DDTs in the configuration	591	managing device DDT variables	455
deferring parameters automatically		device DDTs	
Global Template settings	1002	device DDT of scanned devices	641
delete command		device DDT type and instance names	710
aborting folder delete	171	device DDTs in the configuration	591
aborting instance delete	179	device export files	
deleting		using the invariant culture	664
control projects	271	DeviceConfig.xml file	
controllers	596	creating	787
deleting entities	570	deploying	841
deleting FBD sections during update Control		saving	764
project	904	devices	
deleting I/O items	644	adding devices to systems	565
deleting logical networks	533	analyzing DTM usage	633
deleting physical links	613	assigning IP addresses to topological I/O device	
deleting physical views	534	entities	604
PRMs	602	basic diagnostic for devices	1118
sections	348	changing IP addresses in explicit messaging	948, 951
STB islands	600	communicating through Ethernet explicit	
unmapping communication channels	731	messaging	944
unmapping hardware mappings	752	communicating through EtherNet/IP explicit	
deleting modules	596	messaging	949
Deploy Built Project command		communication through Modbus explicit	
aborting (supervision workstation)	848	messaging	953
aborting (supervision)	843	configuring I/O profiles	638
Deploy Changes command		configuring I/O scanning	641
aborting (supervision)	851	connectivity with supervision	984
deployed control projects		creating EtherNet/IP devices	603
editing online	874	creating Modbus TCP devices	603
deploying		displaying device communication status	1107
controller data to engines	824	exporting	660
deploying changes to controllers (control)	830	Identification mismatch icon	1111
deploying to M580 safety controllers	931	importing CSV files in the topology	667
from workstation (supervision)	846	integrating PROFIBUS devices	988
Identification mismatch icon	1111	managing Modbus TCP communication (I/O	
main redundant configuration scenarios	889	scanning)	937
preparing for deployment	842	mapping I/Os (control)	747
to engines (control)	814		

properties of EtherNet/IP device entities	604	updating	636
properties of Modbus TCP device entities	604	duplicating	
representation of devices	578	Global Templates	1065
resolving import conflicts	671	E	
selecting topological templates	584	Edit/Extend Interface	
using device DTMs	629	command description	1070
DHCP		editing	
attribute descriptions	694	deployed control projects online	874
DHCP server		editing instance properties	
subscribing devices	624	bold formatting	188
DHCP service		italic formatting	188
assigning devices to DHCP servers	624	validity status	188
unsubscribing devices from DHCP service	624	editing instances	
DHCP view		changing element selection after assignment	380
description	550	editing interfaces	
Diagnosis		changes log	1009
command description	570	editing online (control)	
diagnostic messages		changes requiring to stop the controller	840
during generation	440	overview	870
Diagnostic tab		editors	
attributes of the Diagnostic tab	717	Assignment Editor	342
diagnostics		Composite Editor	1018
Altivar Process devices	1114	Facet Editor	1014
basic diagnostic for Altivar devices	1118	Interface Editor	1008
basic diagnostic for devices	1118	Links Editor	1127
displaying device communication status	1107	opening Global Template editors	1007
generic automation devices	1116	EDS library	
Hirschmann switches	1117	exporting and importing	631
Identification mismatch icon	1111	elements	
M580 controllers	1112	definition	38
TeSys devices	1115	replacing templates of elements	1060
DIO ports		updating templates of elements	1054
color coding	539	email	
DIO scanner		attribute descriptions	706
implementing communication by I/O scanning	936	Embedded Interface	
disk variable tags		attribute description	679
I/O device properties	310	End User License Agreement	
docking menus		viewing the End User License Agreement	106
creating additional work areas	110	engineering client	
documentation		engineering client work areas	107
application data	1083	engineering client toolbar	104
generating system documentation	1080	engineering clients	
Global Templates data	1088	engineering client UI description	103
project data	1084	engines	
topological data	1086	backing up controller data	880
downloading		deploying built control projects	814
controller data to engines	824	deploying data (control)	824
deploying changes to controllers (control)	830	deploying to redundant controllers	812
deploying to engines (control)	814	execution with redundant controllers	859
main redundant configuration scenarios	889	managing data backup files (control)	296
redeploying to engines (control)	819	mapping (control)	723
to redundant controllers	812	redeploying built control projects	819
dragging		refreshing engine status and connecting	803
moving the workspace around	108	supported engines for deployment (control)	796
drivers		entities	
creating communication boards and ports	312	adding to a physical view	568
DTM		connecting devices by physical links	608
analyzing DTM usage	633	connecting entities by Ethernet links	608
configuring I/O profiles	638	creating and configuring logical networks	576
cutting, copying, and pasting	632	creating and configuring topological entities	512
exporting and importing	632	displaying entity attributes	538
installed DTM information	688	entity context menu commands	570
naming convention	776	moving entities between physical views	569
naming convention for P2P communication	776	moving entities inside a physical view	568
using device DTMs	629	entity actions	570
DTM Reference		entity identifiers	
creating and configuring local slaves	709	naming rules	31
DTMs			
analyzing DTM usage	633		

entity nodes		backing up controller data	880
description	529	creating (control)	721
enumerations		creating (supervision)	764
exporting parameter values to CSV	236	deploying changes to controllers (control)	830
exporting parameter values to XML	246	deploying changes to workstations	
viewing values in the Instance Editor	183	(supervision)	851
EPE Managed Devices		deploying controller data to engines (control)	824
representation in physical views	578	deploying from workstation (supervision)	846
equipment		deploying to engines (control)	814
adding to a physical view	568	deploying to redundant controllers	812
equipment hierarchy	159	deploying to workstations (supervision)	843
generating equipment (supervision)	437	executable import rules (control)	284
product types and product ranges	517	executable import rules (supervision)	329
representation of devices	578	executing with redundant controllers	859
equipment generation		managing data backup files (control)	296
troubleshooting	440	properties (control)	722
equipment hierarchy		properties (supervision)	765
import rules for equipment (supervision)	327	redeploying to engines (control)	819
errors		redeploying to engines (supervision)	844, 850
detected errors during generation	402	executables (supervision)	
ETH•		status change when importing new content	
attribute description	682	version	1046
Ethernet		executing	
conflicting Ethernet connections	616	deployed participant projects (control)	861
connecting controllers to logical networks	617	to redundant controllers	859
disconnecting controllers from logical networks	618	execution	
managing EtherNet/IP explicit messaging	949	facet execution order	394
managing Modbus TCP explicit messaging	944	FBD section execution order	275
Ethernet Interface		process flowchart	856
interface description	679	synchronizing redundant controllers	866
Ethernet IP devices		execution domains	
import rules	653	assigning to executables (control)	722
Ethernet links		assigning to executables (supervision)	766
connecting entities	608	execution order	
deleting	613	after first generation	403
properties	614	when regenerating	435
Ethernet networks		execution stage	
exporting	647	completing	854
importing	651	concept	68
Ethernet RIO drops		control projects	855
connecting to logical networks	617	supervision projects	907
disconnecting from logical networks	618	expanding	
EtherNet/IP		nodes in the Application Browser pane	168
configuring I/O profiles	638	nodes in the instance pane	185
configuring I/O scanning	641	explorers	
using device DTMs	629	Application Explorer	160
EtherNet/IP connections		Content Repository explorer	1120
creating and modifying I/O items	642	creating work areas in explorers	111
EtherNet/IP devices		Global Templates Explorer	992
changing properties	605	opening	104
creating and configuring devices	603	opening the Project Explorer	264
EtherNet/IP explicit messaging		overview	23
communicating with devices	949	Systems Explorer	135
managing communication	949	task workflow	126
EULA		export command	
viewing the End User License Agreement	106	aborting	229
Exclude command		aborting (control)	279
description	1015	aborting (supervision)	321
executable files		aborting (Topology)	649
changing (control)	781	aborting device export	661
creating (control)	775	aborting template export	1039
creating (supervision)	787	export files	
deploying (supervision)	841	editing the CSV application export file	235
modifying (supervision)	792	editing the XML application export file	246
executables		editing XML to manage application links	251
actions (control)	721	using the invariant culture	236
actions (supervision)	764	using the invariant culture (devices)	664
assigning execution domains (control)	722	export settings	
assigning execution domains (supervision)	766	control projects	278

peer to peer communication export settings	279	moving generated facets	444
supervision projects	321	navigating to the instance	395
exporting		reassigning	444
CSV application export file description	231	relinking	444
editing exported hardware mappings	758	unassign	391
EDS library	631	unassigning generated facets	443
exporting application objects to XML or CSV	227	unlinking generated facets	443
exporting control projects	276	viewing assignments per instance	392
exporting hardware mappings	755	FBD	
exporting physical view layouts	538	using FBD language with Process Expert	24
GSD library	631	FBD sections	
password related information for topological		actions	348
entities	648	changing execution order	275
peer to peer communication data	276	creating during generation	403
supervision projects	318	creating FBD sections	348
templates	1039	deleting	348
topological entities	647	import rules for FBD sections	282
XML application export file description	244	inserting FBD sections	348
exporting asset workspaces	227	properties	349
exporting control projects		when regenerating	435
software version compatibility	276	FDR	
exporting Control projects		attribute description	696
peer to peer communication	491	FdtDtm settings	
exporting devices to CSV file	660	description	631
exporting M580 safety controllers	927	fiber optic connector representation	582
exporting physical views	648	field devices	
exporting safety-related control projects	914	acting on HART field devices	555
exporting supervision projects		acting on PROFIBUS field devices	555
software version compatibility	318	fieldbus editor	
expressions		acting on HART devices	555
Container Naming Rules window	363	acting on PROFIBUS devices	555
importing expressions for control sections	281	HART fieldbus editor description	553
extracted Participant windows		PROFIBUS fieldbus editor description	553
troubleshooting	119	file encryption	
extracting		applying to deployed control projects	144
software Participant windows	118	files	
		adding external files to supervision projects	511
F		filters	
Facet Editor		filtering data	120
changes log	1028	template editor element filters	1030
description	1014	Templates Browser	163
menus and toolbars	1022	to display instance parameters	185
panes	1028	Firmware password	
parameter pane	1028	building control projects	775
using the radar view	1031	refining control projects	447
using the simplified view	1032	regenerating control projects	434
facet template passwords		floating Participant windows	
changing the password for Global Templates ...	1004	troubleshooting	119
clearing the password for Global Templates	1004	folder structure	
facet templates		changing application folder hierarchy	172
changes log	1028	changing the position of folders	219
description	38	folders	
modification strategy	1052	changing the application hierarchy	213
facets		copying and pasting application folders	214
actions after assigning	390	copying and pasting library folders	1038
actions after generation	442	creating Global Template folders	998
assigned facet description	379	deleting topology folders	531
assigning	362	exporting and importing to XML or CSV	227
assigning (control)	375	moving application folders	216
assigning (supervision)	377	moving folders	172
automatic assignment of remaining facets	381	naming rules	31
automatic assignment of Supervision facets	383	resolving application import conflicts	260
changing order inside section	394	S88 representation	176
element selection changes of assigned facets	380	searching application instances and folders	161
generating automatically assigned facets	399	topology folder actions	531
import rules for control project facets	283	function blocks	
managing assigned facets	389	layout after first generation	403
moving	391	layout when generating after changes	433
		layout when regenerating	435

functional safety password	
managing the functional safety password	925

G

gateways	
gateway address of logical networks	577
using for Modbus explicit messaging	953
Generate and Build command	
aborting	399
Building control projects the first time	777
Building supervision projects the first time	790
description	399
Generate command	
aborting generation of project containers	428
aborting generation of projects	427
generated code	
identifying software-generated code	452
generated facets	
unlinking	443
generating	
analyzing control projects	401
automatically assigned facets	399
configuring settings (control)	405
control project specifics	401
detecting errors during generation	402
diagnostic messages	440
entire projects	427
equipment (supervision)	437
facet order	394
FBD section size	403
first-time generation	426
function block layout during first generation	403
function description	430
general description	397
Generate and Build command	399
generating invalid facets again	433
generating safety-related control projects	920
importing sections after generation	429
invalid facets	400
part of a project	428
projects after changes	429
rules for generating graphical links	402
satisfying naming rules	398
section order	275
statuses of facets	424
generation	
canceling generation for version conflict	
(control)	402
changing type versions during generation	
(control)	409
function block layout after changes	433
managing generated facets	442
skipping facets during generation	400
generation name issues	
troubleshooting	440
generation stage	
completing	396
concept	58
generating inconsistent facets	432
GenerationSettings	
configuring generation XML file (control)	405
generic devices	
communicating through Ethernet explicit	
messaging	944
communicating through EtherNet/IP explicit	
messaging	949
communication attributes	709

communication through Modbus explicit	
messaging	953
connectivity with supervision	984
creating EtherNet/IP devices	603
creating Modbus TCP devices	603
Device DDT configuration	709
diagnostic status	1116
diagnostics attributes	717
diagnostics configuration	717
managing Modbus TCP communication (I/O	
scanning)	937
mapping interfaces	748
using device DTMs	629
genies	
editing supervision pages	509
Global Template editors	
opening	1007
Global Templates	
actions	1001
classification	38
copying and pasting	1038
description	38, 1000
duplicating	1065
generating system documentation reports	1088
Global Templates Explorer	992
managing	1035
managing Control facet template passwords	1002
naming convention	993
purging the Global Templates library	1049
renaming	1035
replacing templates of elements	1060
saving changes	1035
updating	1054
updating at folder level	1058
Global Templates definition	
concept	48
Global Templates Explorer	
folder actions	999
opening	104, 992
presentation	992
Toolbox	994
Global Templates library	
viewing in the Global Templates Explorer	992
Global Templates passwords	
resetting a forgotten Global Templates	
password	1005
Go To Instance command	
navigating to the instance	395
graphical links	
generation rule	402
GSD library	
exporting and importing	631

H

HAL DFBs	
managing	784
HAL variables	
managing	784
HAL variables and DFBs	
properties	785
viewing	786
hardware	
adding to a physical view	568
adding to systems	565
configuration stage	512
creating the system topology	562
deploying hardware changes	833
mapping	747

supported components	562	ID	
updating hardware changes	900	analysis rules	1094
hardware abstraction layer variables		identifiers	
managing	784	changing the identifier of Global Templates	1035
hardware mapping		naming rules	31
managing HAL variables	784	identifying engines	
Hardware Mapping Editor		using physical addresses	796
presentation	748	IL	
hardware mappings		using IL language	24
editing exported hardware mappings	758	implicit messaging	
exporting hardware mappings	755	configuring I/O profiles	638
importing hardware mappings	761	configuring I/O scanning	641
managing changes after deploying	753	creating and modifying I/O items	642
unmapping hardware mappings	752	import	
HART		accessing the import window	654
acting on HART devices	555	import command	
HART communication	606	aborting (control)	292
HART fieldbus editor		aborting (supervision)	331
description	553	aborting (Topology)	657
HART modules		aborting application import command	256
configuring HART modules	606	aborting device import command	668
representation	607	aborting template import	1047
help		import rules	
help button	104	executable import rules (control)	284
HTML help for user-created templates	98	executable import rules (supervision)	329
opening the help of software Participants	95	for control project facets	283
using the HTML help for several versions of		for control project-related objects	282
templates	96	for data backup files (control)	288
Hide Deferred filter	1030	for equipment (supervision)	327
Hide Unbound filter	1030	for Included projects	325
Hide Unchecked filter	1030	for mapped peer to peer communication	
hierarchical naming	189	variables	287
highlighting redundant networks	539	for pages of supervision projects	327
Hirschmann switches		for program sections (control)	282
diagnostic status	1117	for supervision page templates	327
hot standby		for supervision project facets	327
managing data backup files	296	for supervision project-related objects	326
Hot Standby		for topological entities	653
attribute descriptions	689	for topological folders	653
hot standby controllers		logical network connection import rules	654
backing up controller data	880	mapping import rules (control)	284
connecting to logical networks	616	mapping import rules (supervision)	329
deploying built control projects	814	import settings	
deploying changes to redundant controllers	831	control projects	281
deploying controller data	824	supervision projects	324
deploying to redundant controllers	812, 814	importing	
disconnecting from logical networks	618	application import settings	258
execution	859	checking consistency after importing control	
logical network connection import rules	654	projects	280
main deployment and execution scenarios	889	color coding when importing application	
redeploying to redundant controllers	819	objects	258
reservation and connection scenarios	892	color coding when importing devices from CSV	669
Hot Standby controllers		Control Expert projects	294
representation of devices	578	control projects	280
HTML help		EDS library	631
HTML help for user-created templates	98	expressions for control project sections	281
using the HTML help for several versions of		GSD library	631
templates	96	importing application objects to XML or CSV	227
icons		importing CSV files in the topology	667
Altivar device diagnostic statuses	1114	importing devices from CSV file	660
assigning icons to folders	176	importing hardware mappings	761
generic automation device diagnostic		importing XML or CSV files into the application	254
statuses	1116	password related information for topological	
Hirschmann switch diagnostic statuses	1117	entities	654
module operating states	1112	resolving contents conflicts for templates	1046
TeSys device diagnostic statuses	1115	sections during online refinement	885
		supervision projects	323
		templates	1042
		topological entities	651

importing control projects		instantiating	
software version compatibility	280	templates	158
importing Control projects		instantiation	
peer to peer communication	491	order when creating instances	177
importing devices		structure of the application	159
resolving conflicts	671	instantiation stage	
importing instances		completing	157
resolving conflicts	260	concept	49
importing M580 safety controllers	927	interface links	
importing physical views	651	editing the XML export file	251
importing safety-related control projects	914	exporting and importing to XML or CSV	227
importing supervision projects		interface models	
software version compatibility	323	types	44
included projects		interfaces	
adding or removing	510	changes log	1009
downloading	791	changing logical network assignments	627
saving (supervision)	764	editing/extending interfaces	1070
Included projects		generic device mapping interfaces	748
importing	325	mapping hardware interfaces	747
importing with templates	1042	modification strategy	1052
resolving contents conflicts on import	1046	invalid	
inconsistent facets		instance data validity status	186
generating	432	invalid facets	
layout of blocks after generation	433	generating	400
infrastructure		generating invalid facets again	433
creating the system topology	562	invalid link status	204
supported components	562	invariant culture	
instance count		editing the application export file	236
About dialog box	106	editing the device export file	664
Instance Editor		editing the XML application export file	248
accessing	182	I/O device actions	570
presentation	183	I/O device addresses	
instance link status	204	Build stage	788
instance path		I/O devices	
viewing instance location information	211	communication mapping import rules	284
instance properties		creating I/O devices	309
naming rules	31	I/O devices for calculated variable tags	310
instances		I/O devices for disk variable tags	310
accessing the Instance Editor	182	properties (supervision)	309
actions	179	I/O items	
changing the application hierarchy	213	creating and modifying	642
color indicators	195	deleting	644
configuration parameters	191	I/O Mapping Editor	
configuring parameters	188	description	547
copying and pasting	214	I/O Profile tab	
counting for licensing	180	attributes of the I/O Profile tab	709
Create Instance dialog box	351	I/O profiles	
creating	177	configuring I/O profiles	638
creating in-place	341	configuring I/O scanning	641
data status	186	creating and configuring	709
editing in-place	390	I/O scanner	
element selection changes of assigned facets	380	configuring for controllers	595
exceeding count of a license	180	disabling	593
exporting and importing to XML or CSV	227	I/O scanner lines	
hierarchical naming	189	memory allocation	778
import rules for control project-related objects	282	I/O scanning	
import rules for supervision project-related		configuring communication	641
objects	326	configuring I/O profiles	638
Inspect Instance window	211	connecting PRMs to logical networks	618
managing	210	creating and configuring Modbus TCP device	603
moving	216	device DDT of scanned devices	641
navigating from facets to the instance	395	disconnecting PRMs from logical networks	619
of templates	45	implementing communication	936
order when creating	177	managing communication with devices	937
resolving import conflicts	260	managing communication with PRMs	941
searching	166	managing communication with STB islands	940
searching application instances and folders	161	managing peer to peer communication	943
validating parameters	188	I/O server redundancy	
viewing facet assignments	392	implementing	314

I/O servers				links	
communication ports	306			changing the routing of links	613
properties	306			connecting entities by physical links	608
INTERBUS fieldbuses				creating links to view instance information in	
fieldbus editor description	553			RTNS	208
Interface Editor				deleting	613
pane description	1009			displaying link names	614
presentation	1008			editing the XML export file	251
Interfaces				exporting and importing to XML or CSV	227
attribute descriptions	679			modifying links between instances	206
IP address				properties	614
assigning to controllers	592			rules for generating graphical links	402
assigning to EtherNet/IP devices	604			Links Editor	
assigning to Modbus TCP devices	604			Asset Workspace Editor	200
assigning to PRMs	601			Links Editor	1127
assigning to STB islands	599			presentation	1128
assigning to workstations	598			LL984	
changing EtherNet/IP device properties	605			updating logical control projects	900
changing for explicit messaging with devices	948, 951			using LL984 language	24
changing Modbus TCP device properties	605			local slaves	
changing STB islands properties	600			configuring I/O scanning	641
deploying changes to redundant controllers	831			creating and configuring local slaves	709
online changes with redundant controllers	875			implementation workflow	645
refreshing engine status and connecting	803			locating	
IP addresses				locating objects in physical views	540
assigning by using the DHCP service	624			location	
attribute description	679			viewing instance location information	211
automatic assignment (Logical Network View)	627			locked code	
changing (Logical Network View)	627			code lock feature	452
displaying entity IP addresses	538			locked facets	
IP forwarding				skipping facets during generation	400
configuring IP forwarding for BMENOC0321	682			locking	
correspondence with Topology Explorer	522			Lock command (engineering client)	104
IPSimPortAssign.xml				locking mechanisms	
port simulation file location	477			multiuser functionality	82
italic formatting				locks	
editing instance properties	188			entities locked for editing	530
items				log files	
creating and modifying I/O items	642			activity log files	92
deleting	644			logging in	
				to server and clients	72
K				logging out	
keyboard shortcuts	123			of server and clients	72
				logical control projects	
L				updating with online changes	897
labels				Logical Network	
displaying entity names	538			attribute description	679
displaying link labels	538			logical network actions	532
displaying link names	614			logical network connections	
layout of function blocks				logical network connection import rules	654
after first generation	403			Logical Network View	
when generating after changes	433			description	541
when regenerating	435			logical networks	
LD				changing entity interface assignments	627
using LD language	24			changing logical network assignments	627
libraries				configuring	576
viewing in the Global Templates Explorer	992			connecting controllers to logical networks	617
licensing				connecting PRMs	618
counting instances	180			connecting PRMs to logical networks	618
exceeding license instance count	180			creating	576
link validity status	204			deleting	533
Linked Production Assets section				disconnecting PRMs	619
creating links to view instances	208			displaying logical network information	538
linking instances	200, 206			highlighting logical networks	625
using the Asset Workspace Editor	200			properties	577
				Logical Networks	
				System Project pane	532
				Logical Number	
				attribute description	687

loops		
generating circular links	402	
M		
M340		
connecting controllers to logical networks	617	
creating a control project	269	
creating controllers	591	
creating PRM PROFIBUS DPs	601	
M340 controllers		
configuration limits	591	
M580		
creating a control project	269	
creating controllers	591	
M580 controllers		
configuration limits	591	
Identification mismatch icon	1111	
M580 safety systems		
implementing	908	
MAC address		
MAC Address attribute	679	
maintenance mode	932	
Manage DTMs command	636	
analyzing DTM usage	633	
Manage Network Variables		
saving changes in Manage Network Variables dialog box	473	
managers		
platform tools	23	
managing		
control projects	274	
supervision projects	317	
managing network variables		
M580 safety controllers	912	
mapping		
changing service mappings (control)	723	
clients (supervision)	767	
communication (control)	725	
communication (supervision)	769	
deleting mapping when updating topological templates	623	
editing exported hardware mappings	758	
exporting hardware mappings	755	
hardware	747	
importing hardware mappings	761	
I/O devices (supervision)	769	
mapping import rules (control)	284	
mapping import rules (supervision)	329	
mappings for safety-related control projects	929	
network variable mapping rules	743	
network variables	738, 757	
selecting variable mapping positions	742	
servers (supervision)	767	
services (control)	723	
services (supervision)	767	
to engines (control)	723	
topological mapping indicator	529	
Variable Mapping Editor window (control)	738	
mapping mode		
configuring I/O profiles	638	
mapping stage		
completing	719	
concept	63	
control projects	720	
exporting and importing hardware mappings	754	
supervision projects	763	
mask		
subnet masks of logical networks	577	
memory allocation		
when creating I/O scanner lines	778	
menus		
common template editor menus	1022	
using menus	117	
migrating		
backing up systems for migration	145	
migrating databases		
code lock feature	452	
Modbus explicit messaging		
communicating with devices	953	
managing communication	952	
Modbus serial		
managing communication	952	
Modbus serial devices		
communication mapping	769	
connectivity with supervision	984	
Modbus TCP		
configuring I/O profiles	638	
configuring I/O scanning	641	
Modbus TCP devices		
changing properties	605	
communication mapping	769	
connectivity with supervision	984	
creating and configuring devices	603	
exporting	647	
import rules	653	
importing	651	
Modbus TCP explicit messaging		
communicating with devices	944	
managing communication	944	
Modbus TCP generic devices		
diagnostic status	1116	
Modbus TCP implicit messaging		
implementing communication	936	
implementing communication with devices	937	
implementing communication with PRMs	941	
implementing communication with STB islands ..	940	
implementing peer to peer communication	943	
Modbus TCP requests		
creating and modifying I/O items	642	
modules		
Configuration with Topology Explorer	519	
deleting modules	596	
Identification mismatch icon	1111	
operating states	1112	
monitoring		
Modbus device data from supervision	984	
monitors		
floating Participant windows	118	
using dual monitors	110	
moving		
application folders	216	
changing the position of folders and maintain hierarchy	219	
facets	391	
generated facets	444	
instances	216	
moving entities between physical views	569	
moving entities inside a physical view	568	
moving folders	172	
moving sections (control)	275	
multiple selection		
selecting multiple items	109	
multiuser		
working with several clients	82	

N

names	
displaying entity names	538
naming convention	
Global Templates naming convention	993
naming instances	
hierarchical naming	189
naming rules	
rules for identifiers and variable names	31
rules for Supervision data	31
navigating	
inside the Application Browser pane	168
inside the instance pane	185
navigating from facets to the instance	395
navigation services	
using after restoring systems	155
network addresses	
generating network addresses for supervision	
components	789
network interface cards	
creating	598
properties	598
network redundancy	
implementing (supervision)	315
network variables	
address allocation during build	778
creating	473
description	463
management function rules	471
managing network variables	471
mapping	738, 757
mapping rules	743
modifying	474
properties	471
selecting variable mapping positions	742
Variable Mapping Editor window	738
networks	
conflicting Ethernet connections	616
connecting controllers to logical networks	617
connecting PRMs to logical networks	618
disconnecting controllers from logical networks ..	618
disconnecting PRMs from logical networks	619
import rules	653
logical network connection import rules	654
physical connection import rules	654
NIC	
assigning IP addresses	598
creating	598
properties	598
NIC actions	570
NICs	
representation of devices	578
NOC	
configuring memory for I/O scanning	595
NOC communication modules	
configuring HART modules	606
non-FBD sections	
notification when regenerating	435
Not Built	
actions changing the build status (control)	781
notification panel	
abort task icon	85
description	85
displaying	104
performance considerations	86
NTP	
attribute descriptions	698

O

object models	
application templates	42
interface models	44
topological templates	43
object templates	
classification	38
description	38
elements	38
references	38
types	40
objects	
object model overview	36
OFS	
configuration file description	788
OFS configuration file	
creating	787
deploying	841
saving	764
OFS services	
creating in workstation	597
OFSOPC	
tag container properties	354
OFSOPC driver	
communication board properties	312
OFSOPC/OPCUA configuration files	
downloading	791
online changes	
changing the deployed control project	874
main redundant configuration scenarios	889
overview	870
refining safety-related control projects online ..	932
selecting IP addresses	875
updating logical control projects	897
updating projects after deploying changes	
online	875
online help	
HTML help for user-created templates	98
using the help for several versions of templates ..	96
Online modification in RUN or STOP	585
online refinement	
importing sections	885
online refinement indicator	529
OPC Factory Server	
OFS configuration file description	788
OPC UA	
attribute descriptions	707
OPC UA Server Expert	
configuration file description	788
OPCUA	
configuration file description	788
tag container properties	354
variable tag addresses	354
OPCUA driver	
citect.ini parameters	312
communication board properties	312
OPCUA embedded	
mapping communication	769
Open Rack Viewer command	570
Open Web Browser command	570
Open With command	570
operating states	
M580 controller operating states	1112
operation clients	
using navigations services after restoring	
systems	155
operation servers	
creating in the topology	597
operator stations	

creating in the topology.....	597
order	
changing facet order inside section.....	394
changing FBD section order.....	275
order of sections	
notification when regenerating non-FBD sections.....	435
Out Of Date	
actions changing the build status (control).....	781
Out Of Date facets	
layout of blocks after generation.....	433
owner	
about network variables.....	463
managing network variables.....	471
owner and consumer projects.....	463, 471
owner project	
description.....	463

P

page templates	
selecting and managing page templates.....	359
supervision page template import rules.....	327
supervision project settings.....	302
pages	
actions.....	358
assigning genie facets.....	509
creating supervision pages.....	358
editing.....	509
properties.....	358
supervision page import rules.....	327
panes	
common template editor panes.....	1028
working with panes.....	113
panning	
moving the workspace around.....	108
parameter pane	
in template editors.....	1028
parameters	
accessing instance properties.....	182
conditional font formats.....	188
configuring instance parameters.....	188
correspondence with Topology Explorer.....	522
instance configuration.....	191
instance parameter filters.....	185
naming rules.....	31
regional options for instance parameter.....	188
validating instance parameters.....	188
Participant projects	
creating.....	396
deploying changes to workstations (supervision).....	851
deploying controller data to engines (control).....	824
deploying from workstation (supervision).....	846
deploying to engines (control).....	814
deploying to redundant controllers.....	812
deploying to workstations from executable (supervision).....	843
executing with redundant controllers.....	859
managing data backup files (control).....	296
redeploying to engines (control).....	819
redeploying to engines from executable (supervision).....	844, 850
system root folder actions.....	267
Participant windows	
troubleshooting floating window appearance.....	119
Participants	
floating Participant windows.....	118
opening the Participant help.....	95

Participant window description.....	118
password protection	
controller properties.....	592
system properties.....	139
passwords	
actions that require providing the password.....	806
backing up systems.....	145
building control projects.....	775
changing passwords of project loaded at simulator startup.....	810
changing the application password of control projects.....	809
changing the password for Global Templates.....	1004
changing the password of systems.....	142
changing the Webuser password.....	588
clearing the application password of control projects.....	810
clearing the password for Global Templates.....	1004
clearing the password of systems.....	142
controller security service attributes.....	692
exporting password related information for topological entities.....	648
Global Template settings.....	1002
importing password related information for topological entities.....	654
managing Control facet template passwords.....	1002
managing passwords of control projects.....	804
managing system access passwords.....	141
password information and settings in system backups.....	147
password information and settings when restoring systems.....	153
project loaded at simulator startup.....	808
recovering a forgotten Webuser password.....	589
refining control projects.....	447
resetting a forgotten controller password.....	810
resetting a forgotten functional safety password.....	926
resetting a forgotten Global Templates password.....	1005
resetting a forgotten system password.....	143
setting the application password of control projects.....	808
verifying password to perform operations.....	809
workstation security service attributes.....	693
paste command	
aborting.....	214
pasting	
Global Templates and folders.....	1038
pasting instances in Control projects.....	348
peer to peer	
communication mapping import rules.....	284
mapping network variables.....	738, 757
peer to peer communication export settings.....	279
peer to peer communication variable import rules.....	287
peer to peer communication	
about network variables.....	463
address allocation during build.....	778
client communication channel properties.....	733
communication mapping (control).....	725
configuring local slave scanning.....	641
configuring the I/O scanner of controllers.....	595
Control Participant functionality.....	475
controller memory size configuration.....	592
creating communication mapping (control).....	729
creating network variables.....	473
examples.....	492
exporting control projects.....	276

exporting Control projects	491	configuring	601
implementing	943	creating PRM PROFIBUS DPs	601
importing Control projects	491	creating PRM PROFIBUS Masters	601
managing network variables	471	deleting	602
modifying network variables	474	import rules	653
network variable properties	471	managing Modbus TCP communication (I/O scanning).....	941
owner and consumer projects	463	properties	601
server communication channel properties	736	representation	602
templates	463, 471	PRM actions	570
variable properties	467	PRMs	
peer to peer communication channels		configuration limits	601
actions	733	connecting to logical networks.....	618
adjusting properties.....	730	disconnecting from logical networks	619
creating	730	product ranges	
moving	732	list of product ranges	517
performance		product types	
notification panel.....	86	list of product types	517
OFSOPC driver configuration.....	312	PROFIBUS	
physical links		acting on PROFIBUS devices	555
connecting entities by physical links	608	connecting to logical networks.....	618
deleting	613	creating PRM PROFIBUS DPs.....	601
properties	614	creating PRM PROFIBUS Masters	601
Physical Ports		disconnecting from logical networks	619
attribute descriptions	682	fieldbus editor description	553
physical view action	534	managing Modbus TCP communication (I/O scanning).....	941
physical view folder actions	534	using device DTMs.....	629
physical views		PROFIBUS remote master and DPs	
creating	534, 536	exporting	647
deleting	534	importing	651
description.....	536	PROFIBUS secondary devices	
exporting physical views	648	integrating in systems.....	988
highlighting logical networks	625	programming languages	
importing physical views	651	supported languages.....	24
moving entities between physical views	569	progress indicators	
moving entities inside a physical view	568	instance color icons.....	195
opening	534	project color icons	333
representation of objects	578	Project Browser	
searching physical views	540	system root folder actions	267
P&ID		project definition stage	
instantiation stage	157	completing.....	262
pin button		concept	53
working with panes.....	113	control projects	268
platform components		Project Browser	263
platform tools	23	Project Explorer	263
platforms		supervision projects	298
supported hardware platforms.....	24	Project Explorer	
PMEPXM0100 modules		description.....	264
integration workflow	988	opening.....	104
port simulation file location	477	project files	
port types		downloading (supervision)	791
analysis rules.....	1101	project settings	
color coding of RIO/DIO ports	539	analyzing M580 safety projects and configurations	924
ports		defining (supervision)	509
attribute description.....	682	setting during build (control).....	777
creating additional communication ports (supervision).....	313	projects	
creating communication ports (supervision)	312	color indicators	333
I/O server communication ports.....	306	generating	427
representation of entities	582	generating after changes	429
supervision server communication ports	307	generating system documentation reports.....	1084
workstation properties	708	types of Participant projects	54
positions		properties	
selecting variable mapping positions	742	accessing instance properties	182
printing		adjusting peer to peer communication channel properties	730
template layouts.....	1024	application folders	172
PRM		changing EtherNet/IP device properties	605
assigning IP addresses	601		
changing the configuration.....	601		
changing the PRM configuration.....	602		

changing Modbus TCP device properties	605	to engines from executable (supervision)	844, 850
changing STB islands properties	600	to redundant controllers	819
configuring instance parameters	188	redundancy	
correspondence with Topology Explorer	522	implementing (supervision)	314
I/O devices (supervision)	309	redundant alarm servers	
I/O server properties	306	implementing alarm server redundancy	315
logical networks	577	redundant controllers	
physical link properties	614	backing up controller data	880
properties of sections	349	communication mapping and local slave	
supervision client properties	308	management	641
supervision server properties	307	connecting to logical networks	616
systems	139	deploying changes to	813
Properties Inspector pane		deploying controller data	824
description	673	I/O profile tab	709
Properties pane		physical connection import rules	654
attributes of the Configuration tab	676	refining online	860
attributes of the Description tab	675	representation of devices	578
attributes of the Diagnostic tab	717	synchronizing	866
attributes of the I/O Profile tab	709	redundant I/O servers	
attributes of the Security tab	690	implementing I/O server redundancy	314
attributes of the Services tab	694	redundant networks	
description	673	highlighting	539
Protocol		implementing network redundancy	
tag container attributes	354	(supervision)	315
Purge command		redundant report servers	
aborting template purge	1050	implementing report server redundancy	315
purging		redundant trend servers	
Global Templates library	1049	implementing trend server redundancy	315
		reference	
		system engineering life cycle task workflow	126
		Reference	
		attribute description	686
		references	
		definition	38
		missing references in Recovery folder	997
		selecting references when creating entities	565
		refine online	
		main redundant configuration scenarios	889
		overview	870
		with redundant controllers	859
		refinement	
		consistency management (control)	460
		refining locked code	452
		refinement stage	
		completing	445
		concept	59
		control projects	446
		supervision project	508
		refining	
		advanced supervision project settings	509
		control projects	446
		editing supervision pages	509
		managing device DDT variables	455
		refining safety-related control projects	921
		refining safety-related control projects online	932
		supervision projects	508
		validity indicators	449
		refining online	
		refining online with redundant controllers	860
		refining safety-related control projects online	932
		ReGenerate command	
		aborting regeneration of projects and sections	435
		regenerating	
		configurable settings (control)	405
		control project sections	436
		control projects	434
		entire control projects	435
		FBD section size	435
		function block layout after changes	435

functionality description	434	creating links to view instance information	208
importing sections after regeneration	429	rules	
removing types	434	analysis rules	1094
regenerating non-FBD sections		truncation rules for equipment (supervision)	437
notification when regenerating	435	run	
regional options		starting controllers and simulators (control)	861
instance parameters	188	runtime navigation services	
release notes		creating links to view instance information	208
viewing platform and library release notes	106	using after restoring systems	155
relinking			
facets	444	S	
removing		S88	
removing entities from a physical view	539	assigning icons to folders	176
removing generated types	404	modeling the equipment hierarchy	159
removing types when regenerating	434	S908	
renaming		building changes	783
Global Templates	1035	using CCOTF with S908 architectures	585
Replace dialog box		safe area password	
description	1069	managing the functional safety password	925
Replace References dialog box	1052	safe signature	
Replace Template		building safety control projects	930
command description	220	deploying to safety controllers	931
replacing		safety controllers	
replacing templates of elements	1060	configuring M580 safety controllers	923
template of instances	220	implementing M580 safety systems	908
templates with new type version	224	safety passwords	
Replacing references in parent templates	1052	resetting a forgotten functional safety	
replacing templates		password	926
validating instance identifier uniqueness	225	Save Control Project As... command	570
replacing the controller module	594	saving	
report server redundancy		backing up controller data	880
implementing	315	changes in Manage Network Variables dialog	
reports		box	473
application data	1083	Global Templates	1035
generating system documentation reports	1080	saving templates	
Global Templates data	1088	tracking changes in the changes log	1035
project data	1084	sbk file	
topological data	1086	exporting templates	1039
requests		searching	
Modbus TCP request attribute descriptions	713	application instances and folders	161
reservation		instances	166
redundant configuration reservation scenarios ...	892	searching physical views	540
R/W Rights of Referenced Variable		template search	163
network variables	471	section layout	
Resolve Uniqueness Conflicts		configurable settings (control)	405
dialog box description	260	section size	
dialog box description for device import	671	configurable settings (control)	405
restarting		sections	
restarting clients that closed unexpectedly	78	actions	348
restore		changing execution order	275
impact on system names in database	155	creating and managing	341
password information and settings when restoring		creating FBD and non-FBD sections	273
systems	153	creating FBD sections	348
restore command		creating sections in safety-related control	
aborting	153	projects	919
restoring		deleting FBD sections during update Control	
deployed Participant project from executable		project	904
(supervision)	844	import rules for program sections	282
deployed Participant project from workstation		importing during online refinement	885
(supervision)	849	inserting FBD sections	348
restoring systems	152	moving sections (control)	275
RIO ports		properties	349
color coding	539	tree view description in Containers pane	384
RJ45 connector representation	582	tree view when refining Control projects	448
rollback		tree view when refining Control projects online ...	875
rolling back generation	402	security	
RSTP		controller password attributes	692
attribute descriptions	697	security service attributes	691
RTNS			

simulator password attributes.....	693	changing the default project loaded at startup.....	800
Security Level		deploying built control projects	814
attribute description.....	690	deploying changes (control).....	830
security services		deploying controller data (control).....	824
system properties	139	managing data backup files (control)	296
Security tab		mapping hardware	747
attributes of the Security tab.....	690	obtaining the IP address	801
selecting		redeploying built control projects	819
selecting multiple items	109	setting passwords of project loaded at simulator	
selecting highlights	536	startup	808
selecting logical networks	536	simulator password attributes.....	693
description of the feature	625	starting	800
server		starting without loading a password-protected	
connection to clients.....	76	project	801
starting again.....	75	simulator ports	800
stopping	80	simulator software	
Server Memory		deploying to	799
attribute descriptions	685	installing.....	799
Server Memory Length		starting and stopping	861
attribute descriptions	685	skipping	
Server Memory Start		skipping facets during generation	400
attribute descriptions	685	SNMP	
servers		attribute descriptions	701
creating servers (supervision)	306	software Participants	
I/O server properties.....	306	floating Participant windows.....	118
supervision server communication ports	307	window description.....	118
supervision server properties	307	sorting	
service mapping		sorting data	120
changing (control)	723	ST	
services		using ST language	24
generating network addresses	789	STA	
security service attributes	691	changing the default project loaded at startup.....	800
Services tab		starting the simulator without password	
attributes of the Services tab.....	694	protection	801
session overcrowded		stages	
troubleshooting	440	assignment stage concept	56
sessions		assignment stage implementation	340
restarting clients that closed unexpectedly	78	build stage concept	65
settings		build stage implementation	773
configuring generation settings (control)	405	concept	46
Settings		configuration stage concept	61
control project settings.....	272	configuration stage implementation	512
Global Template settings	1002	deployment stage concept	67
modifying the template folder layout	996	deployment stage implementation	793
supervision project settings.....	302	execution stage concept	68
Settings pane		execution stage implementation	854
changing application folder icons.....	176	generation stage concept	58
Settings window		generation stage implementation.....	396
Systems Explorer.....	138	Global Templates definition stage concept	48
SFC		instantiation stage concept	49
using SFC language.....	24	instantiation stage implementation	157
shortcuts		mapping stage concept	63
keyboard shortcuts.....	123	mapping stage implementation	719
Show in Active Views command	570	project definition stage concept	53
signature mismatch		project definition stage implementation	262
actions that require providing the password.....	806	refinement stage concept	59
controller data backup	883	refinement stage implementation.....	445
controller data deployment.....	827	tasks	130
deployment stage	830	starting	
execution stage	857	Controller simulator	799
simplified system engineering life cycle		controllers and simulators (control).....	861
graphical representation	69	simulator	800
simplified view		system server	75
Asset Workspace Editor	203	status	
template editors	1032	editing instance properties	188
simulator		instance data validity status	186
backing up controller data.....	880	refreshing engine status	803
changing passwords of projects loaded at		status of supervision executables	
startup.....	810		

status change when importing new content version	1046	mapping stage	763
STB island actions	570	page import rules	327
STB islands		page template import rules	327
adding entities to systems	565	project definition	298
assigning IP addresses	599	Project Explorer and Browser	263
changing properties	600	project import window	329
changing the configuration	599	properties	302
configuring	599	refining	508
creating STB islands	599	related project data import rules	326
deleting	600	restoring after deploying from executable	844
exporting	647	restoring after deploying from workstation	849
importing	651	service mapping import rules	329
managing Modbus TCP communication (I/O scanning)	940	task workflow	126
mapping (control)	723	tasks	130
mapping hardware	747	supervision servers	
properties	599	properties	307
representation	600	Supervision service	
stop		creating in workstation	597
changes requiring to stop the controller	840	support	
stopping		support contact details	106
controllers and simulators (control)	861	symbol table files	
system server	80	generating symbol table files	798
structure of the application		synchronization	
instantiation stage	159	timeout when refining online	860
STU files		Synchronize Changes	
changing (control)	781	command description	590
creating (control)	775	synchronizing	
subnet masks		description	859
subnet masks of logical networks	577	redundant controllers	866
supervision		timeout	866
connectivity with Modbus devices	984	when deploying changes	813
supervision with M580 safety systems	934	syslog server	
supervision architecture		audit trail for user activities	88
description of tasks	299	language settings	88
supervision clients		Syslog Server	
properties	308	attribute description	700
supervision data		System Access Password	
naming rules	31	managing system access passwords	141
supervision pages		system access passwords	
actions	358	changing the password of systems	142
creating pages	358	clearing the password of systems	142
properties	358	resetting a forgotten system password	143
Supervision Project Browser		system backup	
description	266	password information and settings	147
supervision projects		System Diagnostic	
actions	300	attribute descriptions	717
building	787	system engineering life cycle	
changing built Participant projects	792	concept	46
communication mapping import rules	329	platform components	23
creating	300	simplified workflow	69
creating client	308	stage diagram	46
creating components	304	task workflow	126
creating I/O devices	309	tasks	130
creating servers	306	system folders	
deploying	841	actions	135
equipment import rules	327	creating	135
executable import rules	329	system monitoring	
executing	907	displaying device communication status	1107
export settings	321	system names	
exporting	318	impact of system restoration	155
facet import rules	327	System Project pane	
generating after changes	430	creating and configuring logical networks	576
hardware mapping import rules	329	Logical Networks node description	532
implementing redundancy	314	Physical Views node description	534
import settings	324	Topology node description	527
importing	323	System Project pane description	525
managing	317	system restore	
		password information and settings	153
		system server	

logging in and out.....	72	content repository	1124
systems		terminology	20
actions	137	for template components	39
backing up	145	TeSys	
creating	137	adding TeSys devices to systems	566
description	20	diagnostic status	1115
engineering	124	timeout	
managing system access passwords	141	synchronizing standby controllers	866
organizing	135	when synchronizing online changes	860
properties	139	TLS exchange notification	
restoring	152	troubleshooting	789
scheduling system backups	150	toolbars	
Systems Explorer		common template editor toolbars	1022
opening	104	description	104
presentation	135	engineering client UI description	103
		Topology Explorer toolbar description	515
		Toolbox	
		Global Templates Explorer	994
		topological entities	
		import rules	653
		logical network connection import rules	654
		topological entity identifiers	
		naming rules	31
		topological entity nodes	
		description	529
		topological folders	
		import rules	653
		topological templates	
		introduction	43
		subtypes	43
		topology	
		adding entities to a physical view	568
		adding entities to systems	565
		configuration stage	512
		creating	562
		exporting	647
		generating system documentation reports	1086
		import rules for control project-related objects	282
		import rules for supervision project-related	
		objects	326
		importing	651
		managing	622
		managing consistency	1090
		selecting topological templates	584
		supported components	562
		updating templates	622
		Topology Explorer	
		opening	104, 513
		troubleshooting	513
		Topology Explorer description	514
		Topology Explorer toolbar	515
		topology folders	
		actions	563
		creating	563
		properties	564
		Topology node	
		context menu commands	531
		System Project pane	527
		topology objects	
		exporting and importing to CSV	660
		tracking changes	
		template changes log	1035
		transferring	
		deploying changes to workstations	
		(supervision)	851
		deploying from workstation (supervision)	846
		deploying to workstations from executable	
		(supervision)	843

T

tabs bar	
description	107
tag containers	
actions	354
creating tag containers	353
properties	354
TCP listening ports	
workstation properties	708
template passwords	
changing the password for Global Templates ...	1004
clearing the password for Global Templates	1004
templates	
copying and pasting Global Templates	1038
creating Global Template folders	998
description	38
duplicating Global Templates	1065
exporting	1039
Global Template editors	1007
Global Templates actions	1001
Global Templates description	1000
Global Templates Explorer	992
HTML help for user-created templates	98
import rules for control project-related objects	282
import rules for supervision project-related	
objects	326
importing	1042
instances of templates	45
instantiating	158
managing	1035
modification strategy	1052
naming convention	993
purging the Global Templates library	1049
renaming Global Templates	1035
replacing templates of elements	1060
saving changes in Global Templates	1035
selecting	163
selecting topological templates	584
templates Import Editor	1042
terminology	39
updating Global Templates	1054
updating templates at folder level	1058
using the HTML help for several versions of	
templates	96
Templates Browser	
browsing templates	163
templates of instances	
replacing	220
updating	220
templates of the topology	
updating	622
temporary files	

redeploying to engines from executable (supervision)	844, 850	editing templates	1036
synchronizing redundant controllers	866	Updating references in parent templates	1052
tree view		user authentication	
viewing the application structure in control projects	345	OPC UA configuration files	788
tree view in control projects		User Contents folder	
description	384	location in the content repository	1120
when refining	448	user files	
when refining online	875	redeploying from executable after removing user files (supervision)	844, 850
trend server redundancy		user interface	
implementing	315	engineering client	102
troubleshooting		engineering client UI description	103
floating Participant window appearance	119	engineering client work areas	107
generation diagnostic messages	440	User Preferences window	524
troubleshooting Topology Explorer	513	user-created templates	
truncation		HTML help for user-created templates	98
rules for equipment (supervision)	437	users	
type versions		activity log files	92
changing type versions during generation (control)	409	logging into server and clients	72
replacing templates with new type version	224	working with several clients	82
updating templates with new type version	224		
types		V	
removing generated types	404	Validity Diagnosis window	204
removing when regenerating	434	validity indicators	
U		analyzing control projects during generation	401
unassigned facets		description	777
reassigning	444	instance color icons	195
unassigning		project color icons	333
facets	391	refining control Participant projects	449
generated facets	443	Variable Mapping Editor	
Under Construction		description	738
attribute description	676	usage example	742
unlinked facets		variable mappings	
relinking	444	peer to peer communication variable import rules	287
unlinking		variable name conflict	
facets	443	troubleshooting	440
unmap		variable tag addresses	
unmapping hardware mappings	752	OPCUA embedded protocol	788
unmapping		variables	
unmapping communication channels	731	about network variables	463
Update Project command		address allocation during build	778
aborting	905	managing network variables	471
Update References dialog box	1052	naming rules	31
Update Template		properties	467
command description	220	when updating Control projects	901
Update Template command	570	version conflict	
updating		canceling generation for version conflict (control)	402
control projects after deploying changes online	875	troubleshooting	440
controller configurations	897	versions	
Global Templates	1054	changing type versions during generation (control)	409
Global Templates at folder level	1058		
linked application templates	224	W	
logical control projects	897	Web Diagnostic password	
template modification strategy	1052	building control projects	775
template of instances	220	refining control projects	447
templates of elements	1054	regenerating control projects	434
templates with new type version	224	Web diagnostic passwords	
topological templates	622	changing the Webuser password	588
updating control projects		Web Diagnostic passwords	
checking consistency	903	recovering a forgotten password	589
deleting FBD sections during update Control project	904	Web pages	
updating safety-related control projects	932	Rack Viewer description	559
with redundant controllers	859	Webuser passwords	
updating parent templates			

changing the Webuser password	588
recovering a forgotten password	589
windows	
floating Participant windows	118
Participant window description	118
viewing on separate monitors	110
work areas	
creating additional work areas	110
creating work areas in explorers	111
description	107
organizing	110
workflow	
task workflow to implement communication	126
workspace	
changing the zoom level	109
zooming in and out	109
workspaces	
moving the workspace around	108
selecting items	109
working with panes	113
workstation actions	570
workstations	
adding entities to systems	565
attribute description	708
backing up controller data	880
creating	597
exporting	647
import rules	653
importing	651
mapping to emulated simulators (control)	723
selecting templates	584
WRITE_REMOTE	
peer to peer communication	475
WRITE_REMOTE function	
configuration	481

X

X509 user token	
OPC UA configuration files	788
XML	
application export file description	244
editing the application export file	246
editing to manage application links	251
exporting and importing application objects to XML	
or CSV	227
importing XML files into the application	254
XVM files	
generating symbol table files	798

Z

zooming	
magnifying and reducing the workspace view	109
using the radar view	1031
zooming out	
simplified view in Asset Workspace Editor	203
simplified view in template editors	1032

Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison
France

+ 33 (0) 1 41 29 70 00

www.se.com

As standards, specifications, and design change from time to time,
please ask for confirmation of the information given in this publication.

© 2023 Schneider Electric. All rights reserved.

EIO0000001114.18