

# 智能建筑：楼宇管理系统（BMS）“开放性” 评估框架

第 501 号白皮书

版本 1

作者：Jeffrey Bowman  
Chris Megede  
Wendy Torell

## 摘要

随着“智能”建筑这一理念在业主中的普及，楼宇管理系统（BMS）的“开放性”需求也与日俱增。然而，“开放性”一词本身的含义就模棱两可，容易产生混淆，同时可能会导致业主选择那些无法满足当前和未来业务需求的系统。我们认为，在讨论开放性 BMS 时，需明确开放性的三个层级：(1)数据采集/共享，(2)系统集成，以及(3)楼宇统筹，从而避免概念混淆，并就此展开清晰明确的讨论。本白皮书会确定 BMS “开放性”评估框架的定义，详细说明相关术语，并针对关键开放性标准展开讨论，包括它们如何影响 BMS 的复杂性和性能。此外，本白皮书还列举了各应用场景。

## 简介

在过去，楼宇管理系统（BMS）的功能主要是监控楼宇暖通空调（HVAC）系统的运行。然而，随着关于提升楼宇效率、可持续性、生产力和用户体验的社会压力逐步增加，BMS 的作用也应随之改变。为此，业主需改变传统楼宇系统的孤立状态，实现互联互通，集成物联网（IoT）设备和第三方应用，最终实现整个楼宇运营的自动化和优化。

在施耐德电气第 500 号白皮书《[下一代楼宇管理系统（BMS）需要具备的三个基本要素](#)》中，我们讨论了传统 BMS 无法进一步提升其功能的原因，并说明了下一代 BMS 作为主动监测、控制和自动化的智能楼宇系统平台所必备的特性。此外，我们还讨论了诸多业主希望实现智能建筑的目标，以及信息技术（IT）、物联网和“智能建筑”等技术的进步如何从以下方面赋能楼宇：

- 通过利用电网服务和人工智能（AI），最大限度地减少能源消耗
- 统筹 HVAC、安保/安全、照明和入住等系统，提升用户工作效率及满意度
- 尽可能以最可持续和可靠的方式开展自动化运营

供应商认为，“开放性”是 BMS 的特性；业主则认为，“开放性”是其对 BMS 的要求。但是，**由于业内尚未对“开放性”一词拟定标准定义**，人们对其理解通常模糊不清，有混淆之处。“开放性系统”的一般定义是“一个结合了互操作性、可移植性和开放性软件标准的系统”。<sup>1</sup>但是，这个定义过于宽泛，还有很多有待阐释的内容。请注意，本白皮书并未涉及“开源”（可自由获取并可修改的软件）。

当论及开放性 BMS 时，人们经常会谈到技术能力（即协议兼容性），有时也会提到系统调试和维护的便利性，此外，也会说到与其他楼宇系统互联互通的能力。由此可见，人们对开放性楼宇管理系统的定义各有不同，也难以就此展开明智的谈话。这种情况下，人们很有可能选择一个无法满足当前和未来业务需求的系统。

为此，本白皮书拟定了开放性 BMS 的讨论框架（如图 1 所示），以便楼宇决策者评估系统功能、在了解相关背景情况下进行深入讨论、知晓实现开放性的复杂度及其影响，并根据建筑的预期效果，最终做出更明智的系统选择。

图 1  
“开放性”BMS 的  
评估框架



<sup>1</sup> [https://en.wikipedia.org/wiki/Open\\_system\\_\(computing\)](https://en.wikipedia.org/wiki/Open_system_(computing))

开放性是一个复杂的话题。每栋楼宇都是独特的，也有不同的目的。在本框架中，我们为决策者划分了三个不同的开放性层级，每层的开放程度有所不同，也代表了有着不同预期和挑战。同时，每层都建立在上一层级之上。换言之，第 1 层的功能是获得第 2 层功能的先决条件，第 2 层的功能是获取第 3 层（即，实现“智能建筑”目标）功能的先决条件。为了确定系统在各层级的开放程度，我们制定了三项标准：

- **互操作性**——系统的某一部分与其他部分的协作能力，或一个系统与其他系统的协作能力。系统的开放程度越高，可共享的数据就越多，互操作性也会越复杂。
- **工程复杂性**——实现互操作性的难度。关键考量因素在于所需的定制化服务和编程量。
- **执行人**——该项工作是否需要娴熟的专家？是否仅限于持证人员？对专业人员和专业技能的需求是执行该项工作的重要一环。

然而，我们通常需要在这些标准中做出权衡。例如，您的系统可能在互操作性方面实现高度开放，但却要以牺牲工程简易性为代价。

下文中，我们将详细说明相关术语及其定义，并分别说明三个层级，其中包括用例介绍以及开放环境下，上述标准影响系统性能的方式。本白皮书提供的用例均基于一家医院案例，以便说明这些层层构建的层级将如何实现最终目标——智能且统筹的楼宇。

## 术语

在针对 BMS 进行讨论时，用到了很多术语，但是这些术语通常都被曲解和误用。因此，在说明上述三个层级前，需详细阐释描述 BMS 时所需的基本术语，以及该等术语对开放性的意义。

**BMS 系统**是一个综合的软硬件相结合的系统，包括下列主要部件。

- **I/O (传感器、继电器和执行器)**<sup>2</sup>——传感器可针对您想了解的各项环境信息（包括温度、入住情况、气压）提供输入信息；继电器是电动操作的开关，可用于设备启动和关闭（即风扇开关、电灯开关等）。执行器是指可移动对环境有影响的机械装置的设备（即打开阻尼器、调整阀门等）。
- **现场控制器**——现场控制器是指可通过物理方式连接至传感器和执行器，并且能运行本机控制逻辑的可编程设备。此外，这些设备还能和楼宇控制器或其他现场控制器互相收发楼宇层面的信息，以影响本地控制逻辑，进而促成相应的操作（即关闭阻尼器）。
- **楼宇控制器**——有时亦指楼宇服务器，是指对多个现场控制器和/或其他楼宇控制器的活动进行集成和协调的设备。楼宇控制器会通过其自身的控制逻辑进行协调，此外也可以为 BMS 的 UI 提供服务。

<sup>2</sup> 通常而言，人们最开始接受的是底层的 I/O 设备，然后是现场控制器和楼宇控制器。

- **云基础设施**——云计算技术可用于提供长期历史数据储存、分析和/或接触并与第三方应用互动。
- **客户应用 (UI)**——系指构建、维护和操作系统的用户界面。它可以是建立在 PC 之上的应用程序，可以是建立在 web 上的应用程序，也可以是移动应用程序。该应用可执行编程功能，并通过图形和仪表盘展示系统视图。

在讨论开放性这个话题时，通常会围绕通信协议展开。但是，相关术语同样会有所混淆。例如，协议可能是开放协议或封闭协议，也可能是标准协议或专有协议。但是，人们通常将开放和标准及封闭和专有这两组词换用。实际上，协议可能同时兼具标准和开放属性，或专有和开放属性。如下所示为这些术语的定义以及应用程序编程接口 (API) 与协议的不同之处。

- **协议**——通信协议是允许两个或多个实体传输信息的规则系统，而这些规则又定义了语法和语义等事项<sup>3</sup>。实际上，协议可允许两个设备相互通信。当提到协议时，有些是指运营技术 (OT) 协议，有些是指信息技术 (IT) 协议，还有一些则是指物联网协议。如需了解各协议的更多信息，请参见第 500 号白皮书——[《下一代楼宇管理系统 \(BMS\) 需要具备的三个基本因素》](#)。智能建筑通常会涵盖三项协议。
- **开放协议**——通过开放协议，不同供应商的设备都能互相操作，且无需专有接口或网关<sup>4</sup>。它们操着相同的语言，无需翻译。LonWorks 和 BACnet 便是开放协议很好的案例。
- **封闭协议**——系指专有且需要接口或网关才能与其他供应商的产品进行通信的协议。在某些情况下，供应商可选择创建封闭协议，以便在该供应商的两个设备间启用供应商特定的功能。
- **标准协议**——系指由公认第三方标准机构制定并控制的协议，且该等协议是业内认可的通信方式，对所有人开放。例如，BACnet 是 ASHRAE 推动的标准 (和开放) OT 协议，TCP/IP 是 IEEE 推动的标准 (和开放) IT 协议。
- **专有协议**——系指由单个组织或个人拥有的通信协议<sup>5</sup>；例如，N2 Open 是由江森自控拥有并维护的专有 (但开放的) 操作技术协议。
- **应用程序编程接口 (API)**——是一种计算接口，定义了多个软件中介之间的互动。通过归纳底层实现和仅公开软件开发人员需要的对象或操作，API 可简化编程<sup>6</sup>。

<sup>3</sup> [https://en.wikipedia.org/wiki/Communication\\_protocol](https://en.wikipedia.org/wiki/Communication_protocol)

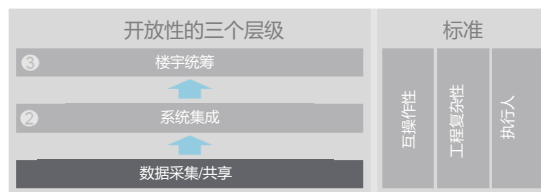
<sup>4</sup> 简言之，网关是多种协议间的翻译器，可以是硬件，也可以是软件。

<sup>5</sup> [https://en.wikipedia.org/wiki/Proprietary\\_protocol](https://en.wikipedia.org/wiki/Proprietary_protocol)

<sup>6</sup> <https://en.wikipedia.org/wiki/API>

## 第 1 层:数据 采集/共享

第 1 层——数据采集和共享，是“开放性”BMS 的关键基础。BMS 将控制以 HVAC 系统为代表的楼宇内各个系统。在传统的单一系统范围内，BMS 必须能够与**传感器/执行器**

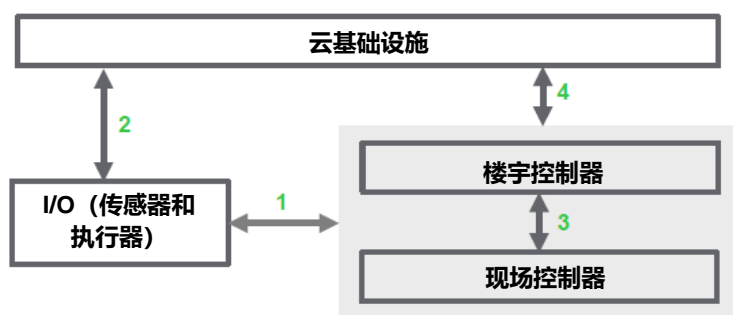


**器**和其他**楼宇控制器**互相收发数据（控制信号）。这种情况下，可能会产生以下四种数据流（如图 2 所示）：

1. 传感器和执行器与控制器共享/接收数据
2. 传感器与云基础设施共享/接收数据
3. 控制器与其他控制器共享/接收数据
4. 控制器与云基础设施共享/接收数据

图 2

第 1 层数据采集/共享的四种数据流



这些输入和输出数据流的通信方式都是利用电压/电流/电阻信号或 OT 协议或 IT 协议和 IoT 协议的有线通信。接下来，我们将详细说明传感器/执行器和控制器是如何利用这些通信方式的。

**传感器和执行器**——传统上，在 BMS 平台内，传感器和执行器的通用性和集成便利性最佳。同时，传感器和执行器是一种单向设备，只能向控制器共享其数据或接收控制器数据，主要通过简单的电气信号（即电压信号、电流信号、电阻信号）进行有线通信。它们可能是一个简单的压力、温度、入住率、二氧化碳传感器，或者是开启阀门的执行器，或者是调整风扇速度的驱动器。

智能传感器和执行器因具备以下功能，已经得到越来越普遍的应用：(1)将多个传感器集成到一个盒子上，从而减少连接点/电线数量以及相关人力；(2)无线通信，降低工程成本和安装；(3)云连接，降低本地部署基础设施的资本费用。然而，随着智能设备的涌现，人们对集成的需求也与日俱增。人们希望设备能支持双向通信，并同时感应和控制多个数值。鉴于设备的功能及其支持的通信协议，集成或多或少有些复杂。自此，“开放性”概念闪亮登场。

**控制器**——传感器和执行器可在没有协议的情况下实现数据流通信（即简单的有线信号）。与此不同，控制器与控制器之间的通信通常都需要协议。过去，人们使用的是封闭专有的系统，供应商之间互动较少，甚至没有互动。最近，得益于 BACnet 和 LonWorks 等协议，系统“开放性”得以实现。然而，控制器使用开放协议并不意味着控制器可实现互操作性，还需考虑复杂性以及开放程度。接下来，我们将就此展开讨论。

## 用例

为了协助说明传感器和控制器集成的复杂性，我们考虑了两种用例，详情如下。这些用例基于 BMS 系统图（如图 3）。

图 3

### 数据采集/共享应用

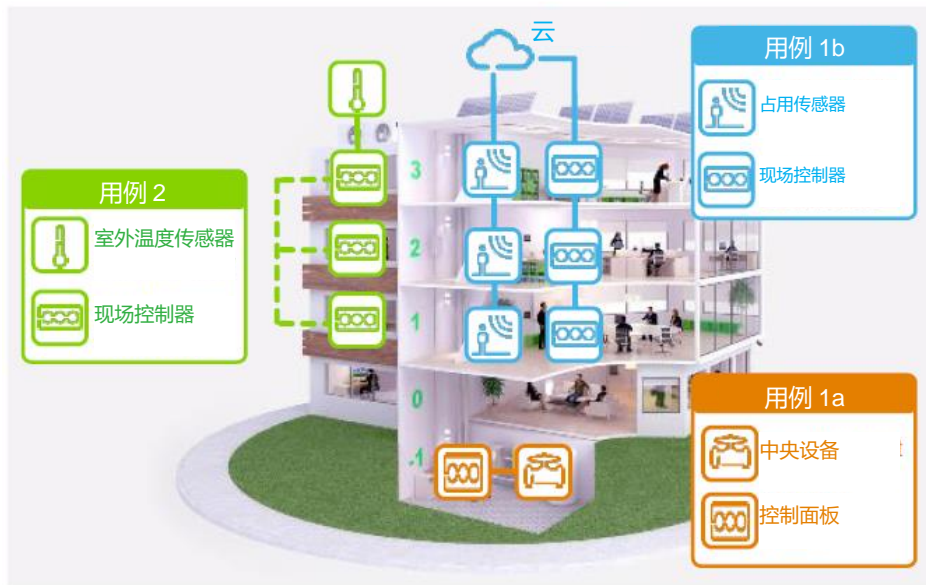
情景：

1a. 本地部署

传感器

1b. 云传感器

2. 控制器与控制器



### 用例 1：与通信传感器集成

传感器/执行器可能需要直接将数据发送给本地设备(1a)；但有时也需要通过云计算实现数据通信(1b)。

- **本地部署**——我想要控制中央制冷设备，而设备内含有多种传感器。我的设备有智能制冷设备，各种泵以及与泵有关的阀门和传感器/执行器。就此用例而言，我们需要控制泵的变速传动装置，包括控制和系统状态及能耗量监测。
- **云端部署**——连接至云计算的物联网传感器已变得越来越普遍。我希望，空间内的目标温度能针对占用情况变动做出反应。就此用例而言，我需要将物联网占用传感器（该传感器将接触 API）连接至 BMS，作为温度控制逻辑的输入信息。

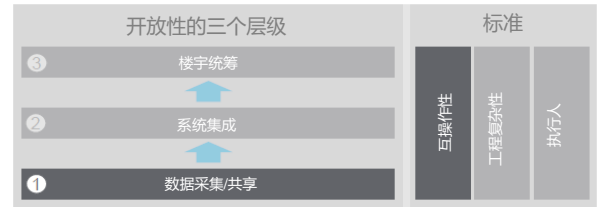
### 用例 2：控制器集成

为了控制成本，我只安装了一个室外温度传感器，我希望这个传感器能为所有需要室外温度数据的控制器提供输入信息。我也希望已与此传感器物理连接的控制器能直接将室外温度数据分享给网络中的其他控制器。

通过这些用例，我们可评估 BMS 在第 1 层的开放性。为了评估 BMS 的“开放性”，我们需要考量图 1 所示的三个标准——(1)互操作性；(2)工程复杂性；以及(3)执行人。请注意，这些标准适用于“开放性的三个层级”。

## 开放性标准 1——互操作性

根据用例 1 可知，传感器用于测量和记录数值，并向 BMS 系统进行报告。但是，如果系统无法读取传感器数据，那传感器还有什么用呢？



传感器的通信方式各有不同。因此，为了确保传感器的通信方式与目标 BMS 兼容，应了解您想要部署的传感器的功能，这一点至关重要。本地部署的传感器一般会通过 BACnet、LonWorks、Modbus、OPC 和 KNX 等操作技术协议进行通信。基于云的新型传感器通常支持蓝牙和 MQTT 等新型 IT 协议，并且/或通过 API 进行通信。

大多数 BMS 都支持 OT 协议，但支持用例 1b（云部署传感器）所示的新型 IT 协议的 BMS 却很有限，甚至没有。在这种情况下，为了实现数据传输，就需要为 BMS 设置一个网关或附加程序，如中间软件或自定义代码。

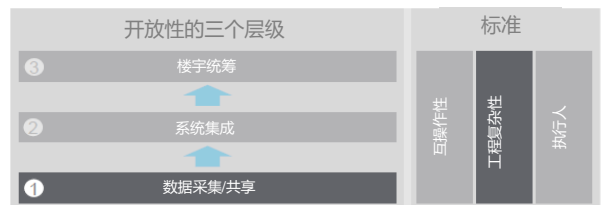
在用例 2 中，控制器之间的集成也会带来一些其他的互操作性问题。如下所示为与之有关的两项要求：

- **两个控制器的协议相同**——与传感器一样，集成能力取决于控制器和 BMS 使用和支持的协议。人们选择 OT 协议的可能最高，其中 BACnet 便是标准的 OT 协议。
- **所需数据点已公开**——尽管 OT 协议是开放协议并可实现互操作，但确保数据共享仍是一项挑战。如果设备制造商已选择公开所需数据值，那么其他使用相同语言的控制器将能获取这些数据。

例如，在用例 2 中，只有当两个控制器使用相同的 OT 协议并能够共享所需的确切数据时，才能将数据从一个现场控制器传递到另一个现场控制器（这种方式比集中数据更有效）。如果以上两个标准不成立，控制器之间就不能直接对话，而另一种选择是将数据直接迁移至楼宇控制器中，然后再从楼宇控制器分配数据。这种方法的效率较低，同时需要楼宇控制器充当不同现场控制器之间的网关。如果所需数据仅以专有方式公开，那么唯一的选择就是将您的楼宇控制器更换为支持该专有标准的控制器，或者如果供应商公开了协议定义（虽然这种情况不太可能），则可选择编写自定义驱动程序。

## 开放性标准 2——难度/工程复杂性

就 BMS 的开放性而言，应考虑传感器和/或控制器与系统实现可互操作的难度或工程复杂性。在 BMS 的集成和设置期间，有几个必要的操作，而这些操作的复杂性和耗时程度则取决于开放性程度。这种情况适用于上述三个用例。



## 点发现和映射

一方面，如果有自动发现功能，服务器便可发送一个广播，查看网络上的新设备，并自动查看任何新传感器或控制器（如 BACnet）。另一方面，如果没有自动发现功能时，点发现和映射的复杂性就会增加，也需要更多的手动操作。首先，您必须指定控制器（即 Modbus）的地址，然后在设备规范中手动查找寄存器，以找到所需数据点的特定寄存器号（地址）。

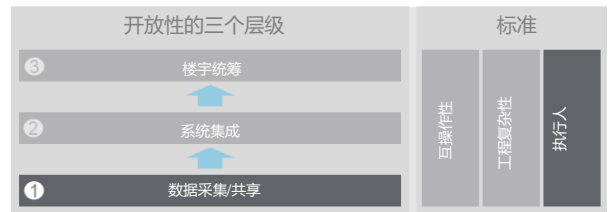
## 控制器和系统编程

BMS 编程涉及编写控制回路、配置警报、构建图形和设置时间表等。其中，固定功能、可配置和自由编程是三大控制器编程变体。有些情况下，固定功能设备的集成可能更简单，因为可能不需要专门的配置工具。然而，这些设备无法满足所有应用场景。

相较于无法支持多种协议的 BMS，支持多种协议的 BMS 的工程复杂性更低。通过支持多种协议，不同现场控制器的点可以在没有其他硬件/软件的情况下，在配置设置期间进行互操作，并在图形用户界面（GUI）上互联互通。当只支持一种协议时，如果想要与使用了不同协议的现场控制器的点实现互操作，就需要桥接器或网关来转换协议。这可能意味着需要一台额外的计算机来运行桥接器或网关，同时，也需要完成冗长且耗时的手工编程/映射。除了劳动力成本增长外，还有一些意料之外的或“隐藏”的成本，如随着时间的推移，还需获得相应许可并雇用维护人员。

## 开放性标准 3——执行人

第三个需要考虑的开放性标准是，是否有人可以完成集成工作，或者是否需要经过专门培训的人员来完成这项工作。如果每次系统变更都需要第三方协助，那 BMS 的运营将举步维艰。



通常认为，表 1 所示的控制逻辑任务需要一定程度的专门技能/资源，但是，日常操作不应也是如此。

如果完成相应任务所需的自定义工作量越多，用户的“封锁感”就会越强，因为他们不得不依赖专家，才能继续使用系统。这就涉及上文针对协议和公开必要数据展开的论述。如果没有相关协议并且尚未公开必要数据，就需要更多的自定义工作，以及更多的专业知识。当需要专业技能时，接受过良好培训并且有工作经验的人才将有助于确保系统的成功运行。

## 控制逻辑

## 日常操作

表 1

控制逻辑和日常操作任务；对开放系统的期望是内部工作人员能完成日常操作

- 新建楼宇 HVAC 控制的初始设置——可能需要设计、配置、BMS 工具编程方面的知识，以及了解相关集成至 BMS 的组件规格的人才。
- 更换传感器/执行器——如果仅仅是一个简单的电阻式温度传感器，那么只需要技术人员对控制器的 I/O 进行简单的重新布线。但是，如果设备使用了专有协议，就需要重新配置并更新软件，以连接至新设备。
- 添加一个新的区域控制器并公开所需数据点——这项任务可能需要进行初始设置的专家，来发现并补足控制循环等问题。
- 修改设定值（如室温）——包括壁装式控制器/温控器的设定值，以及 GUI 的设定值。
- 创建趋势日志和警报，以了解发生的事件以及关联数据——对于创建趋势日志和警报并将其链接至数据点，应该在无需第三方专家的情况下完成。
- 更新日程安排（如每日安排、节假日安排、季度安排）——这项任务应该无需任何经过严格培训的人员。

## 第 2 层： 系统集成

到目前为止，我们已经讨论了具有有限数据点的传感器和执行器等有限组件的集成或连接。BMS 不再仅仅局限于 HVAC 控制，而是需与其他系统集成，如照明系统，而照



明系统的控制器和传感器（照明传感器，调整百叶窗的执行器等等）本身具备潜在复杂性。为此，需加大集成力度，在第 1 层的基础上进一步发展。BMS 须以不同的方式使用数据，包括向第三方系统推送数据、从第三方系统中查询数据，配合与专有云通信的分析服务。

这种情况下，我们应针对“开放性”一词展开不同的思考。尽管在第 1 层中，开放性主要以协议和公开数据为中心。在第 2 层中，我们必须扩大范围并在生态系统中进一步摘要，以涵盖 BMS 和第三方系统之间共享数据的其他方法。接下来，我们将详细说明第 2 层的两个用例，并在图 4 中予以阐释。

### 用例 1：与其他楼宇系统集成

在医院的大楼中，入院/出院/转院系统有望连接至 BMS 上，以便明智地确定应在何时控制病房退房，通过了解病房的空置时间，医院可节省能源费用。

### 用例 2：人工智能/分析系统集成

医院的大楼经理希望在 BMS 中应用故障检测和诊断系统，该系统需要 BMS 各个方面的数据，包括传感器数据、输入数据、输出数据和内部逻辑，从而利用这些数据运行一系列算法，以确定系统是否可以高效运行，是否存在任何故障，或舒适度是否下降。如果存在这些问题，系统将协助诊断，并指出潜在原因。

图 4

## 系统集成应用场景



## 开放性标准 1——互操作性

相较于传感器和控制集成,本层级涉及的系统集成更为复杂。影响互操作性的关键话题包括: (1)API 与文件/数据库的使用情况比较, (2)语义/本体, (3)所需数据类型, 以及(4)验证。



## API、文件/数据库和操作技术协议的使用情况比较

在本层, 您可将部分系统与 OT 协议集成 (无需专门的软件工程师进行自定义编码, 当属首选方法), 但是, 您很可能遇到 REST API<sup>7</sup>。如果供应商想要系统具备“开放性”并将数据传输至另一个系统, 那么, 就需要通过 API 或 OT 协议公开数据。

在上文提到的用例 1 中, 医院的入院/出院/转院系统供应商创建了 BMS 使用的 API。在用例 2 中, BMS 供应商可创建供人工智能/分析系统使用的 API。

然而, 那些开放程度较低的系统并不会主动公开数据, 而是将数据存储起来并通过.csv 文件共享, 或直接存储到计划使用该数据的系统数据库中。在这些模式下, 如果不是编写该数据库的人, 其他人将很难理解数据结构。数据库模式越难, 花费的时间就越多, 集成成本也会随之增加。选择待集成系统时, 应选择那些可通过 API 或协议公开数据的系统。

## 语义/本体

语义标注 (事项标记) 和本体 (该等事项之间的关系) 对互操作性至关重要。如果想要在分析用例中完成人工智能集成, 需要人工智能系统很好地理解 BMS 中相关数据点的环境。如果分析系统想要查看楼宇的室温设定值, 那么系统将很难查看相关数据点, 也很难厘清每个数据点与楼宇之间的关系, 比如其所在的房间, 以及接收数据的设备。如果没有标准的语义和本体, 就需要有人来解释单体建筑的数据点命名约定 (如位置、控制设备)、了解各项信息、理解层次结构等。以标准方式公开数据可避免这项手动操作。目前, 我们拥有 [Brick](#) 和 [Haystack](#) 等不同的语义标准, 但是, 对许多供应商来说, 语义标准仍在发展中。在某些情况下, 特定行业 (例如医疗行业的 HL7) 有相应的数据标准, 从而降低了数据模式的理解难度。如适用, 请选择那些遵循该等标准的系统。

<sup>7</sup> <https://www.redhat.com/en/topics/api/what-is-a-rest-api>

## 所需数据类型

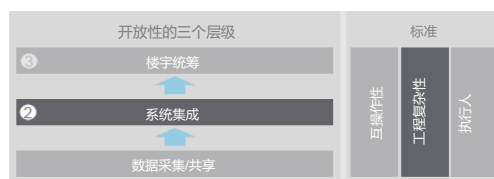
不同系统会采集不同类型的数据（即趋势记录、实时数据和报警数据等历史数据），而您获取数据的能力也可能受此影响。虽然可以访问实时数据，但您可能无法访问那些由单独历史数据库收集或存档的历史数据。在开放环境下，这就归结于系统是否拥有访问各类数据的方法。

## 验证

验证<sup>8</sup>机制作为一个限制因素，可能增加集成的复杂性，因此，也是系统间互操作性的重要考量因素。API 会传递不安全的数据，所以使用验证时，通常会存在中间件。泛化验证绝非易事。通过 UI 很难做到这一点。同时，泛化验证通常都是很模糊的，也需要为部分验证编写代码。虽然业内有简化验证流程并减少自定义代码的标准验证协议（即 OAuth），但是这些协议的使用频率并不高。在开放的环境下，应寻求具有下列属性的软件：支持非模糊的验证类型（如 OAuth），以及允许自定义开发中间件来访问使用专有验证的第三方。

## 开放性标准 2——难度/工程复杂性

当本层未使用标准协议时（这种情况较为常见），想要将系统连接起来，便需要进行自定义开发。



在某些情况下，仅需一天时间便能完成自定义开发，有时也需要花费数月。一般情况下，系统的开放程度越低，集成所需的工程时间便越多。当系统使用 web 服务、语义和本体等最优方法时，工程复杂性能降至最低。

为了简化设施运营，人们通常希望通过用户界面拉取集成系统中的信息（即用例 1 中的医院入院/出院系统和 BMS）。此外，如果将视频、网页、方案、设定点和数值集成并入一个控制面板中（“单一虚拟管理平台”），工程复杂性也会随之增加。如果无法直接自定义或修改 UI，以显示不同类型的数据流，那么，集成的有用性也会受限。同时，如果系统不支持此类自定义，那么，其开放程度也是较低的。

当您希望一个系统的第三方能控制另一个系统时，复杂性也会随之层层叠加。例如，在用例 2 中，如果我们想要分析系统控制 HVAC 系统的一个阀门，则可能需要改变 BMS 的控制回路逻辑，以允许第三方输入信息。这种情况下，可能需要自定义操作。如果系统所需的自定义操作越少，系统的开放程度就越高。

<sup>8</sup> <https://en.wikipedia.org/wiki/authentication>

### 开放性标准 3——执行人

我们在第 1 层讨论的控制逻辑任务和日常任务之间的区别，第 2 层同样适用。当还未实现“映射”自动化时，应当考虑由谁来执行这项工作。映射执行人需了解这两个领域，以便准确地映射数据。

一些情况下，在知晓特定楼宇的详细信息之后，才能决定应该由谁来完成这项任务，或者是否需要组建团队。

通常，BMS 供应商会组建一个团队专门负责与第三方系统集成的相关工作。如果供应商未组建上述团队，那么业主就需要雇用第三方来编写集成代码。无论是寻找和雇用咨询公司，还是之后花时间来学习并理解系统和楼宇，都会增加所需时间和成本。还需获取合规的文档、许可证和技术支持，这种情况下，学习曲线可能会变得越来越复杂。

在上文的人工智能/分析用例中，为了提取和映射数据，第三方分析系统供应商可能需要编写中间件，而为了允许第三方控制系统，BMS 供应商还需完成一些其他的工作。鉴于中间件的复杂性，可能会需要更多专门人士参与集成，从而降低开放性。

第 3 层——楼宇统筹，涉及跨系统全面协调。第 3 层和第 2 层类似，但规模更大。想要构建智能建筑，就要完成第 2 层的系统集成，并将集成范围扩大到所有楼宇系统。

通过简化复杂的楼宇任务并实现相应的自动化，楼宇统筹将有助于楼宇优化能源效率、运营效率、租户福利和租户生产力。因此，应设立明确的集成目标，确保有形资产的价值，而不是为了集成而集成。

我们认为，BMS 要么发挥“协调器”的作用，要么作为子系统集成并入统筹层。（后者的重点在于系统到系统的集成，已在第 2 层展开讨论。）作为楼宇协调器，BMS 不仅负责管理楼宇的 HVAC 控制，还需大规模地摄取数据、应用环境、操作 workflow 并将 workflow 用于楼宇内的任何其他系统，并进行任何组合操作。对于构建可持续、有韧性、高效以及以人为中心的楼宇而言，一个内聚系统是必不可少的。

#### 用例：智慧医院

为了建设“智慧医院”，需要在医院的不同系统间实现相互操作和数据共享，以确保患者满意度、舒适度和安全性，并节能增效。图 5 所示为医院中使用的一些系统。

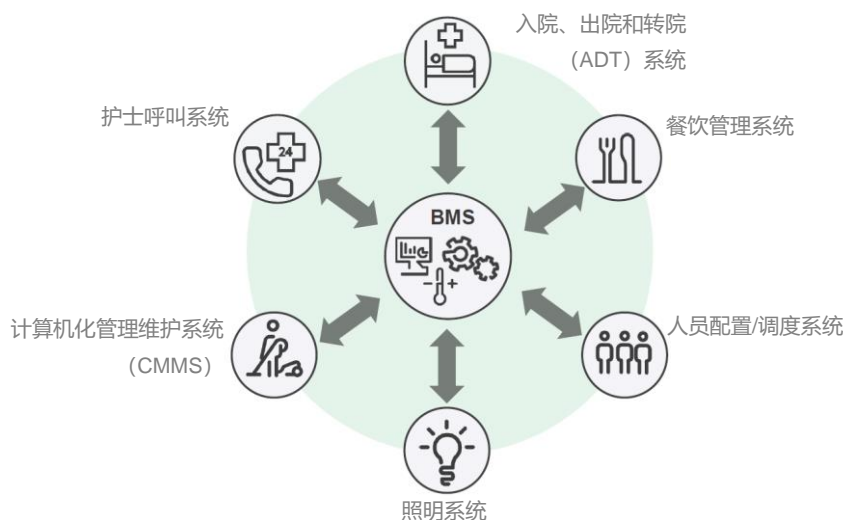


## 第 3 层:楼宇统筹



图 5

楼宇统筹医院用例



接下来这个例子便说明了智慧医院的具体情况。如果我知道患者的手术时间，以及患者回到病房静待恢复的时间，那么我就可以在相应的病房里调整设定值，减少无人期间的能耗，当患者返回病房休养时，再重新调整该等设定值，病房状态恢复如故。当患者不在病房时，还可自动安排维护人员入内开展例行维护，以免打扰患者。因此，可将上述所有功能并入一个 UI，从而设施经理、护士、医生、办公室工作人员和维护人员都可访问根据相同核心数据创建的情境视图。

### 开放性标准 1——互操作性

第 2 层中讨论的互操作性话题，同样适用于本层，包括：(1) API 与文件/数据库的使用情况比较，(2) 语义/本体，(3) 所需数据类型，以及(4) 验证。



然而，在本层，更希望 BMS 利用更多的工具（包括基于 UI 的工具，或编程工具，或两者皆有）来简化系统集成。

通常而言，这些集成都是双向的，因此其中可能包括更复杂的对象类型。而在其他层，该等数据的复杂程度较低，如数据值。在本层，您很有可能遇到更复杂的数据结构，如时间安排。在这种更为复杂的数据结构下，系统间更有可能出现语义差异，也可能出现无法在系统间直接映射出不同功能的情况。这种时候，为了弥补差距，就可能需要自定义规则和工作流，而复杂性也会随之增加。一个开放程度更高的 BMS 应该具备一个支持该等自定义规则和工作流的工具集。

## 开放性标准 2——难度/工程复杂性

第 3 层的目标和第 2 层类似，但规模更大。因此，复杂性和自定义需求也会随之增加。在本层，影响工程复杂性的关键考量因素共有四项：



### 展示

在开放系统中，协调器应能查看单一虚拟管理平台中各子系统的数据库。而相应的视图应可自定义，并符合运营商和租户的期望。同样重要的是，用户控制面板还应列有子系统的标准化数据（例如，相同的度量单位，时间尺度等等），并根据相关规则和工作流计算数据，以便为查看者提供可行见解。例如，在上述用例中，可视化控制面板对维护人员而言大有裨益，可一目了然地了解医院的可用病房以及需要维护的病房。为了避免不必要的编程/开发，能以简单的方式创建可视化控制面板（如拖放小工具，自动创建）至关重要。

### 点发现/映射

由于需要互操作和相互理解的系统数量不详，系统之间可能会存在一些协议和语义差异（例如，医院的住院系统使用 HL7 协议，而 BMS 使用 BACnet 协议）。协调器需理解这些不同的“语言”，并充当系统间的翻译。这是一个非常复杂的问题，而且在许多情况下，这些翻译需要自定义开发或建模，而这些只能由专家执行。如果想要实现大规模的翻译，就需要一个非常复杂的工具包。在评估 BMS 的开放性时，您应该考虑其通过 UI 可以实现的集成程度。只有最复杂的事项才需要完全自定义开发，即使是这样，理想情况下也应该在协调器的范围内完成，而不是作为一个单独的应用程序安装在协调器旁边。随着行业的发展，我们希望看到更多可用的 API，以简化语言翻译。

### 编程

在多个系统协同作业的情况下，协调器应能自定义工作流。从而，该等工作流可从各个子系统中获取楼宇的现有信息，以便为运营商和租户提供所需的体验和结果（例如医院用例）。在开放系统中，应在协调器的 UI 中开发该等工作流，且无需完全自定义开发。另一个重要的考虑因素是工作流在整个楼宇生命周期中适应变化的能力。换言之，如果子系统有内容新增/移动/删除，如何确保工作流适应这些变化是需要考虑的问题。因此，我们需要一个灵活、开放的系统，以便确保协调器知晓更改，并且在无需人工交互的情况下应对新条件。

### API

协调器用于未指定数量的楼宇子系统创建工作流的数据非常有价值。在第 2 层中，我们解释了 API 在其中的重要性。同时，对于其他类型的系统，甚至是楼宇情境以外的系统而言，这些数据也可能大有用处。一个完全开放的系统意味着其他第三方应用程序也能使用这些数据。例如，协调器中包含的数据是创建移动应用程序所必需的；或者电力公司想要在全市范围内进行数据分析。API 允许向外部公开数据。

### 开放性标准 3——执行人

正如我们在第 1 层和第 2 层中讨论的那样，当集成工作不再依赖标准协议时，便强烈需要软件程序员来执行与楼宇相关的自定义工作。



由于在第 3 层中，我们处理许多被捆绑在一起的子系统，因此很有可能需要将一个“开放”程度不那么好的子系统集成起来。这种情况下，需要软件开发人员参与其中的可能性也会随着增加，除非 UI 和驱动层非常复杂。接下来，我们将介绍三个可能需要不同技能和专门知识的集成活动。

- **设置 workflow**——大多数情况下，集成商<sup>9</sup>可使用一种经过少量训练的工具对协调器内的工作流进行配置。然而，在较为复杂的工作流中，则可能需要软件开发人员协助执行自定义功能。
- **设置自定义 UI**——大多数情况下，集成商可使用一种经过少量训练的工具对协调器内的 UI 进行配置。然而，对于较为复杂的 UI，为了创建独特的视觉效果或计算数值，就可能需要在软件开发人员的协助下使用自定义功能来扩展 UI。
- **连接至移动应用和第三方应用**——通过 API 在协调器之外公开的数据应具备足够的描述性，以便第三方应用程序集成商或软件开发人员在其应用程序内解释和利用该等数据，且无需专家支持。

## 总结

随着 BMS 功能的不断发展，其作用已远不止于 HVAC 控制。对于智能建筑的建设而言，BMS 发挥着关键作用。

人们在讨论和选择 BMS 时，通常会提出“开放性”要求。然而，业内尚未就此制定明确的定义、分类和标准，无法就此主题展开明智对话。本白皮书介绍了楼宇管理系统“开放性”评估框架，其中包括开放性的三个层级：(1)数据采集，(2)系统集成，以及(3)系统统筹。

每栋楼宇都是独特的，也有不同的目标。在各个层级中，系统的期望和挑战各有不同，同时，每层都是建立在上一层级之上的。在评价各层级 BMS 的开放程度时，应遵循以下三个关键标准：(1)互操作性，(2)工程复杂性，以及(3)执行人。

<sup>9</sup> 集成商是专门将各个子系统组成一个整体并实现协同工作的人或公司 ([https://en.wikipedia.org/wiki/Svs-tems\\_integrator](https://en.wikipedia.org/wiki/Svs-tems_integrator))

表 2 所示为框架内各层级“开放性”的关键要点。

表 2

开放性 BMS 框架内各层级的关键要点

层级	“开放性” BMS 应.....
1 - 数据采集/共享	<ul style="list-style-type: none"> <li>可在多种 OT 协议间实现互操作。</li> <li>支持本地协议扩展，限制与传感器、执行器和控制器通信所需的网关数量。</li> <li>日常操作不依赖于“专家”。</li> </ul>
2 - 系统集成	<ul style="list-style-type: none"> <li>有能力通过任何必要的方法进行系统集成。</li> <li>最大限度地减少系统集成所需的时间。</li> <li>在供应商专家团队的协助下，简化系统集成。</li> </ul>
3 - 楼宇统筹	<ul style="list-style-type: none"> <li>可作为协调器执行自定义规则和工作流。</li> <li>可支持展示、点发现/映射、编程和 API 自定义。</li> <li>允许程序员通过设置工作流、设置自定义 UI 和连接第三方应用程序来扩展功能。</li> </ul>

## 关于作者

**Jeffrey Bowman** 是施耐德电气数字楼宇首席技术官办公室的高级软件工程师，在施耐德电气数字楼宇团队任职已 10 年余年，是集成和聚合不同楼宇系统的爱迪生专家。

**Chris Megede** 是施耐德电气首席技术官办公室的软件架构师，在施耐德电气数字楼宇团队任职已 20 年余年，期间曾担任全球 BMS 架构总监以及物联网和云计划的产品经理

**Wendy Torell** 是施耐德电气科研中心的高级研究分析师，主要负责研究数据中心以及楼宇设计和运营的最佳实践，出版白皮书和文章，并开发 Trade Tools，以帮助客户优化设施的可用性、效率和资本支出/运营成本。同时，她还会针对可用性科学方法和设计实践，为客户提供咨询服务，以协助客户实现性能目标。Wendy 持有纽约斯克内克塔迪联合学院的机械工程学士学位，以及罗德岛大学的工商管理硕士学位。此外，Wendy 也是 ASQ 注册可靠性工程师。



资料来源



[下一代楼宇管理系统 \(BMS\) 的三个基本要素](#)

第 500 号白皮书



[浏览所有白皮书](#)

<https://www.se.com/ww/en/download/>



[浏览所有 TradeOff Tools™ 工具](#)

[tools.apc.com](https://tools.apc.com)

**注：**网络链接可能随时间推移失效。参考链接在本文撰写时可供访问，但目前可能无法访问。



联系我们

对此白皮书的反馈与评价，请点击：

施耐德电气科研中心

[dcsc@schneider-electric.com](mailto:dcsc@schneider-electric.com)