Module-5

Real-World Design Constraints- Introduction, Technical Design Constraints-hardware is popular again, Data representation and visualization, Interaction and remote control. **Industrial Automation**- Service-oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the web of things to the cloud of things, **Commercial Building Automation**- Introduction, Case study: phase one-commercial building automation today, Case study: phase two-commercial building automation in the future.

5.1 Introduction

This chapter outlines the technical design constraints to illustrate the questions that need to be taken into account when developing and implementing M2M and IoT solutions in the real world.

5.2 Technical Design Constraints- hardware is popular again

- The IoT will see additional circuitry built into a number of existing products and machines _ from washing machines to meters. Giving these things an identity, and the ability to represent themselves online and communicate with applications and other things, represents a significant, widely recognized opportunity.
- For manufacturers of products that typically contain electronic components, this process will be relatively straightforward. Selection of appropriate communications technologies that can be integrated with legacy designs (e.g. motherboards) will be relatively painless.
- The operational environments and the criticality of the information transmitted to and from these products, however, will present some unconventional challenges and design considerations.
- The IoT will, on the other hand, allow for the development of novel applications in all imaginable scenarios. Emerging applications of M2M and wireless sensor and actuator networks have seen deployment of sensing capabilities in the wild that allow stakeholders to optimize their businesses, glean new insight into relevant physical and environmental processes, and understand and control situations that would have previously been inaccessible.
- The technical design of any M2M or IoT solution requires a fundamental understanding of the specificity of the intended application and business proposition, in addition to heterogeneity of existing solutions.
- Developing an end-to-end instance of an M2M or IoT solution requires the careful selection, and in most cases, development of a number of complementary technologies.
- This can be both a difficult conceptual problem and integration challenge, and requires the involvement of the key stakeholder(s) on a number of conceptual and technological levels. Typically, it can be considered to be a combinatorial optimization problem _ where the optimal solution is the one that satisfies all functional and nonfunctional requirements, whilst simultaneously delivering a satisfactory cost-benefit ratio.
- This is particularly relevant for organizations wishing to compete with existing offerings, or for start-up ventures in novel application areas. Typically, capital costs in terms of

"commissioning" and operational costs in "maintenance" must be considered. These may be balanced by resultant optimizations.

5.2.1 Devices and Networks

- Devices that form networks in the M2M Area Network domain must be selected, or designed, with certain functionality in mind.
- At a minimum, they must have an energy source (e.g. batteries, increasingly EH), computational capability (e.g. an MCU), appropriate communications interface (e.g. a Radio Frequency Integrated Circuit (RFIC) and front end RF circuitry), memory (program and data), and sensing (and/or actuation) capability.
- These must be integrated in such a way that the functional requirements of the desired application can be satisfied, in addition to a number of nonfunctional requirements that will exist in all cases.

5.2.1.1 Functional requirements

Specific sensing and actuating capabilities are basic functional requirements. In every case with the exception of devices that might be deployed as a routing device in the case of range issues between sensing and/or actuating devices- the device must be capable of sensing or perceiving something interesting from the environment.

This is the basis of the application. Sensors, broadly speaking, are difficult to categorize effectively. Selecting a sensor that is capable of detecting a particular phenomenon of interest is essential. The sensor may directly measure the phenomenon of interest (e.g. temperature), or may be used to derive data or information about the phenomenon of interest, based on additional knowledge (e.g. a level of comfort). Sensors may sense a phenomenon that is local (i.e. a meter detecting total electricity consumption of a space) or distributed (e.g. the weather).

In many cases, sensing may be prohibitively expensive or unjustifiable at scale, and thus motivates the derivation of models that can reason over the sensor readings that are available. Air and water quality monitoring systems are typical of this type of problem. Given a particular phenomenon of interest, there are often numerous sensors capable of detecting the same phenomenon (e.g. types of temperature sensors), but have widely varying characteristics.

These characteristics relate to the accuracy of the sensor, its susceptibility to changing environmental conditions, its power requirements, its signal conditioning requirements, and so forth. In some cases, for example, a complementary (e.g. temperature) sensor is required in addition to the primary sensor such that variations in readings of the primary sensor that are caused by variation in temperature can be understood in context. Sensing principle and data requirements are also of essence when considering the real-world application. Consider a continuously sampling sensor, such as an accelerometer, versus a displacement transducer.

Displacement can be sampled intermittently, whereas if an accelerometer is duty-cycled, it is likely that data points of interest (i.e. real-world events) may be missed. Furthermore, the data requirements of the stakeholder must be taken into account. If all data points are required to be transmitted (which is the case in many scenarios, irrespective of the ability to reason locally within an M2M Area Network or WSN), this implies higher network

throughput, data loss, energy use, etc. These requirements tend to change on a case-by-case basis.

5.2.1.2 Sensing and communications field

The sensing field is of importance when considering both the phenomenon to be sensed (i.e. Is it local or distributed?) and the distance between sensing points. The physical environment has an implication on the communications technologies selected and the reliability of the system in operation thereafter. Devices must be placed in close enough proximity to communicate. Where the distance is too great, routing devices may be necessary.

Devices may become intermittently disconnected due to the time varying, stochastic nature of the wireless medium. Certain environments may be fundamentally more suited to wireless propagation than others. For example, studies have shown that tunnels are excellent environments for wireless propagation, whereas, where RF shielding can occur (e.g. in a heavy construction environment), communication range of devices can be significantly reduced.

5.2.1.3 Programming and embedded intelligence

Devices in the IoT are fundamentally heterogeneous. There are, and will continue to be, various computational architectures, including MCUs (8-, 16-, 32-bit, ARM, 8051, RISC, Intel, etc.), signal conditioning (e.g. ADC), and memory (ROM, (S/F/D)RAM, etc.), in addition to communications media, peripheral components (sensors, actuators, buttons, screens, LEDs), etc. In some applications, where it would previously have been typical to have homogeneous devices, a variety of sensors and actuators can actually exist, working collaboratively, but constituting a heterogeneous network in reality.

In every case, an application programmer must consider the hardware selected or designed, and its capabilities. Typically, applications may be thought of cyclically and logically. Application-level logic dictates the sampling rate of the sensor, the local processing performed on sensor readings, the transmission schedule, and the management of the communications protocol stack, among other things.

The ability to reconfigure and reprogram devices is still an unresolved issue for the research community in sensor networks, M2M, and the IoT. It relates both to the physical composition of devices, logical construction of the embedded software, and addressability of individual devices and security, to name a few. Operating systems are typically used to make programming simpler and modular for embedded systems designers, but each comes with conceptual and implementation differences that impact the ability to handle certain desirable features.

5.2.1.4 Power

Power is essential for any embedded or IoT device. Depending on the application, power may be provided by the mains, batteries, or conversion from energy scavengers (often implemented as hybrid power sources). The power source has a significant implication on the design of the entire system. If a finite power supply is used, such as a battery, then the

hardware selected, in addition to the application level logic and communications technology, collectively have a major impact on the longevity of the application. This results in short-lived applications or increased maintenance costs. In most cases, it should be possible to analytically model the power requirements of the application prior to deployment. This allows the designer to estimate the cost of maintenance over time.

5.2.1.5 Gateway

The Gateway, is typically more straightforward to design if it usually acts as a proxy; however, there are very few effective M2M or IoT Gateway devices available on the market today. Depending on the application, power considerations must be taken into account. It is also thought that the Gateway device can be exploited for performing some level of analytics on data transitioning to and from capillary networks.

5.2.1.6 Non-Functional Gateway

There are a number of nonfunctional requirements that need to be satisfied for every application. These are technical and non-technical:

- Regulations
  - For applications that require placing nodes in public places, planning permission often becomes an issue.
  - Radio Frequency (RF) regulations limit the power with which transmitters can broadcast. This varies by region and frequency band.
- Ease of use, installation, maintenance, accessibility
  - Simplification of installation and configuration of IoT applications is as yet unresolved beyond well-known, off-the-shelf systems. It is difficult to conceive a general solution to this problem. This relates to positioning, placement, site surveying, programming, and physical accessibility of devices for maintenance purposes.
  - Physical constraints (from several perspectives)
  - Can the additional electronics be easily integrated into the existing system?
  - Are there physical size limitations on the device as a result of the deployment scenario?
  - What kind of packaging is most suitable (e.g. IP-rated enclosures for outdoor deployment)?
  - What kind and size of antenna can I use?
  - What kind of power supply can I use given size restrictions (relates to harvesting, batteries, and alternative storage, e.g. supercapacitors)?

5.2.1.7 Financial Cost

Financial cost considerations are as follows:

- Component Selection: Typically, the use of these devices in the M2M Area Network domain is seen to reduce the overall cost burden by using non-leased communications infrastructure. However, there are research and development costs likely to be incurred for each individual application in the IoT that requires device development or integration. Developing devices in small quantities is expensive.
- Integrated Device Design: Once the energy, sensors, actuators, computation, memory, power, connectivity, physical, and other functional and nonfunctional requirements are

considered, it is likely that an integrated device must be produced. This is essentially going to be an exercise in Printed Circuit Board (PCB) design, but will in many cases require some consideration to be paid to the RF front-end design. This means that the PCB design will require specific attention to be paid to the reference designs of the RFIC manufacturer during development, or potentially the integration of an additional Integrated Circuit (IC) that deals with the balun and matching network required.

5.3 Data Representation

Each IoT application has an optimal visual representation of the data and the system. Data that is generated from heterogeneous systems has heterogeneous visualization requirements. There are currently no satisfactory standard data representation and storage methods that satisfy all of the potential IoT applications.

Data-derivative products will have further ad hoc visualization requirements. A derivative in these terms exists once a function has been performed on an initial data set _ which may or may not be raw data. These can be further integrated at various levels of abstraction, depending on the logic of the integrator. New information sources, such as those derived from integrated data streams from various logically correlated IoT applications, will present interesting representation and visualization challenges.

5.4 Interaction and remote control

To exploit remote interaction and control over IoT applications, connectivity that spans the traditional Internet (i.e. from anywhere) on the side of the application manager, or other authorized entity, to the end-point (i.e. an embedded device), continues to be a challenging problem. Aside from authentication and availability challenges, for most constrained devices, heterogeneous software architectures, such as event-based operating systems running on devices with significantly varying concurrency models, continue to pose significant challenges from a remote management perspective.

Elements of Device Management, specifically reprogramming and reconfiguration of deeply embedded devices, will be required, particularly for devices deployed in inaccessible locations. This requires, among others, reliability, availability, security, energy efficiency, and latency performance, to be satisfactory whilst communicating across complex distributed systems.

Another significantly under-researched topic is the definition and delivery of end-to-end quality of service (QoS) metrics and mechanisms in IoT type applications. These will be necessary if Service Agreements (SA) or Service Level Agreements (SLA) are to be defined in the case of service provisions for IoT applications _ which may or may not be desirable to the application owner. This will be situation-specific. End-to-end latency, security, reliability, availability, times between failure and repair, responsibility, etc., are all likely to feature in such agreements.

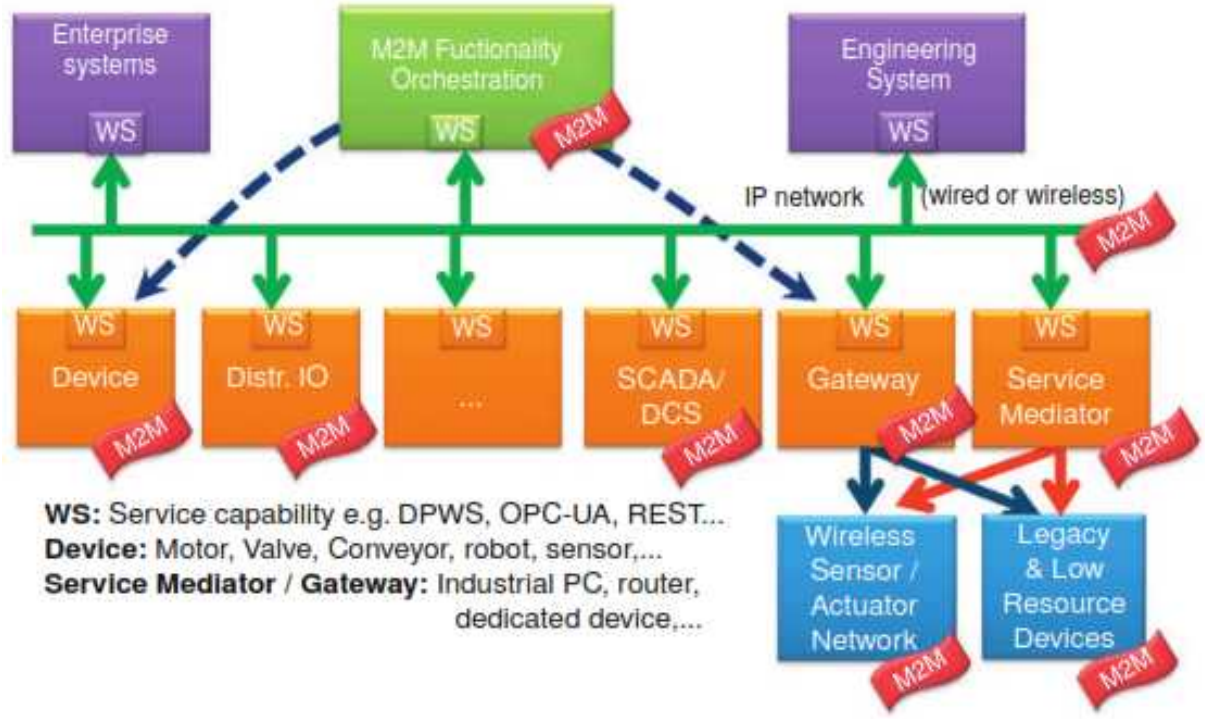
5.5 Industrial Automation

Service-oriented architecture-based device integration

The emerging approach in industrial environments is to create system intelligence by a large population of intelligent, small, networked, embedded devices at a high level of granularity, as

opposed to the traditional approach of focusing intelligence on a few large and monolithic applications. This increased granularity of intelligence distributed among loosely coupled intelligent physical objects facilitates the adaptability and re-configurability of the system, allowing it to meet business demands not foreseen at the time of design, and providing real business benefits.

The Service-Oriented Architecture (SOA) paradigm can act as a unifying technology that spans several layers, from sensors and actuators used for monitoring and control at shop-floor level, up to enterprise systems and their processes as envisioned in Figure 5.1. This common "backbone" means that M2M is not limited to direct (e.g. proximity) device interaction, but includes a wide range of interactions in a cross-layer way with a variety of heterogeneous devices, as well as systems and their services. This yields multiple benefits for all stakeholders involved. Such visions have been proposed and realized, demonstrating the benefits and challenges involved. Internet Protocol (IP)-based, and more specifically web technologies and protocols (e.g. OPC-UA, DPWS, REST, Web Services (WS), etc.), constitute a promising approach towards the fundamental goal of enabling easy integration of device-level services.

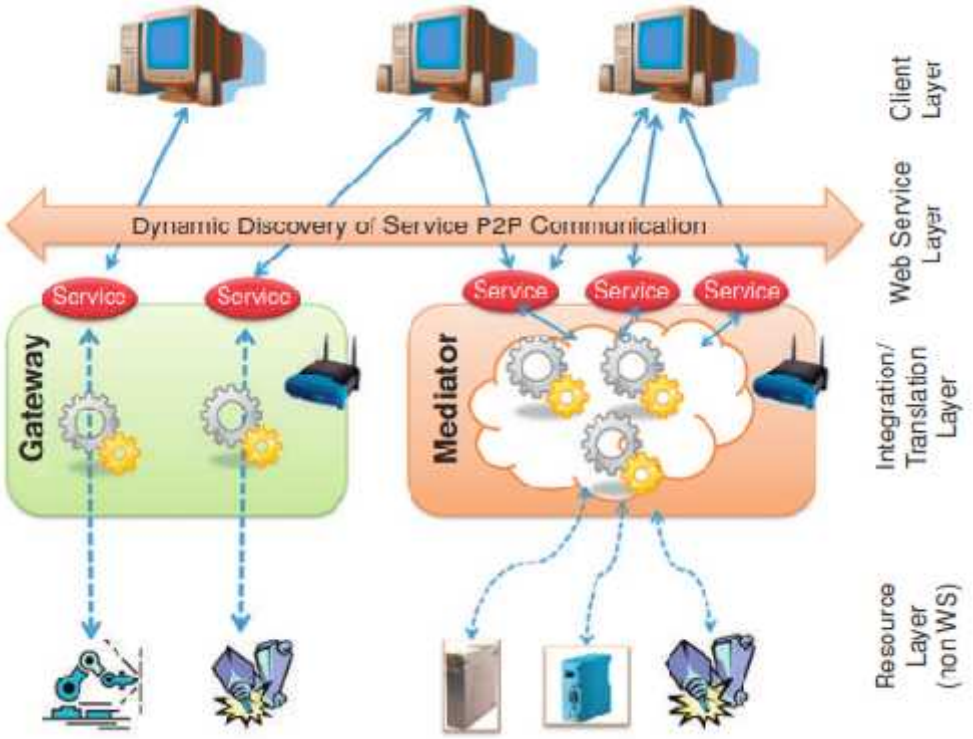


5.1 M2M SOA-based integration

with enterprise systems overcoming the heterogeneity and specific implementation of hardware and software of the device. Surely industry specific requirements for security, resilience, and availability of near real-time event information needs to be effectively tackled. The latter are also seen as key enablers for the new generation of enterprise system applications such as business activity monitoring, overall equipment effectiveness optimization, maintenance optimization, etc.

The SOA-based vision is not expected to be realized overnight, but may take a considerable time depending on the lifecycle processes of the specific industry, and may be impacted by micro- and macro-economic aspects. Hence, it is important that migration capabilities are provided so that we

can harvest some of the benefits today and provide a stepwise process towards achieving the vision. The concepts of gateway and service mediator, as depicted in Figure 5.2, can help towards this direction. Dynamic device discovery is a key functionality in the future M2M. As an example, Figure 11.3 depicts how Windows 7 can discover dynamically heterogeneous devices that are SOA-ready (i.e. equipped with web services; Devices Profile for Web Services, DPWS).



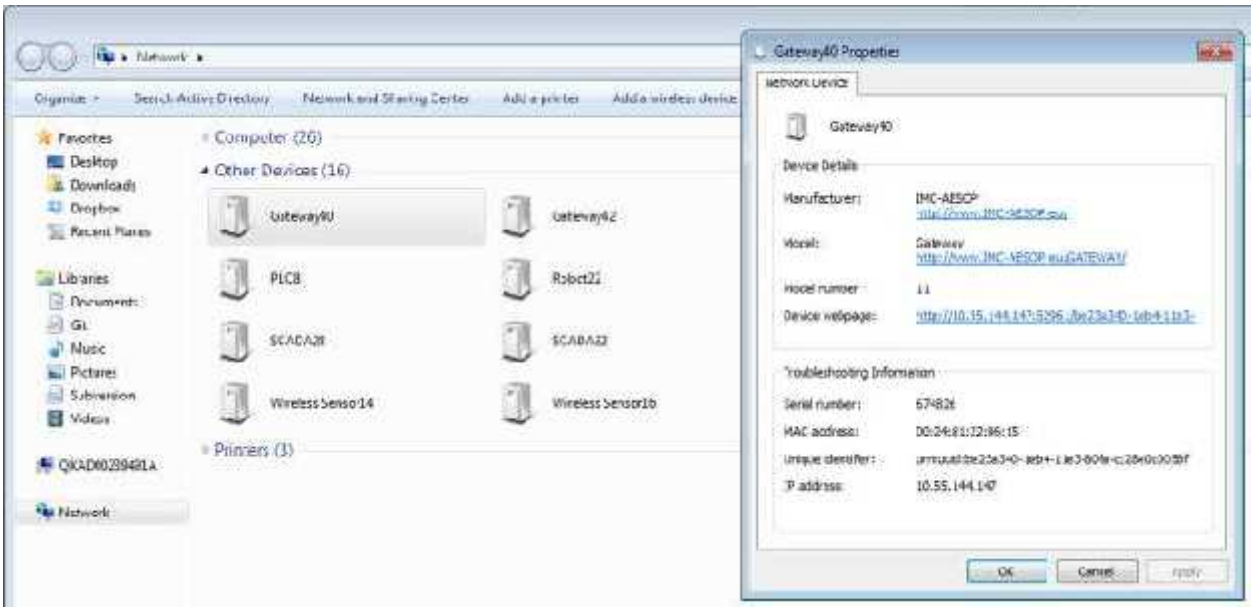
5.2 Non- Service-enabled device integration: Gateway vs. Service Mediator

A Gateway is a device that controls a set of lower-level non-service-enabled devices, each of which is exposed by the Gateway as a service-enabled device. This approach allows gradually replacing limited-resource devices or legacy devices by natively WS-enabled devices without impacting the applications using these devices. This is possible since the same WS interface is offered this time by the WS-enabled device and not by the Gateway. This approach is used when each of the controlled devices needs to be known and addressed individually by higher-level services or applications. Originally meant to aggregate various data sources (e.g. databases, log files, etc.), the Mediator components evolved and are now used to not only aggregate various services, but possibly also compute/process the data they acquire before exposing it as a service. Service Mediators aggregate, manage, and eventually represent services based on some semantics (e.g. using ontologies). In our case, the Service Mediator could be used to aggregate various non WS-enabled devices. This way, higher-level applications could communicate to Service Mediators offering WS instead of communicating to devices with proprietary interfaces. The benefits are clear, as we don't have the hassle of (proprietary) driver integration. Furthermore, now processing of data can be done at Service Mediator level, and more complex behavior can be created which was not possible before from the standalone devices.

As we can see in future IoT infrastructures dominated by billions of devices with different capabilities and needs, we have to consider how these integrate with each other and enable the

realization of new innovative approaches. This assumes increased integration and collaboration among the various layers existing in industries (i.e. from the shop floor up to enterprise systems). Several concepts and efforts are directed towards abstracting from the device-specific aspects and defining device-agnostic, but functionality-focused, layers of integration.

5.6 SOCRADES: realizing the enterprise integrated Web of Things



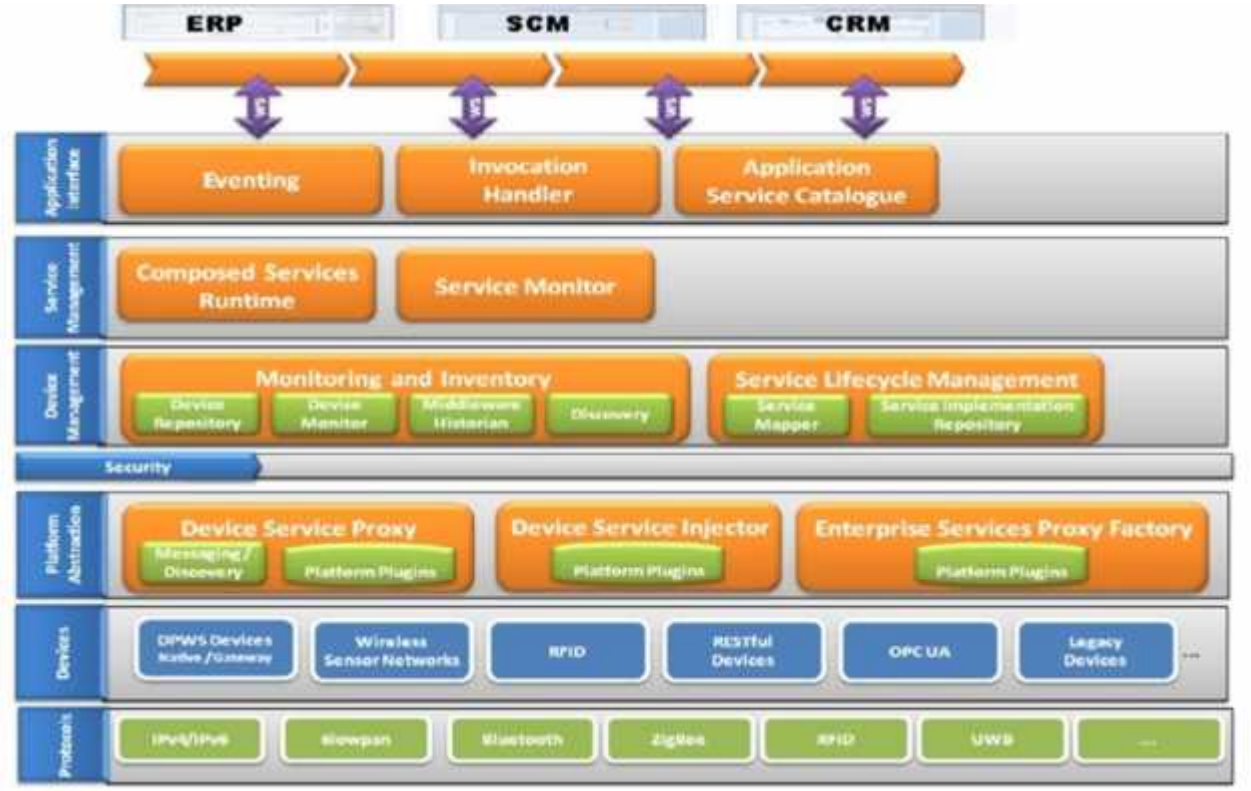
5.3 Dynamic device discovery via DPWS in Windows 7

Agility and flexibility are required from modern factories. This, in conjunction with the rapid advances in Information Technology (IT), both in hardware and software, as well as the increasing level of dependency on cross-factory functionalities, sets new challenging goals for future factories. The latter are expected to rely on a large ecosystem of systems where collaboration at large scale will take place. Mashing up services has proven to be a key advantage in the Internet application area; and if now the devices can either host web services natively or be represented as such in higher systems, then existing tools and approaches can be used to create mash-up apps that depend on these devices.

The SOCRADES Integration Architecture (SIA), as analyzed by Karnouskos et al. (2010a), enables enterprise-level applications to interact with and consume data from a wide range of networked devices using a high-level, abstract interface that features WS standards (Figure 5.4). One can clearly distinguish the various levels such as:

- **Application Interface:** This part enables the interaction with traditional enterprise systems and other applications. It acts as the glue for integrating the industrial devices, and their data and functionalities with enterprise repos and traditional information stores.
- **Service Management:** Functionalities offered by the devices are depicted as services here to ease the integration in traditional enterprise landscapes. Tools for their monitoring are provided.
- **Device Management:** Includes monitoring and inventory of devices, including service lifecycle management.

- **Platform Abstraction:** This layer enables the abstraction of all devices independent of whether they natively support WS or not, to be wrapped and represented as services on the higher systems. In addition to service-enabling the communication with devices, this layer also provides a unified view on remotely installing or updating the software that runs on devices.
- **Devices & Protocols**: These layers include the actual devices that connect over multiple protocols to the infrastructure. The respective plugins of course need to be in place so that they can be seamlessly integrated to SIA.



To realize discovery and interaction in a P2P way, a local gateway/service mediator is implemented. This prototype was named a Local Discovery Unit (LDU), and would enable the dynamic discovery of devices on premise and their coupling with the SIA. SIA has been used in several scenarios as proof of concept for the integration among different devices, both locally and with enterprise systems. Examples include (Karnouskos et al. 2010a):

- Integration between a programmable logic controller, a robotic gripper, and SunSPOT wireless sensor nodes, while these are monitored by the SAP Manufacturing Integration and Intelligence software (SAP MII), which is also responsible for the execution of the business logic.
- Event-based interaction between Radio Frequency Identification (RFID) Reader (product ID via the RFID tag), robotic arm (used to demo transportation), a wireless sensor which monitored the usage of an emergency button, an IP-plugged emergency lamp, and a web application monitoring the actual production status and producing analytics.
- Production planning, execution, and monitoring via SAP MII of a test-rig controlled by Siemens Power Line Communication (PLC) and communicating over OPC.

- Passive energy monitoring via the usage of sensors (Ploggs) and gateways.

5.7 IMC-AESOP: from the web of things to the cloud of things

Visionary approaches have been further developed in the industry-driven project IMC-AESOP (2013). There the vision of SOCRADES has been pushed forward by considering the rapid advances in hardware and software, as well as IT concepts. Therefore, we go beyond WS-enabled devices towards the cloud in order to harness its benefits, such as resource flexibility, scalability, etc. The result will be a highly dynamic flat information-driven infrastructure (Figure 5.5) that will empower the rapid development of better and more efficient next generation industrial applications, while in parallel satisfying the agility required by modern enterprises.

This vision is only realizable due to the distributed, autonomous, intelligent, pro-active, fault-tolerant, reusable (intelligent) systems, which expose their capabilities, functionalities, and structural characteristics as services located in a "service cloud."

Although today factories are composed and structured by several systems viewed and interacting in a hierarchical fashion following mainly the specifications of standard enterprise architectures, there is an increasing trend to move towards information-driven interaction that goes beyond traditional hierarchical deployments and can coexist with them. With the empowerment offered by modern SOAs, the functionalities of each system (or even device) can be offered as one or more services of varying complexity, which may be hosted in the cloud and composed by other (potentially cross-layer) services, as depicted in Figure 5.5.

This transition marks a paradigm change in the interaction among the different systems, applications, and users. Although the traditional hierarchical view coexists, there is now a flat information-based architecture that depends on a big variety of services exposed by the cyber-physical systems and their composition. Next generation industrial applications can now rapidly be composed by selecting and combining the new information and capabilities offered (as services in the cloud) to realize their goals.

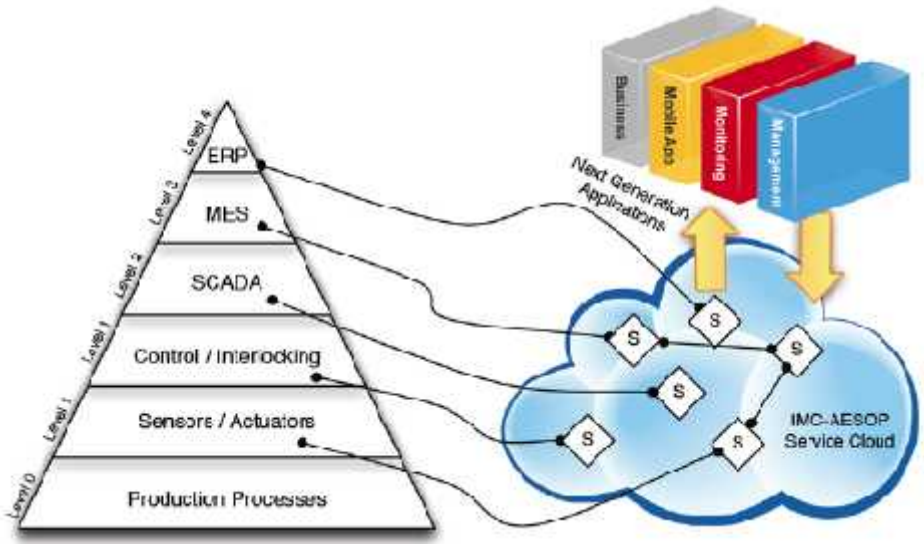


Figure No 5.5 Future industrial system view of cloud-based composition of cyber-physical services.

The envisioned transition to the future cloud-based industrial systems (Karnouskos & Colombo 2011, Karnouskos et al. 2012) is depicted in a Figure 5.6

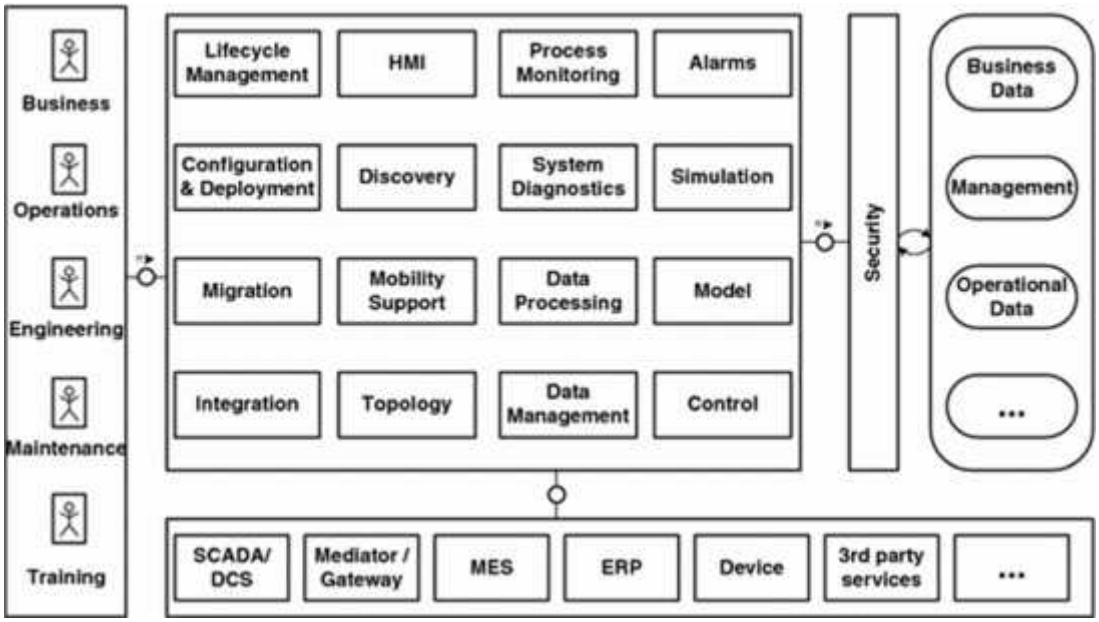


Figure No 5.6 IMC-AESOP cloud-based architecture vision

Several "user roles" will interact with the envisioned architecture (Figure 5.6), either directly or indirectly as part of their participation in a process plant. The roles define actions performed by staff and management, and simplifies grouping of tasks into categories such as business, operations, engineering, maintenance, engineering training, etc.

As depicted in Figure 5.6, it is possible to distinguish several service groups for which there have also been defined some initial services. All of the services are considered essential, with varying degrees of importance for next generation, cloud-based, collaborative automation systems. The services are to provide key enabling functionalities to all stakeholders (i.e. other services, as well) as cyber-physical systems populating the infrastructure. As such, all these systems can be seen as entities that may have a physical part realized in on-premise hardware, as well as a virtual part realized in software potentially on-device and in-cloud. This emerging "*Cloud of Things*" has the potential to transform the way we design, deploy, and use applications and cyber-physical systems.

Typical functionalities include alarms, configuration and deployment, some control, data management, data processing, discovery, lifecycle management, HMI, integration, simulation, mobility support, monitoring, security, etc. It is clear that this is a proposal that will need to be further refined in real-world scenarios, however, it clearly depicts a step towards a highly flexible M2M infrastructure for the automation domain that abstracts from devices and focuses on functionalities that can reside on-device, in-network, and harness the power of the cloud.

5.8 Commercial Building Automation-

Introduction

A Building Automation System (BAS) is a computerized, intelligent system that controls and measures lighting, climate, security, and other mechanical and electrical systems in a building. The purpose of a BAS is typically to reduce energy and maintenance costs, as well as to increase control, comfort, reliability, and ease of use for maintenance staff and tenants. Some example use cases:

- Control of heating, cooling, and ventilation based on time of day, outside temperature, and occupancy (e.g. Morning Warm-up).
- Automatic control of air handlers to optimize mix of outside air in ventilation based on, for example, inside temperature, pressure, and time of day.
- Supervisory control and monitoring to allow maintenance staff to quickly detect problems and perform adjustments.
- Outsourcing of monitoring and operations to a remote operations center.
- Data collection to provide statistics and facilitate efficiency improvements. Alarms for high carbon monoxide and carbon dioxide levels.
- Individual metering per apartment (to give incentive to save energy in multi-tenant buildings).
- Intrusion and fire detection.
- Building access control.

A BAS is normally distributed by nature to allow every sub-system to continue operation in case of failure in another system. A BAS consists of the following components (Figure .5.7):

- Sensors (i.e. devices that measure, such as thermometers, motion sensors, and air pressure sensors).
- Actuators (i.e. controllable devices, such as power switches, thermostats, and valves).
- Programmable logic controllers (PLCs) that can handle multiple inputs and outputs in real time and perform regulating functions, for example.
- A server which monitors and automatically adjusts the parameters of the system, while allowing an operator to observe and perform supervisory control.
- One or more network buses (e.g. KNX, LonWorks, or BACnet).

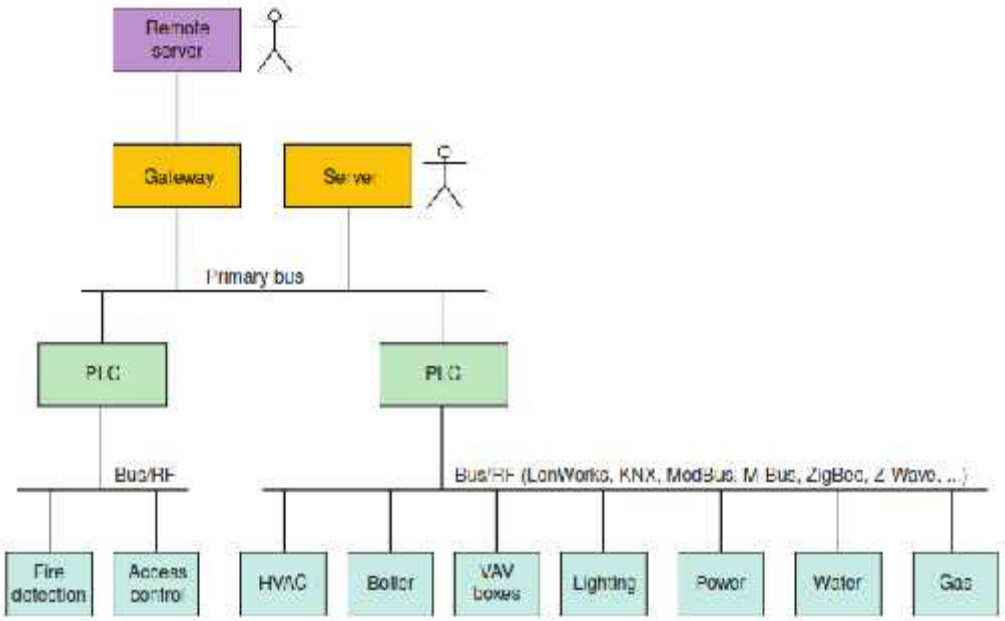


Figure No 5.7 Central parts in a BAS

The case study into two phases. In Phase One we give an example of what is commonly available today in regards to building automation. In Phase Two we explore new opportunities for building automation, such as the Smart Grid and the IoT.

We have divided the case study into two phases. In Phase One we give an example of what is commonly available today in regards to building automation. In Phase Two we explore new opportunities for building automation, such as the Smart Grid and the IoT.

13.2 Case study: phase one – commercial building automation today

13.2.1 Background

Company A wants to improve energy efficiency in their buildings and become GreenBuilding Partner (2013) certified, which requires lowering their energy consumption by at least 25%.

After discussions with a building automation company (Company B), they have come to understand that this is a very good investment that will quickly justify itself in terms of reduced energy costs. They agree on a five-step plan that starts with collecting data from the buildings, followed by analysis, adjustments, and connecting the systems in the buildings to a local server, and finally connecting the buildings to a remote operations center.

They can now start with collecting data from existing systems. In some cases this requires new meters to be installed. Everything from water usage to heat and electricity consumption is logged continuously, as well as performance of the ventilation and room temperatures.

By comparing the key performance indicators with comparative figures, the need for corrective actions is assessed and used as a basis for an action plan that consists of adjusting the existing systems and installing new software. These adjustments quickly increase the efficiency of the systems and are continuously optimized during the project. Examples of adjustments are hot water temperature, improved control of indoor temperatures, as well as better control of fan and pump operation to avoid unnecessary operation.

One of the most central features of the improved system is the new web-based E-report. It provides information about current energy consumption and other key parameters from the buildings. This information is used to make both short-term decisions as well as long-term planning. Everyone has access to the web portal because it's not only important for the maintenance staff, but also needed to create awareness across everyone in the company.

The next phase of the project consists of connecting the systems in the buildings and analyzing the dynamics to be able to perform

intelligent control. This both improves performance as well as reduces maintenance costs.

The final step to completion involves setting up a web-based Supervisory Control and Data Acquisition (SCADA) system for remote monitoring of the building systems. Through the web portal, the users can access information from the buildings in a coherent manner. Company A decided to outsource the operations and daily maintenance of the systems to Company B by utilizing their cloud-based offering. Company B's remote operations center is continuously monitoring the building systems. When building system operations deviate from their expected behavior, Company A's maintenance staff and their supervisors are notified by SMS and email. Typical events that can trigger a notification are, for example, mechanical failures or undesirable temperature deviations. Apart from notifications, Company B can also assist with equipment operation and adjustments remotely. For Company A, this arrangement is perfect because their in-house maintenance staff can respond to an alert 24 hours a day.

For Company A, the most important improvement has been the 35% energy reduction after the completion of the project. Another critical aspect has been the knowledge transfer from the experts at Company B that allows Company A to maintain the efficiency of the systems as well as the ability to continuously improve the operations of them.

13.2.2 Technology overview

Figure 13.2 depicts the setup for Company A.

Each building is equipped with a set of meters and sensors to measure temperature, water consumption, and power consumption, as well as one or more PLCs.

As seen in Figure 13.2, the PLCs perform real-time monitoring and control of the devices in the building. They also feature a user interface for configuration and calibration of (for example) the regulators, curves, and time relays. It is possible to remotely configure the PLCs from the Operations Center using the PLC Control system, which is connected to the PLC via a 3G-modem and an Internet Protocol (IP) modem that converts between RS-485 networks and Transmission Control Protocol/Internet Protocol (TCP/IP) networks. The PLCs communicate with the devices using several protocols, such as M-BUS, analog, digital, and Z-Wave, which is a low-power radio mesh-network technology. All logic necessary to operate the buildings is contained within the PLCs, allowing for minimal bandwidth requirements on the connection towards the

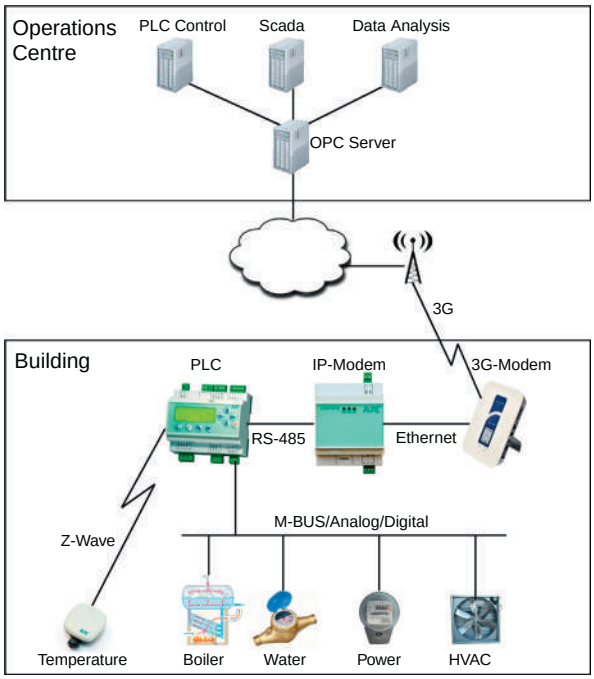


FIGURE 13.2

Illustration of the BAS.

Operations Center as well. It also means that the building systems can remain fully operational during periods of network outage.

The OLE for Process Control (OPC) server provides access to data, alarms, and statistics from the PLCs. When a value is requested from a user, a request is sent from the user's OPC Client to the OPC Server, containing an OPC Tag that identifies which PLC to contact and which value to ask for. The type of OPC communication used is called OPC Data Access. The OPC server then contacts the PLC in question and asks for the value using a protocol supported by the PLC (LonWorks or ModBus).

The SCADA system is used for operational monitoring of the buildings and provides information from all the relevant building systems. It uses the open and standardized OPC protocol, which enables integration with devices from many different vendors. The maintenance and operations staff can connect to the system using a web browser with a username and password to access dynamic flowcharts, drawing tools, timers, set points, actual values, historic readings, alarm management, event logs, as well as configuration for notifications over email, fax, or SMS.

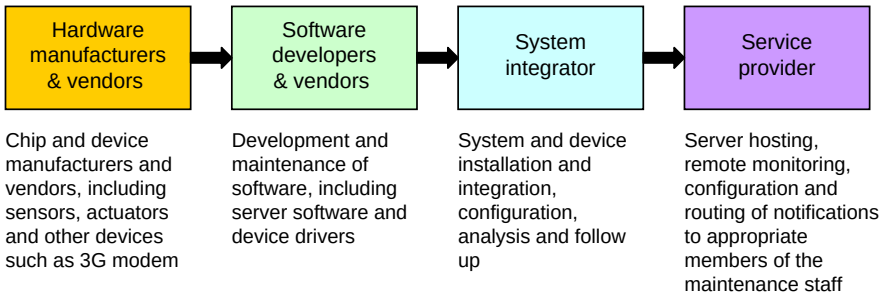


FIGURE 13.3

Applied value chain for Company A's system.

The Data Analysis server logs all historical readings from the buildings and makes it possible to follow up on different aspects of the energy and resource consumption, satisfying the varying needs of the tenants, economy department, and landlord. Through the OPC server it's possible to gather readings from all the building systems, regardless of vendor. Typical reports include trends, cost, budget, prognosis, environment, and consumption of electricity, heating, water, and cooling.

13.2.3 Value chain

Figure 13.3 shows an applied value chain.

13.3 Case study: phase two – commercial building automation in the future

13.3.1 Evolution of commercial building automation

Two major factors will drive the evolution of Building Automation: information and legislation (Figure 13.4).

Access to well-packaged information will provide the basis needed for decisions and behavioral changes. This can (for example) be electricity prices or where and when energy is used, and will allow for well-founded decisions that provide the best results.

Legislation, and taxes or tax credits to some degree, will provide the second driver. Legislative demands on green buildings and the Smart Grid will give rise to new opportunities, such as Demand/Response, Micro Generation, and Time-of-Day Metering.

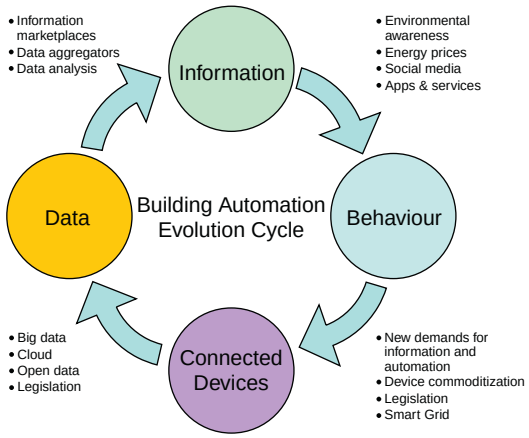


FIGURE 13.4

Building automation evolution cycle.

Market growth will result in economies of scale, standardization and commoditization, driving down prices, and increasing availability of devices and services. It will be possible to buy advanced devices off-the-shelf, perform installation, and connect them directly to service providers on the Internet.

13.3.2 Background

A few years have passed and Company A has decided to outsource the maintenance of its buildings to a local contractor who provides services to several other customers in the neighborhood. This will save money since that will enable them to utilize a shared caretaker pool.

At the same time they plan to upgrade their buildings to become fully automated with, for example, occupancy sensors, automated lighting, and integrated access control. To make this cost efficiently, they intend to make use of the existing IP infrastructure in the buildings, which also saves on operating expenditures as the network administrators can also manage the BAS infrastructure. According to studies, a converged IP and BAS network can reduce maintenance costs by around 30% while also lowering the initial investment for installation and integration by around 20% (according to studies performed by Cisco®). A shared infrastructure also leads to increased energy efficiency.

New political incentives in regards to energy efficiency have increased the development pace in the building automation area. Many neighboring buildings in Company A's area are now fitted with building automation,

which allows for sharing of information and resources. The increased customer base has also enabled new niches in the value chain, which has been split up to a large degree. Where before the rule was to have one single integrator and service provider, we now see a multitude of new actors, such as specialized service providers for remote monitoring, security, optimization, data collection, and data analytics. This allows Company A to choose freely what combination of service providers to use, while also providing a smooth transition when moving to a new provider. This is made possible by a new niche in the value chain: the Cloud Service Broker (Figure 13.5).

The process of integrating with the maintenance contractor's systems is simplified by the service broker because it provides immediate access to Company A's BAS. The caretakers can use their own specialized software as the service broker provides a bridge that can convert between several common protocols used for building automation.

When it comes to selection of devices, Company A opts for using standardized protocols to avoid vendor lock-in. They also decide to keep certain parts of the old system, as these would be too costly to replace. To still benefit from a fully integrated system, they also invest in a constrained application protocol (CoAP) gateway that translates between legacy devices and the new system.

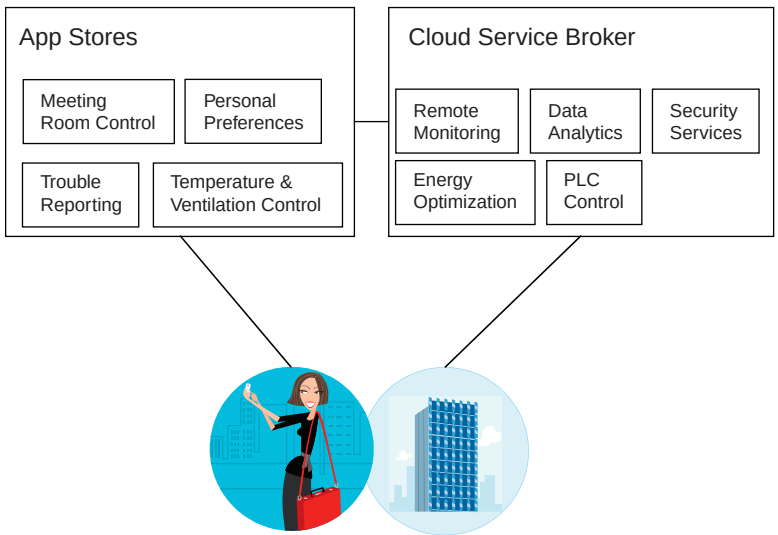


FIGURE 13.5

Cloud Service Broker.

By exporting historical data and configuration parameters from the old OPC Server it's possible for Company A to choose new service providers that can replace the old systems for PLC control, SCADA, and data analytics with a minimum of manual effort.

As an added service, the new platform also provides data brokering. This gives access to a multitude of data sources, such as the following:

- Historical and current KPIs to similar buildings.
- Integration with local government facilities.
- Weather forecast information.
- Utility prices, both current and future.

Apart from providing access to new service providers, the cloud broker also hosts a client API that enables third-party app developers to create smartphone applications. A number of users have purchased apps that allow them to do the following, for example:

- Control HVAC settings in meeting rooms.
- Report problems and service requests.
- Integrate with Outlook to adjust meeting rooms in advance.
- Create personal profiles to automatically adjust room settings.
- View instant and historical personal energy consumption and compare to others using social media.

13.3.3 Technology overview

Thanks to the rapid development of technologies for IP Smart Objects, it's now possible to use IP for both constrained devices, such as battery-powered sensors and actuators.

The new system is to a large degree based on IP technology (Figure 13.6). There are several IP-based protocols to select from, but in this case CoAP and Sensor Markup Language (SenML) were selected. CoAP provides both automatic discovery as well as a semantic description of the services the device provides. This drastically reduces installation costs, as much less configuration is needed. CoAP is similar to Hypertext Transfer Protocol (HTTP), but is binary to reduce the size of the messages. It also defines a Representational State Transfer (REST)-like Application Programming Interface (API) optimized for M2M applications. As with HTTP, a format for the content is also needed, in this case SenML, which is used as a format for sensor measurements and device parameters.

As mentioned before, there are still a few legacy devices, and these need a gateway to enable communication with the IP-based systems.