International Telecommunication Union

ITU-T Technical Report

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(15 September 2017)

YSTR-M2M-IDE
oneM2M – Industrial domain enablement

Technical Report ITU-T YSTR-M2M-IDE

oneM2M – Industrial domain enablement

Summary

This Technical Report collects the use cases of the industrial domain and the requirements needed to support the use cases collectively. Furthermore, this Technical Report also identifies necessary technical work needing to be addressed while enhancing future oneM2M specifications.

History

This document contains Version 0 of the ITU-T Technical Report on "*oneM2M Industrial domain enablement"* approved at the ITU-T Study Group 20 meeting held in Geneva, 4-15 September 2018.

Keywords

Industrial, interworking, oneM2M, OPC-UA, use cases.

Table of Contents

Technical Report ITU-T YSTR-M2M-IDE

oneM2M – Industrial domain enablement

1 Scope

This Technical Report collects the use cases of the industrial domain and the requirements needed to support the use cases collectively. In addition, it identifies the necessary technical work needed to be addressed while enhancing future oneM2M specifications.

2 References

The following ITU-T Technical Reports and other references contain provisions which, through reference in this text, constitute provisions of this Technical Report. At the time of publication, the editions indicated were valid. All Technical Reports and other references are subject to revision; users of this Technical Report are therefore encouraged to investigate the possibility of applying the most recent edition of the Technical Reports and other references listed below. A list of the currently valid ITU-T Technical Reports is regularly published. The reference to a document within this Technical Report does not give it, as a stand-alone document, the status of a Technical Report.

None.

3 Terms and definitions

3.1 Terms defined elsewhere

None.

3.2 Terms defined in this Technical Report

None.

4 Abbreviations and acronyms

For the purposes of the present document, the terms and definitions given in [b-ITU-T Y.4500.11] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in [b-ITU-T Y.4500.11].

A&E	Alarms and Events
ACP	Access Control Policy
AES	Advanced Encryption Standard
CR	Change Request
CSE	Common Services Entity
CSF	Critical Success Factor
DA	Data Access
DCS	Distributed Control System
DMZ	Demilitarized Zone
DoS	Denial of Service
DSL	Digital Subscriber Line
DTLS	Datagram Transport Layer Security

ERP	Enterprise Resource Planning
FIPS	Federal Information Processing Standardization
FT	Flow Transmitter
GPS	Global Positioning System
GSM	Global System for Mobile Communication
HDA	Historical Data Access
HMI	Human Machine Interface
IACS	Industrial Automation and Control System
ID	Identifier
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
IIC	Industrial Internet Consortium
IN	Infrastructure Node
IoT	Internet of Things
IPE	Interworking Proxy application Entity
ISDN	Integrated Services Digital Network
LAN	Local Area Network
LI	Level Indicator
MES	Manufacturing Execution Systems
MIC	Message Integrity Code
MN	Middle Node
NSE	Network Service Entity
OPC	Open Process Communications
PLC	Programmable Logic Controller
QoI	Quality of Information
QoS	Quality of Service
RBAC	Role-based Access Control
SCM	Supply Chain Management
SHA	Secure Hash Algorithm
SL	Security Level
SMB	Standardization Management Board
SMLC	Smart Manufacturing Leadership Coalition
SOA	Service Oriented Architecture
TLS	Transport Layer Security
UA	Unified Architecture
UMTS	Universal Mobile Telecommunications System
VPN	Virtual Private Network

WiFi	Wireless Fidelity
WLAN	Wireless Local Area Network
XML	extensible Markup Language

5 Conventions

None.

6 Introduction to industrial domain

6.1 Industrial domain overview

In previous industrial domains, the information exchange from factory-to-factory or centre-to-factory needed support from humans. Normally the exchange is non-synchronous, discrete, inefficient and unable to achieve the capacity to respond rapidly to market changes.

Currently, M2M technologies are considered to achieve the communication and interaction from machine-to-machine without human support. It brings opportunities to achieve synchronous, continuous and effective information exchange in manufacturing scenarios. Based on M2M, new manufacturing methods can be suitable to increase complex requirements of future market needs.

Many industrial companies are aware of the potential power to update traditional manufacturing systems by introducing M2M technologies. They are not restricted to the technical requirements, such as improving the performance of productivity, quality, delivery, cost reduction and security, but also new opportunities to cooperate with other domains for mass production, and the potential to build the new architecture for next generation industry. Figure 6.1-1 is an example architecture.

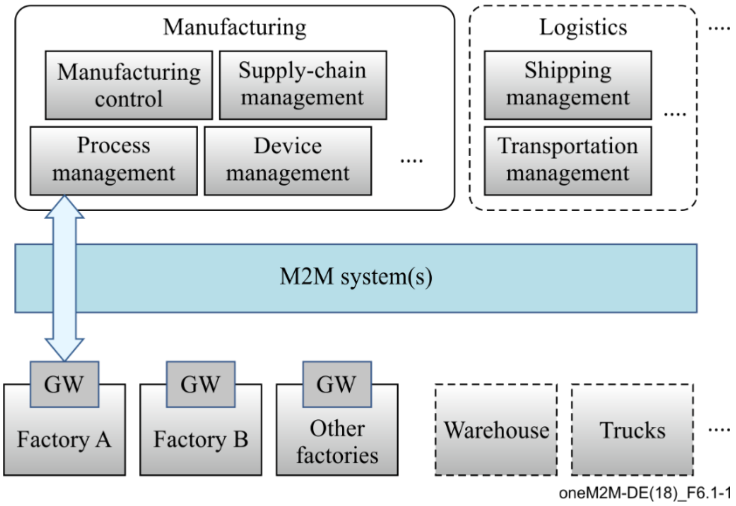


Figure 6.1-1 – Industrial domain architecture

In Figure 6.1-1, factories will be connected with manufacturing services via the M2M system(s). Generally, the gateway in the factory will collect data from the factory and send it to manufacturing services in a management centre. The service will be initiated by different management modules and sent to factories.

In addition, with the M2M system(s), the complex service can be sent to several factories synchronously, to enable effective collaboration between factories. Every factory is expected be able to make accurate decisions and to operate effectively, because it can work based on the results of data analysis and the data is from all the factories rather than from only one. The management centre with manufacturing services is also expected to be able to make accurate decisions by utilizing field data from all factories, and also via other support systems, such as cloud computing, to improve efficiency of local or global services.

In the future, if more and more industry related domains, such as logistics and power management systems, can be connected into the M2M system, resources (e.g., warehouses, trucks, ships, power) can be integrated efficiently. Therefore, more flexible services will be created to face this complex situation.

As the oneM2M architecture provides general application layer, common services layer and the underlying network services layer, and will be connected with other vertical systems, it is important to consider the integration of industrial domain systems with the oneM2M architecture.

6.2 Technology trends in industrial domain

To accelerate the update of manufacturing systems, many worldwide organizations have been established and have started making efforts.

In June 2014, the International Electrotechnical Commission (IEC), Standardization Management Board (SMB) set up a Strategy Group, SG8, to deal with a number of tasks related to smart manufacturing [b-IEC TC].

Table 6.2-1 – Industrial domain research in IEC SMB SG8

Mission and scope	• Develop a function model/reference architecture that helps to identify gaps in standardization based on to-be-collected use cases • Develop a common strategy for the implementation of Industry 4.0 • Extend standards towards: environmental conditions, security, properties, energy efficiency, product and functional safety
Technical keywords	• Industrial process measurement, control and automation • Application: semantics relationships descriptive technologies • Services: web services/service oriented architecture (SOA) repositories /cloud dependable connections • Communication: data access real-time communications

The Industrial Internet Consortium (IIC) was founded in March 2014 to bring together the organizations and technologies necessary to accelerate growth of the industrial Internet by identifying, assembling and promoting best practices [b-IIC website].

Table 6.2-2 – Industrial domain research in [b-IIC Engineering] and [b-IIC Engineering Update]

Mission and scope	• Productivity and efficiencies can be improved by production process governing themselves with intelligent machines and devices • Real-time data report from handheld digital device • Wearable sensors track location of employees in case of emergency • Future scenarios: new steering instruments will interlink things to ensure the entire value chain and trigger adjustments on the factory floor in case of chain changing; raw materials will be programmed to record standard process and their customer to realize automatic customization
Technical keywords	• Representative use case areas include connectivity, logistics, transportation, and healthcare • Key capabilities system characteristics including resilience, safety and security (e.g., key system characteristic, intelligent and resilient control, operations support, connectivity, integration and orchestration, security, trust and privacy, and business viewpoint) • Data management and analytics • Security: endpoint security, secure communications and security management and monitoring (currently focused on general security use case)

[b-IEEE P2413 report] defines an architectural framework for the Internet of things (IoT), which includes descriptions of various IoT domains including the industrial domain and is sponsored by the Institute of Electrical and Electronic Engineers (IEEE)-SA [b-IEEE P2413].

Table 6.2-3 – Industrial domain research in [b-IEEE P2413] and [b-IEEE P2413 presentation]

Mission and scope	• Ranges from the connected consumer to smart home and buildings, e-health, smart grids, next-generation manufacturing and smart cities • Promote cross-domain interaction instead of being confined to specific domains
Technical keywords	• Energy efficiency during data transmission • Areas of interest: industrial Internet, cross sector common areas, common architecture, security safety privacy

The Smart Manufacturing Leadership Coalition (SMLC) is a non-profit organization committed to the development and deployment of smart manufacturing systems. SMLC activities are built around industry-driven development, application and scaling of a shared infrastructure that will achieve economic-wide impact and manufacturing innovation [b-SMLC website].

Table 6.2-4 – Industrial domain research in SMLC [b-SMLC presentation]

Mission and scope	• To build a cloud-based, open-architecture platform that integrates existing and future plant level data, simulations and systems across manufacturing seams and orchestrate business real-time action
Technical keywords	• Cloud-based networked data • Enterprise real-time • Plant-level data • Information and action • Security

Plattform Industrie 4.0 is the central alliance for the coordination of the digital structural transition in German industry and unites all of the stakeholders from business, associations, trade unions and academia. Results so far have been summarized under the title "Reference Architecture Model Industrie 4.0 (RAMI4.0)". RAMI 4.0 provides a conceptual superstructure for organizational aspects of Industrie 4.0, with emphasis on collaboration infrastructures and on communication structures. It also introduces a concept of an administration shell that covers detailed questions on semantic standards, technical integration and security challenges. RAMI4.0 will be published as DIN SPEC 91345 "Reference Architecture Model Industrie 4.0" (RAMI4.0).

Table 6.2-5 – Industrial domain research in Platform Industrie 4.0 [b-Industrie 4.0]

Mission and scope	• Identify all relevant trends and developments in the manufacturing sector and combine them to produce a common overall understanding of Industrie 4.0 • Develop ambitious but achievable joint recommendations for all stakeholders, that serve as the basis for a consistent and reliable framework • Identify where action is required on standards and norms and actively express recommendations for national and international committee work
Technical keywords	• Reference architectures, standards and norms Incorporate existing norms and standards in RAMI4.0 (Reference Architecture Model Industrie 4.0). RAMI4.0 is an initial proposal for a solution-neutral reference architecture model • Research and innovation Evaluate current case studies to identify research and innovation requirements from the industry perspective

Table 6.2-5 – Industrial domain research in Platform Industrie 4.0 [b-Industrie 4.0]

	• Security of networked systems Resolve the outstanding issues concerning secure communication and secure identities of value chain partners Detect cyber attacks on production processes and their implications

Based on the information above and the current oneM2M architecture, the technology trends below are becoming more and more important:

– data management and analytics

In some industrial organizations, data management and data analytics are independent layers for data processing (such as filtering and catalogue management) and data analytics. Since large amounts of data are generated in industrial scenarios, further functionality design for data management and data analytics critical success factors (CSFs) may need to be considered in oneM2M.

– real-time command and control

M2M technologies enable real-time response manufacturing practices in complex supplier networks. Realizing real-time command and control by highly available and time critical technologies will bring benefits to process automation and the optimization of supply chains. Use cases with real-time command and control features may need to be considered in oneM2M. Additionally, requirements from these use cases may need to be taken into consideration.

– connectivity

Since connectivity in the industrial domain needs to co-exist and evolve with legacy protocols, legacy connectivity (both wired and wireless) and legacy wiring, connectivity for manufacturing processes needs to be considered and this may have an impact on network service entity (NSE) functionalities.

– security

Increased networking and wireless technologies are the main security concerns for industrial companies. Undoubtedly, the risk trade-off will not stop companies from manufacturing evolution. Thus, a renewed risk for management and ensuring security for the industrial domain may need to be considered.

Meanwhile more trends, such as web services over M2M devices and protocols in industrial domain, will be further tracked and analysed.

7 Use cases

7.1 An industrial use case for on-demand data collection for factories

7.1.1 Description

In factories, a lot of data are created from programmable logic controllers (PLCs) every second, and data are utilized to monitor production lines. This data is available via industrial bus systems, e.g., real-time Ethernet. In order to monitor remotely, data is gathered by the M2M service platform that needs to interface with such industrial bus systems via M2M gateways. However, it is difficult to gather all data to the M2M service platform because sometimes more than 1mega bit data is created per second. In such cases, only necessary data is gathered depending on situations and filtering/ pre-processing of the raw data needs to be performed at the gateways.

This use case proposes that the oneM2M system offers pre-processing capabilities, e.g., rule-based collection policies (averages, thresholds, etc.). These rules (e.g., in extensible markup language (XML) format) are called "data catalogues".

7.1.2 Source

Not applicable.

7.1.3 Actors

- PLC: controls sensors and devices in a production line according to embedded programs. It also has interface to real-time Ethernet. It broadcasts data related to the production line to real-time Ethernet.
- M2M gateway: provides an interface from the real-time Ethernet to the oneM2M system. An application on the gateway collects necessary data from real-time Ethernet according to the configuration called data catalogue, and send collected/pre-processed data to M2M service platform.
- M2M service platform: stores data gathered from gateway(s), and provide data to applications. It also manages data catalogue in gateway(s).
- Application: an M2M Application in the infrastructure domain that monitors production lines by using collected data in M2M service platform, and send change request (CR) of data catalogue depending on situations.
- Real-time Ethernet: a technology standardized in IEC TC 65 [b-IEC TC 65]. Ethernet is used at the physical layer, but upper protocol is designed for industry purpose. In this use case, broadcast protocol is assumed. On top of Ethernet cable, data is broadcast with ID. Address configuration is not necessary here.
- Internet connection: a M2M service platform and gateway(s) are connected by the Internet physically.

7.1.4 Pre-conditions

- PLCs and the gateway are connected to the real-time Ethernet. PLCs broadcast data to the real-time Ethernet. The gateway is configured to pick up necessary data from the real-time Ethernet.
- On top of the Internet, a virtual private network (VPN) connection is established between the M2M service platform and the gateway(s).
- The data catalogue is managed by the M2M service platform.

7.1.5 Triggers

- Data catalogue is configured for the gateway to pick up data in the real-time Ethernet.

7.1.6 Normal flow

1) The gateway picks up the broadcasted data. It picks up only data that matches conditions described in the data catalogue. If data does not match the conditions, the gateway ignores the data.

2) The gateway sends the collected data to the M2M service platform.

3) The M2M service platform receives the data and stores it.

4) The application utilizes the data. For example, it monitors the status of the production line.

5) If the application user finds some problems in a production line, they change the data catalogue in the M2M service platform to collect all data related to the production line and send the data catalogue to the targeted gateway.

7.1.7 High-level illustration

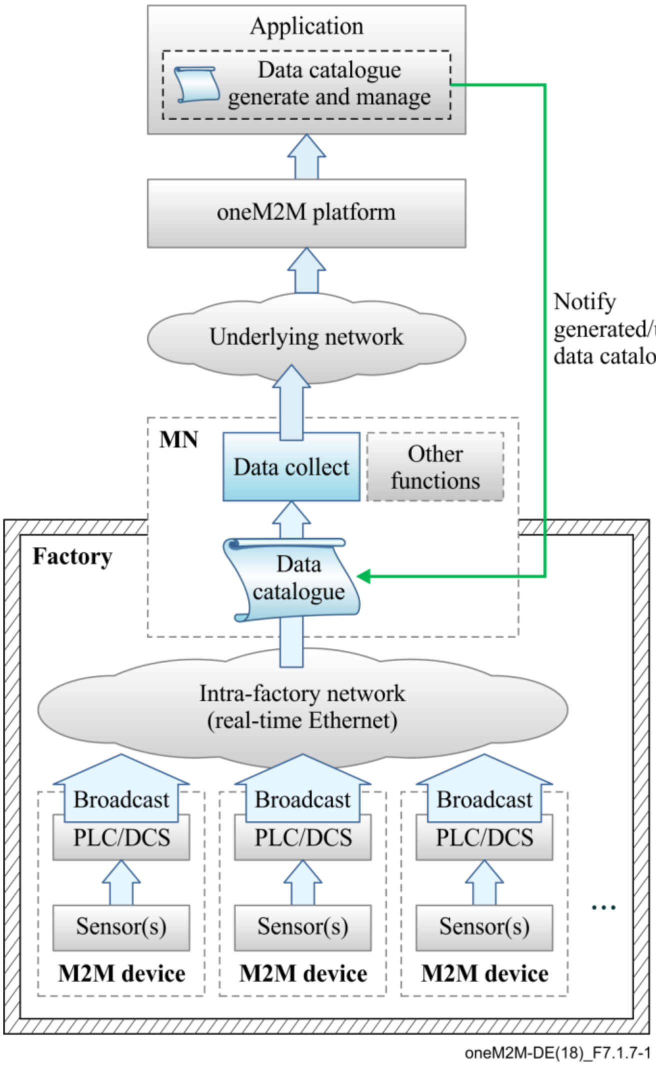


Figure 7.1.7-1 – High-level illustration of on-demand data collection for factories

7.1.8 Potential requirements

1) The gateway shall be able to collect data from the field area network (e.g., industrial bus systems) according to the data collection policy stored in the gateway (OSR-081 [b-ITU-T Y.4500.2]).

2) The data collection policy shall be manageable (e.g., configured, updated, deleted) by M2M applications on the M2M service platform (OSR-082 [b-ITU-T Y.4500.2]).

7.2 Integrity of data collection monitoring

7.2.1 Description

In factories, a lot of data is created from PLCs every second and data is utilized to monitor production lines. This data is available via industrial bus systems, e.g., real-time Ethernet.

This type of data is called time series data which is a sequence of data points, typically consisting of successive measurements made over a time interval.

To monitor remotely, data is gathered by the oneM2M service platform that needs to interface with such industrial bus systems via the M2M gateway (middle node (MN)).

When some of the data is lost due to various reasons, such as damage of production line, temporal network delay, continuous network capacity overload and so on, action will be required immediately for safety reasons. In addition, some considerations may be necessary, such as switching to a new network service with larger capacity, changing to the backup network or adjusting data collecting policy to address the original cause of data loss. Other considerations may be effective when remote monitoring application queries the oneM2M platform about the condition of network traffic, e.g., temporal delay, continuous capacity overflow, or connection failure.

Similar to the remote monitoring application, the MN in each factory receives the results of analysis or some commands, which could be lost due to for example, failure in analysis process, temporal network delay, or continuous network capacity overflow. The MN can detect the loss when the analysis results or the commands are in the form of time series data, or it can detect potential loss by monitoring the condition of network traffic. When temporal network delay or continuous network capacity overflow occurs, analysis results or commands may be lost. This loss also requires immediate decision and addressing at the root cause.

This use case proposes that the oneM2M system shall be able to provide the capability to collect, store time series data as well as monitor the integrity of the data.

Additionally, the oneM2M system shall be able to provide the capability to monitor the condition of network traffic.

7.2.2 Source

Not applicable.

7.2.3 Actors

– PLC: controls sensors and devices in a production line according to embedded programs. It also has interface to real-time Ethernet. It broadcasts data related to the production line to real-time Ethernet.

– MN: provides an interface from the real-time Ethernet to the oneM2M system. The gateway collects and stores time series data from real-time Ethernet then sends them to the oneM2M service platform. It also receives analysis results or commands from the oneM2M service platform. Furthermore, the gateway monitors the integrity of received analysis results or commands by monitoring the condition of the network. It can detect a loss when the analysis results or the commands are in the form of time series data, or it can detect potential loss with the help of monitoring the condition of the network traffic when temporal network delay or continuous network capacity overflow occurs.

– oneM2M service platform: stores data gathered from gateway(s), and provides data to applications.

– Application: an M2M application in the infrastructure domain monitors production lines by using collected data in the oneM2M service platform and sends analysis results or commands depending on the situation.

– Real-time Ethernet: a technology standardized in [b-IEC TC 65]. Ethernet is used at the physical layer; however, the upper protocol is designed for industry purposes. In this use case, broadcast protocol is assumed.

– Internet connection: the oneM2M service platform and gateway(s) are connected by the Internet physically.

7.2.4 Pre-conditions

– PLCs and the gateway are connected to the real-time Ethernet. PLCs broadcast data to the real-time Ethernet. The gateway is configured to pick up necessary data from the real-time Ethernet.

– On top of the Internet, a VPN connection is established between the oneM2M service platform and gateway(s).

7.2.5 Triggers

– The gateway starts to receive time series data.

7.2.6 Normal flow

1) The gateway picks up time series data which is broadcasted via real-time Ethernet. The gateway sends collected data to oneM2M service platform.

2) The oneM2M service platform receives data, then stores it and sends it to the application.

3) The application monitors the integrity of the time series data which is sent from the gateway. If data loss occurs, the application user will send a command for immediate control action. Then the application user will check the condition of the network traffic to determine which means are used to solve the data loss.

7.2.7 High-level illustration

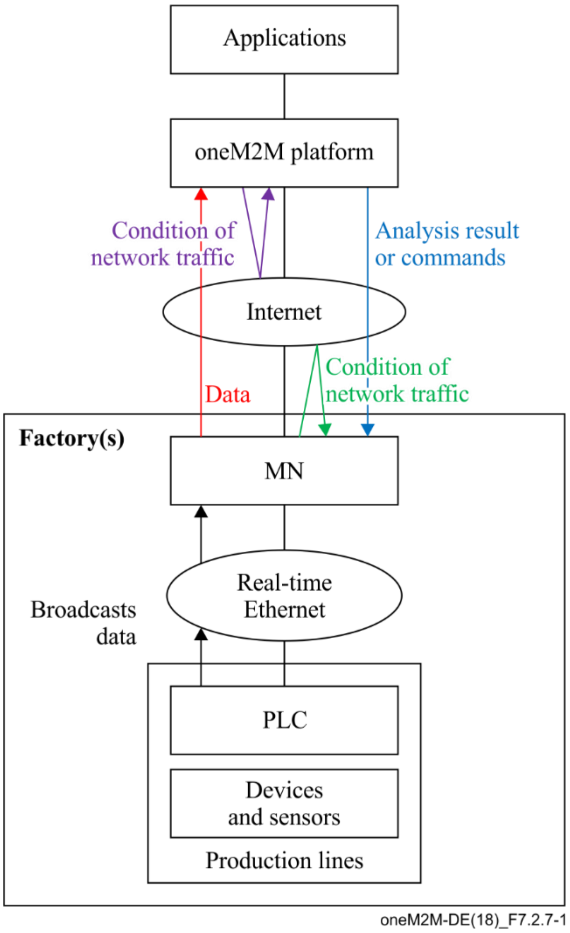


Figure 7.2.7-1 – High-level illustration of integrity of data collection monitoring

7.2.8 Potential requirements

1) The oneM2M system shall be able to collect and store time series data, as well as monitor the integrity of this data (OSR-075 and OSR-076 [b-ITU-T Y.4500.2]).

2) The oneM2M system shall be able to provide the capability to monitor the condition of the network traffic (OPR-007 and OPR-008 [b-ITU-T Y.4500.2]).

7.3 Data process for inter-factory manufacturing

7.3.1 Description

To achieve remote manufacturing, numerous sensors are placed at production lines in factories and a significant amount of data is generated for monitoring purposes. This data is broadcast via an intra-factory network (e.g., real-time Ethernet) through PLCs or distributed control systems (DCSs), etc. For monitoring product lines efficiently and effectively, MNs (which means the gateway) will selectively collect necessary data from an intra-factory network and then send this data to the oneM2M services platform for use by manufacturing control applications. The data collection policy (named data catalogue) utilized at the MNs is generated and managed by the application layer and may vary based on the specific monitoring purpose (e.g., collect only temperature data for the purpose of monitoring device temperature, or collect both humidity and gas data in order to monitor product quality). Data process functionality is also needed at the MNs in order to filter out error data or to summarize the percentage of data exceeding a threshold.

To respond rapidly to market changes, new factories may be needed to ensure sufficient productivity. Thus, inter-factory collaboration provides significant benefit and efficiency which can be used in establishing a new factory where the reference data catalogue can be used to establish a new environment.

7.3.2 Source

Not applicable.

7.3.3 Actors

- M2M device: sensors, controllers, etc., located in factories (e.g., located at product lines) which measure and generate data. The PLC/DCS control sensors in production lines according to embedded programs. Both PLC and DCS can broadcast data related to production lines to intra-factory networks.
- Intra-factory network: in this use case, real-time Ethernet is assumed. It is standardized in IEC TC 65 [b-IEC TC 65] for which Ethernet is used at physical layer, but upper protocol is designed for industrial purposes. Meanwhile, broadcast protocol is assumed and data is broadcast with unique identifier/parameter (e.g., device ID).
- oneM2M MN: the MN provides an interface from the intra-factory network to the oneM2M system. The MN collects data from the intra-factory network according to the data catalogue, which is the data collection policy. The MN may process collected data and send the data to the oneM2M services platform through the underlying network.
- oneM2M services platform: the oneM2M services platform stores data gathered from MNs, and provides data to applications.
- oneM2M application: a oneM2M application in the infrastructure domain that monitors production lines for remote manufacturing control by using collected data from the oneM2M services platform. For monitoring purposes, the M2M application defines and generates the data catalogue, then, the application provides the data catalogue to MNs. The M2M application also shares data catalogue in the MNs with other factories.

7.3.4 Pre-conditions

- The PLCs or DCSs control sensors in production lines according to embedded programs. Both PLC and DCS can broadcast data related to production lines into an intra-factory network.
- Real-time Ethernet is assumed as the intra-factory network. Broadcast protocol is assumed and real-time Ethernet data is broadcast with a unique identifier/parameter (e.g., device ID).

7.3.5 Triggers

– A remote manufacturing control application generates a data catalogue to collect data from product lines in factories.

7.3.6 Normal flow

1) The application generates the data collection policy into a data catalogue, or updates the data catalogue when the monitoring purpose has changed. The application then notifies the data catalogue to the MNs in a factory.
2) The application providing the data catalogue to the MNs may include the following condition:
   – in some inter-factory collaboration cases, the application also provides the data catalogue to the MNs in other collaborating factories, e.g., a newly built factory.
3) MNs start to selectively collect data from real-time Ethernet according to the data catalogue.
4) The oneM2M services platform receives, stores and provides the data to manufacturing control applications.
5) The application analyses collected data for remote manufacturing control.

7.3.7 Post-conditions

The application utilizes (e.g., monitors and analyses) the data collected according to the data catalogue.

7.3.8 High-level illustration

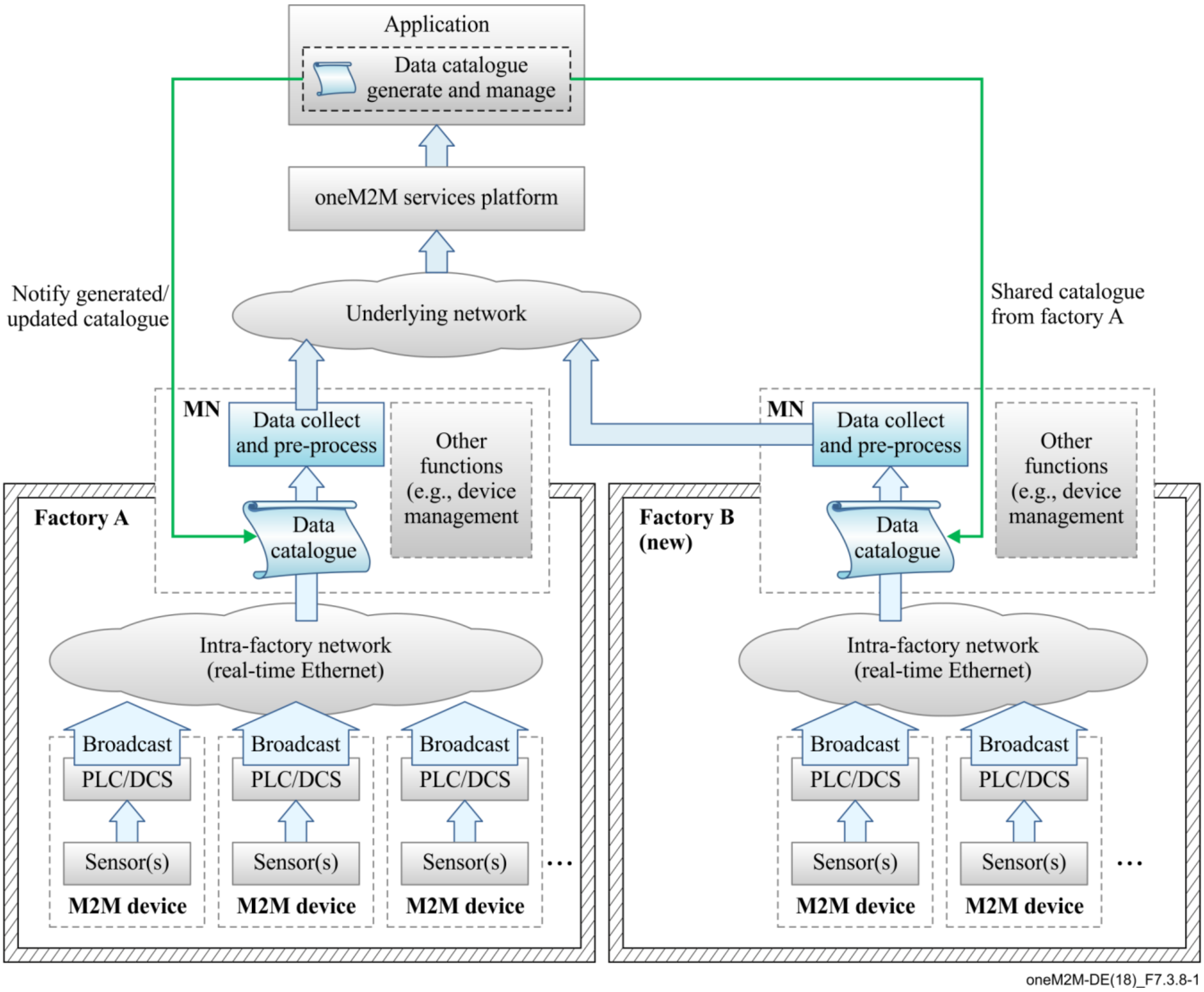

Application
Data catalogue generate and manage
oneM2M services platform
Notify generated/ updated catalogue
Underlying network
Shared catalogue from factory A
MN
Data collect and pre-process
Other functions (e.g., device management
Factory A
Data catalogue
MN
Data collect and pre-process
Other functions (e.g., device management
Factory B (new)
Data catalogue
Intra-factory network (real-time Ethernet)
Intra-factory network (real-time Ethernet)
Broadcast
PLC/DCS
Sensor(s)
M2M device
Broadcast
PLC/DCS
Sensor(s)
M2M device
Broadcast
PLC/DCS
Sensor(s)
M2M device
…
Broadcast
PLC/DCS
Sensor(s)
M2M device
Broadcast
PLC/DCS
Sensor(s)
M2M device
…
oneM2M-DE(18)_F7.3.8-1

Figure 7.3.8-1 – High-level illustration of data process for inter-factory manufacturing

7.3.9 Potential requirements

1) The oneM2M system shall be able to share data collection policies among different applications.

7.4 Aircraft construction and maintenance

7.4.1 Description

In aircraft construction, there are precise regulations that specify the type of screw and the amount of force that needs to be used to join specific parts. When it comes to passenger aircraft, there are thousands of such screws that have to be tightened and precisely documented. Joints on the wings naturally require a different amount of force than those on an aircraft window.

The tools are Wi-Fi-enabled and can identify their precise location on the shop floor. The position of the aircraft in the hangar is also fixed. With fixed coordinates and Wi-Fi connectivity, we know, for example, that a particular tool is located at the vertical stabilizer. Instructions that specify the force it should use to tighten screws can automatically be sent to the tool.

Although this use case focuses on aircraft construction and maintenance, the connected power tool technology is expected to be effective in more applications:

- Safety-critical work processes are closely monitored and analysed. Anomalies are automatically detected through the central processing, analysis, and visualization of production process data in near real time. Role-specific alerts can be triggered automatically.
- The power tool fleet manager has an exact overview of the power tool fleet status and utilization thanks to central access to process data. Organizational processes can be triggered automatically.
- Quality controls are automated and shifted to earlier stages of the production process. For example, hundreds of thousands of torque recordings are made available in their entirety for quality monitoring.
- Indoor geofencing alarms ensure that power tools are used according to regulations. Not all tools are allowed for all production and maintenance steps, e.g., in aircraft maintenance. As soon as power tools know their location, they can switch off when used in error.

7.4.2 Source

Not applicable.

7.4.3 Actors

- Power tools: people in a shop floor utilize them to tighten screws of the airplane. They are Wi-Fi-enabled, and can identify their precise location on the shop floor. They receive instruction for the amount of power needed to tighten screws.
- Indoor localization technology: the technology is utilized to identify the location of the power tools.
- oneM2M service platform: the oneM2M service platform receives location information for the power tools and sends it to the application. It also receives instructions for the amount of power needed to tighten the screws and sends it to each one of the power tools.
- Application: in this use case, the application is for aircraft construction and maintenance. It holds the position of the aircraft in the hangar and matches it to the location information of the power tools to calculate which power tool is utilized for which part of the aircraft. Then it gets instructions that specify the force a power tool should use to tighten screws by utilizing regulations that specify the kind of screw and the amount of force that has to be used to join specific parts. Finally, it sends the instructions to each one of the power tools.
- Underline network: in this use case, Wi-Fi is assumed.

7.4.4 Pre-conditions

- The application holds the position of the aircraft in the hanger and the regulations that specify the kind of screw and the amount of force that has to be used to join specific parts.
- Power tools are Wi-Fi-enabled.

7.4.5 Triggers

- A person on a shop floor starts to use a power tool.

7.4.6 Normal flow

1) The power tool identifies precise location information with the use of indoor localization technology and sends the information to the oneM2M platform.

2) The oneM2M service platform receives the location information and sends it to the application.

3) The application matches the position of the aircraft in the hanger to the location information of the power tool to calculate for which part of the aircraft the power tool is used.

4) The application gets instructions that specify the force the power tool should use to tighten screws by utilizing regulations that specify the kind of screw and the amount of force that needs to be used to join specific parts.

5) The application sends the instruction to the oneM2M platform.

6) The oneM2M platform receives the instruction and sends it to the power tool.

7) The power tool uses the specified amount of force to tighten a screw.

7.4.7 High-level illustration

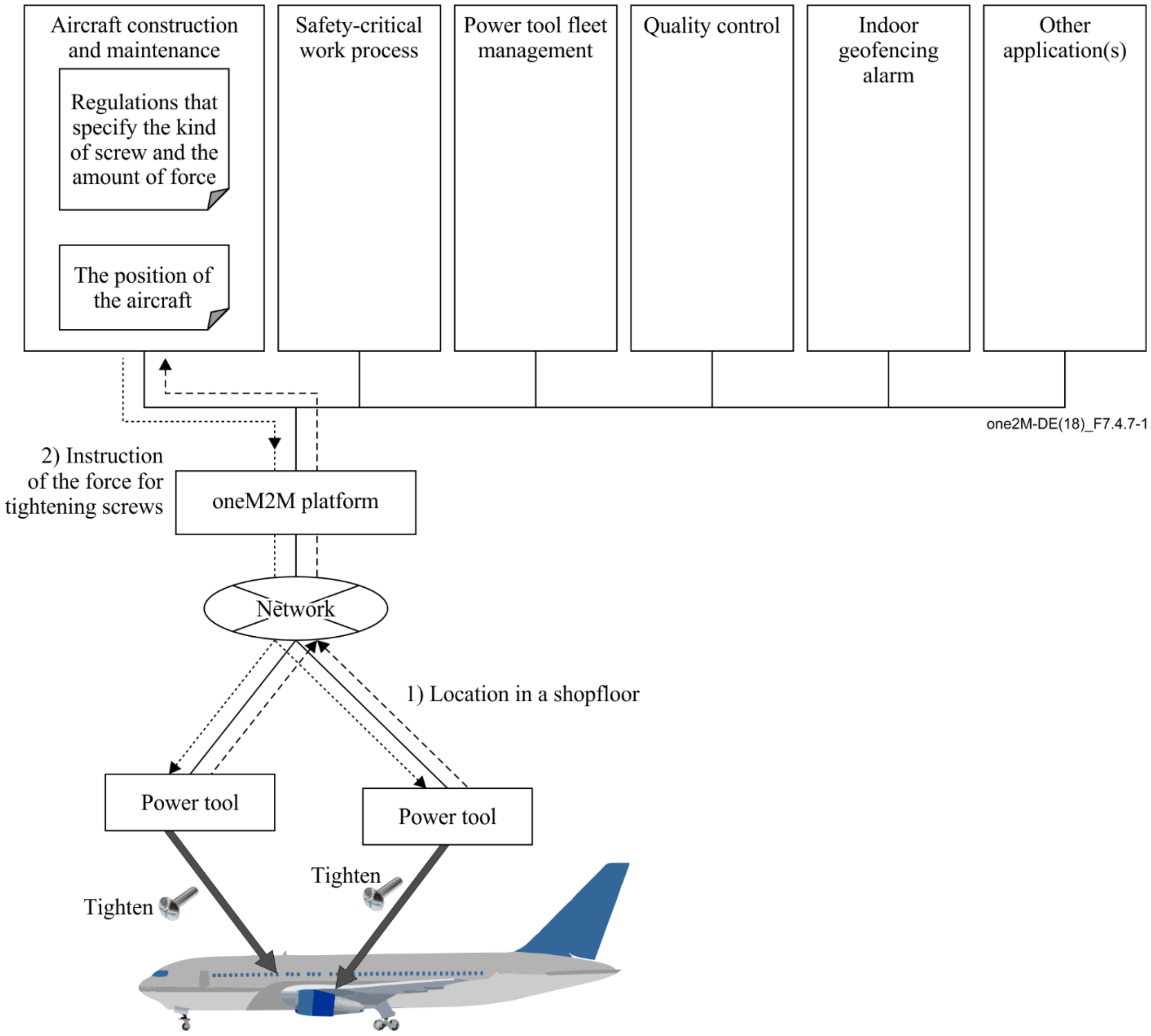

Aircraft construction and maintenance
Regulations that specify the kind of screw and the amount of force
The position of the aircraft
Safety-critical work process
Power tool fleet management
Quality control
Indoor geofencing alarm
Other application(s)
one2M-DE(18)_F7.4.7-1
2) Instruction of the force for tightening screws
oneM2M platform
Network
1) Location in a shopfloor
Power tool
Power tool
Tighten
Tighten

Figure 7.4.7-1 – High-level illustration of aircraft construction and maintenance

7.4.8 Potential requirements

1) The oneM2M system shall be able to support mechanisms for the M2M devices and/or gateways to report their geographical location information to M2M applications (OSR-047 [b-ITU-T Y.4500.2]).

2) The oneM2M system shall support the ability for single or multiple M2M applications to interact with a single or multiple M2M devices/gateways (application in the device/gateway) (OSR-009 [b-ITU-T Y.4500.2]).

7.5 Real-time data collection

7.5.1 Description

In automated production using information and communication technology, behaviours of devices are controlled according to sensor values. In order to achieve adequate control, real-time Ethernet, with which sensors and devices are connected through controllers, are required to provide real-time transmission and a high level of reliability. Real-time Ethernet is standardized in [b-IEC TC 65] and has characteristics such as the following:

- Enables real-time transmission by message priority control, according to the importance of the data (network re-configuration communication is first, time series data is second, controlling commands for devices are third, and information data is the forth priority). "Information data" contains video, audio data or objected sensor data. Higher priority data can be transmitted without interference from lower priority data. Priority of data depends not only on the kind of data but also on source node or destination node. For example, although controlling commands for devices shall not be delayed, log information for logging servers does not require strict real-time transmission. Therefore, the data have a different value of priority.
- When real-time Ethernet is introduced, size and frequency of high priority data transmission are designed to ensure that the total volume of sent data does not exceed the capacity. With this design, real-time transmission is ensured and time series data and controlling commands are received within pre-defined intervals.
- Utilizes a duplex local area network LAN, which consists of two physically independent network paths between end-nodes, or dual node to provide a high level of reliability.
- Achieves a high level of reliability on the basis of operations on ring-based topologies and following the reconfiguration process. Each node connects to the ring via two ports. In the normal state, two neighbouring nodes of the network become the blocking State and cut off the connections logically between them to prevent loops. Each node monitors the neighbouring segments including the blocking segment all the time. If any segment of the network is disconnected, it will be back to a normal state within a short recovery time by automatically moving the blocking segments to each end of the faulty part, thus isolating it.
- Broadcasts data to make each node have data, and it enables autonomous de-centred distributed process where each node can keep working even if network failure has occurred somewhere, and can achieve a high level of reliability.
- This use case describes the M2M gateway which is a oneM2M MN and sends data received from real-time Ethernet.
- The oneM2M MN shall be able to transmit data according to priority in preparation for temporal performance degradation of underlying network, and for temporal increase of the amount of information data. The oneM2M MN shall also be able to identify series of data (e.g., time series data) and to indicate the individual data belonging to this series. With this function, the MN transmits data of each series with the same priority, even though the amount of one series of data temporally increases.

7.5.2 Source

Not applicable.

7.5.3 Actors

- M2M device: sensors, controllers, etc., located in factories (e.g., located at product lines) which measure and generate data. PLC/DCS control sensors in production lines according to embedded programs.

- Real-time Ethernet: in this use case, real-time Ethernet with the characteristics mentioned above is assumed. It is standardized in IEC TC 65 [b-IEC TC 65].
- oneM2M MN: the MN provides an interface from the real-time Ethernet to the oneM2M system. The oneM2M MN is developed as a dual node (primary and secondary) to achieve a high-level of reliability.
- oneM2M services platform: the oneM2M services platform stores data gathered from MNs and provides data to applications.
- oneM2M application: a oneM2M application in the industrial domain.

7.5.4 Pre-conditions

- PLCs or DCSs send and receive data through real-time Ethernet.

7.5.5 Triggers

- The primary MN and secondary MN receive data which is sent from PLCs or DCSs.

7.5.6 Normal flow

1) The primary MN receives data and buffers it. If the buffer is overloaded then the data with the lowest priority is discarded.
2) The secondary MN also receives data and buffers it. If the buffer is overloaded then the data with the lowest priority is discarded.
3) The primary MN sends the buffered data with the highest priority to the oneM2M platform. If multiple data have the highest priority then the data flow of the least recent data transmitted is selected.
4) The secondary MN confirms the status of the primary MN and, as the primary MN is active, it stops the secondary MN from sending buffered data.
5) After a pre-defined time-interval, the primary or secondary MN receives further data from either the PLCs or DCS.

7.5.7 Alternative flow

1) When the secondary MN confirms that the primary MN is not working the secondary MN sends buffered data.
2) After a pre-defined time-interval, the secondary MN receives further data from either the PLCs or DCSs.

7.5.8 Post-conditions

When the process of sending data is not completed within the time interval, or the oneM2M platform does not receive the data, the primary or secondary MN sends the buffered data again.

7.5.9 High-level illustration

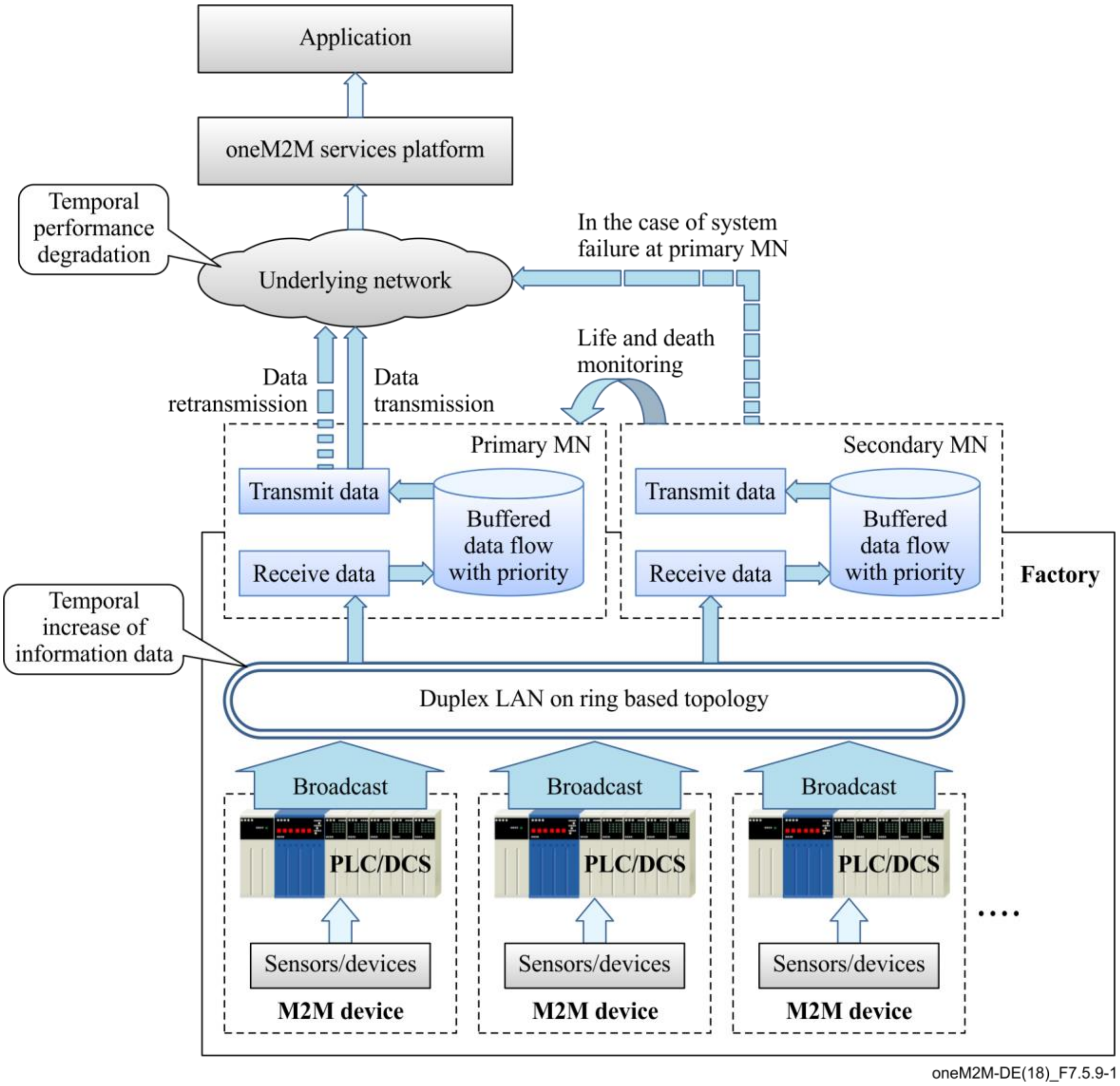


Figure 7.5.9-1 – High-level illustration of real-time data collection

7.5.10 Potential requirements

1) The oneM2M system shall be able to identify a series of data (e.g., time series data) and indicate the individual data belonging to this series (CMR-015 [b-ITU-T Y.4500.2]).

2) The oneM2M system shall be able to transmit data according to priority (CMR-003 [b-ITU-T Y.4500.2]).

7.6 Data encryption in industrial domain

7.6.1 Description

In smart factories data is essential for the execution of automation and efficient manufacturing. Data needs to be secured by measures such as authentication, authorization and encryption. As smart factories may connect to remote servers through multiple, partially external public networks, it cannot be assumed that these networks are secure. Encrypting data to avoid it being stolen in an external network and protecting the integrity of data to avoid it being modified is necessary for data security.

In the industrial domain, the requirements of encryption are different (e.g., sensitive data such as confidential commercial secrets need to be protected by strong encryption algorithms) and the capabilities of encryption/decryption vary for different devices (e.g., for some low-cost devices, encryption/decryption with a long key may cause heavy loads to their poor processing units). Therefore, each M2M application for smart factories needs proper encryption schemes to satisfy the various encryption requirements from application data while keeping the encryption/decryption loads acceptable to constrained devices.

M2M applications classify data into various levels based on the importance of the data and the capabilities of the devices. In the industrial domain the definition of sensitive data varies between products. For example, for manufactures of electric power grids, oil and gas, most data is sensitive since it includes confidential state secrets (required by governments to provide strong protection) while for makers of electronic products, information of product designs such as appearance or material is sensitive, since competitors stealing the product information will cause commercial damage. Based on common requirements from the industrial domain classification of application data is as follows, but not limit to, the categories below (dependent on M2M applications):

– **Sensitive data**: encryption strength is required (e.g., with longer key or electronic signature); sensitive data includes such as:
  - confidential commercial secrets (e.g., customer information, intellectual properties);
  - confidential data from infrastructure manufacturers (e.g., waveform data to diagnose, which is collected from devices in the power grid);
  - data for industrial control systems (including filed bus, supervisory control and data acquisition (SCADA), controllers as PLC or DCS;
  - keys transmitted for encryption algorithms.

– **Normal data**: encryption is recommended (optional) and proper applicable schemes should be adopted which are dependent on the capabilities of the devices; normal data includes such as:
  - status data for product line and device monitoring (e.g., device availability) in normal products manufacture which do not include any commercial secrets;
  - data for human resources monitoring and employee performance assessment (e.g., global positioning system (GPS) information of workers collected from carried mobile tablets).

When various levels of application data in industrial domain is encrypted/decrypted based on its associated encryption scheme, the essential data is secured and the loads caused by normal data encryption are acceptable for constrained devices.

7.6.2 Source

Not applicable.

7.6.3 Actors

– M2M device: machines, sensors, controllers, etc., located in factories which measure and generate data. PLC/DCS control machines and sensors in production lines according to embedded programs.

– Intra-factory network: in this use case, the intra-factory network is assumed to be managed by factory owners (different from the underlying network operated by external parties).

– oneM2M MN: the MN collects data from the intra-factory network and sends the data to the oneM2M services platform through underlying network.

– oneM2M services platform: support secure data transmission and mapping of application data levels to applicable encryption schemes.

– oneM2M application: classify data based on sensitivity.

7.6.4 Pre-conditions

– The oneM2M services platform and security common services entity (CSE) inside M2M devices/MNs support various application data levels and mapping these levels to applicable encryption schemes.

7.6.5 Normal flow

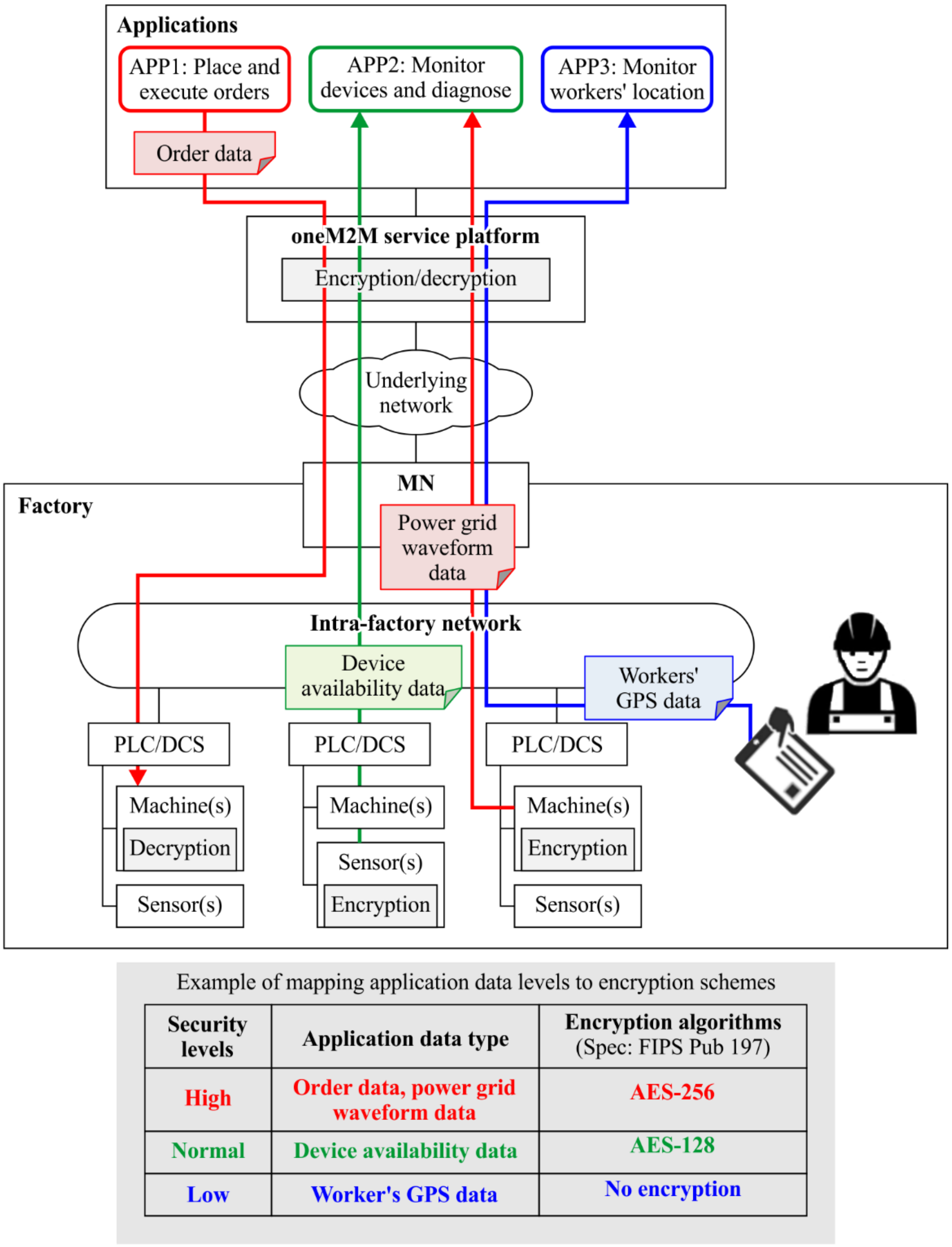
The M2M application classifies data into various levels (e.g., into sensitive data and normal data).

1) The oneM2M services platform or security CSE inside M2M devices/MNs maps these levels to applicable encryption schemes.
2) Each pair of transmitter and receiver performs preparation procedure (e.g., share symmetric key and store it in respective storage inside nodes) before sending encrypted data.
3) The transmitter (e.g., a server located at one end of the application) prepares the data for sending; encrypts the data based on its associated security level (with respective encryption algorithm, key length, etc.).
4) The encrypted data is sent to the receiver through intra-factory network and public underlying networks.
5) The receiver (e.g., a machine located at the other end of the application) receives the data from transmitter and decrypts the data.

7.6.6 Post-conditions

The M2M application utilizes the decrypted data for smart manufacturing, such as executing orders, monitoring devices and diagnosis, monitoring workers' location, etc.

7.6.7 High-level illustration



Example of mapping application data levels to encryption schemes

Security levels	Application data type	Encryption algorithms (Spec: FIPS Pub 197)
High	Order data, power grid waveform data	AES-256
Normal	Device availability data	AES-128
Low	Worker's GPS data	No encryption

oneM2M-DE(18)_F7.6.7-1

Figure 7.6.7-1 – High-level illustration of data encryption in industrial domain

7.6.8 Potential requirements

1) The oneM2M system shall support classification of application data by oneM2M applications into various security levels that are specified by oneM2M and support the mapping of these levels to applicable security capabilities (SER-045 [b-ITU-T Y.4500.2]).

7.7 Qos/QoI monitoring in industrial domain

7.7.1 Description

In factories, a lot of data are generated from M2M devices (e.g., machines and program logic controllers) and the data are delivered to the M2M gateway via industrial bus systems, e.g., real-time Ethernet. In addition, factory management applications can get factory status information through the oneM2M service platform (infrastructure node (IN)) which gathers data from M2M gateways located in each factory domain.

In local industrial communications data packet transmission between M2M gateway and M2M devices has real-time transmission characteristic delivered over an Ethernet-based communication system. However, to enable remote mechanisms (remote supervisory, operation, service), wide area networks is composed of broad and heterogeneous communication technologies, e.g., digital wireless telecommunication systems (global system for mobile communication (GSM)-based, universal mobile telecommunications system (UMTS)-based), digital wired telecommunication systems (integrated services digital network (ISDN), digital subscriber line (DSL)).

In this environment the M2M gateway can use various telecommunication systems to send and receive data packets from the oneM2M service platform. In addition, according to industrial application service types, it requires hard real-time data delivery, soft real-time data delivery or real-time not requiring data delivery when it comes to communication between the M2M gateway and the oneM2M service platform.

If quality of service (QoS) required from the application cannot be guaranteed, this situation limits service scenarios in industrial domains. In order to prevent this situation, the M2M gateway can decrease the volume of data needed to send the oneM2M platform via data processing based on data catalogue. At the same time, if the M2M gateway can monitor network environments, it can dynamically choose the network type or network provider who guarantees the required QoS.

In addition, to satisfy QoS, real-time data generated from M2M devices can be pre-processed/filtered, based on the data catalogue. In this situation, post-processed data is to include a kind of quality of information (QoI) and if QoI monitored and delivered to the oneM2M service platform, this information can be used for further data processing in the oneM2M platform.

This use case proposes that the oneM2M system offers QoS/QoI Monitoring capabilities, which includes data accuracy, data age, cost, communication, encryption, etc.

7.7.2 Source

Not applicable.

7.7.3 Actors

– M2M devices: sensors, controllers, etc., located in factories (e.g., located at product lines) which measure and generate data. PLC/DCS control sensors in production lines according to embedded programs.

– Real-time Ethernet: a technology standardized in [b-IEC TC 65] for use in industrial control system.

– MN gateway (MN): provides an interface from the real-time Ethernet to the oneM2M system. The gateway collects data from M2M devices which are connected via real-time Ethernet communication technology. The gateway can conduct data pre-processing/filtering based on the data catalog delivered from the oneM2M service platform.

– oneM2M service platform (IN): acts as a oneM2M IN. It communicates with MNs in the remote industrial domains and gathers the data from the MN. The data in the oneM2M service platform can be delivered to the applications, e.g., factory monitoring application.

– Applications: an M2M application in the application service provider domain. It monitors production lines and sends analyzed results or alert messages to the factory administrator.

7.7.4 Pre-conditions

– Devices (e.g., PLC, machines) and the gateway are connected to real-time Ethernet. PLCs broadcast data to real-time Ethernet.

- The gateway can have a capability of various network hardware interfaces (e.g., GSM-based, UMTS-based, ISDN, DSL) and can also use various network service providers who guarantee the required QoS level.

7.7.5 Triggers

- The application initiates service which require QoS/QoI requirement (e.g., response time, data freshness).
- The application sends the QoS/QoI requirement to the oneM2M platform.

7.7.6 Normal flow

1) The oneM2M service platform requests QoS/QoI monitoring data from the MN and based on this information, the oneM2M service platform negotiates the supported QoS/QoI parameter with the application.
2) To enable end-to-end services, the oneM2M service platform sends the QoS requirement to the MN. Based on the QoS requirement, the MN dynamically chooses the network type and network service provider who guarantees the required QoS level.
3) In the MN, the QoS/QoI monitoring function can annotate data with quality information.
4) After receiving data from the MN, the oneM2M service platform can further process the data referring to the quality attributes.

7.7.7 Alternative flow

None.

7.7.8 Post-conditions

None.

7.7.9 High-level illustration

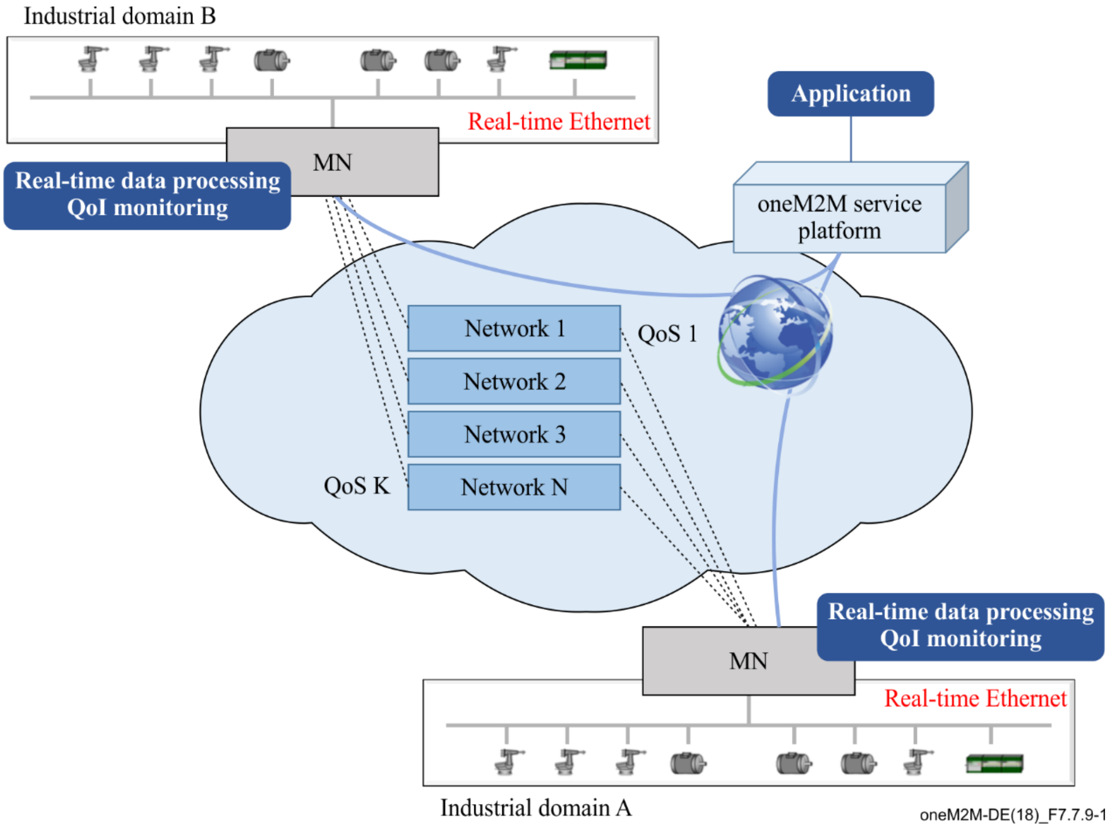


Figure 7.7.9-1 – High-level illustration of QoS/QoI monitoring in industrial domain

7.7.10 Potential requirements

1) The oneM2M system shall support the inclusion of M2M Application's QoS preference in service requests to underlying networks (OSR-038 [b-ITU-T Y.4500.2]).

2) The oneM2M system shall provide the capability to monitor and describe data streams with associated attributes, e.g., data freshness, accuracy, sampling rate, data integrity (OSR-092 [b-ITU-T Y.4500.2]).

8 Overview of potential requirements

Potential requirements from all industrial use cases collected in this technical report are summarized as follows:

1) The gateway shall be able to collect data from the field area network (e.g., industrial bus systems) according to the data collection policy stored in the gateway. (OSR-081 [b-ITU-T Y.4500.2]).

NOTE 1 –This requirement addresses the use case 7.1 "An industrial use case for on-demand data collection for factories".

2) The data collection policy shall be manageable (e.g., configured, updated, deleted) by M2M applications on the M2M service platform. (OSR-082 [b-ITU-T Y.4500.2]).

NOTE 2 –This requirement addresses the use case 7.1.

3) The oneM2M system shall be able to collect and store time series data, as well as monitor the integrity of this data (OSR-075 and OSR-076 [b-ITU-T Y.4500.2]).

NOTE 3 –This requirement addresses the use case 7.2 "Integrity of data collection monitoring".

4) The oneM2M system shall be able to provide the capability to monitor the condition of the network traffic. (OPR-007 and OPR-008 [b-ITU-T Y.4500.2]).

NOTE 4 –This requirement addresses the use case 7.2.

5) The oneM2M system shall be able to share data collection policies among different applications (OSR-097 [b-ITU-T Y.4500.2]).

NOTE 5 –This requirement addresses the use case 7.3 "Data process for inter-factory manufacturing".

6) The oneM2M system shall be able to support mechanism for the M2M devices and/or gateways to report their geographical location information to M2M applications (OSR-047 [b-ITU-T Y.4500.2]).

NOTE 6 –This requirement addresses the use case 7.4 "Aircraft construction and maintenance".

7) The oneM2M system shall support the ability for single or multiple M2M applications to interact with a single or multiple M2M devices/gateways (application in the device/gateway) (OSR-009 [b-ITU-T Y.4500.2]).

NOTE 7 –This requirement addresses the use case 7.4.

8) The oneM2M system shall be able to identify a series of data (e.g., time series data) and indicate individual data belong to this series (CMR-015 [b-ITU-T Y.4500.2]).

NOTE 8 – This requirement addresses the use case 7.5 "Real-time data collection".

9) The oneM2M system shall be able to transmit data according to priority.

NOTE 9 – This requirement addresses the use case 7.5 and it is included in/supported by CMR-003 [b-ITU-T Y.4500.2].

10) The oneM2M system shall support classification of application data by oneM2M applications into various security levels that are specified by oneM2M and support the mapping of these levels to applicable security capabilities (SER-045 [b-ITU-T Y.4500.2]).

NOTE 10 – This requirement addresses the use case 7.6 "Data encryption in industrial domain".

11) The oneM2M system shall support the inclusion of M2M application's QoS preference in service requests to underlying networks (OSR-038 [b-ITU-T Y.4500.2]).

NOTE 11 –This requirement addresses the use case 7.7 "QoS/QoI monitoring in industrial domain".

12) The oneM2M system shall provide the capability for monitoring and describing data streams with associated attributes e.g., data freshness, accuracy, sampling rate, data integrity (OSR-092 [b-ITU-T Y.4500.2]).

NOTE 12 – This requirement addresses the use case 7.7.

9 High-level architecture

9.1 Introduction

The seven use cases in the industrial domain discussed in the present document are listed in Table 9.1-1.

Table 9.1-1 – Use cases in the industrial domain

Use case No.	Title	Description
1	An industrial use case for on-demand data collection for factories	See clause 7.1
2	Integrity of data collection monitoring	See clause 7.2
3	Data process for inter-factory manufacturing	See clause 7.3
4	Aircraft construction and maintenance	See clause 7.4
5	Real-time data collection	See clause 7.5
6	Data encryption in industrial domain	See clause 7.6
7	QoS/QoI monitoring in industrial domain	See clause 7.7

The deployments which support these use cases require the use of M2M devices which use broadcasting mode through the PLC or DCS. The following clauses provide the high level oneM2M architecture mapping for these deployments.

9.2 Deployment mapping using IPE

Table 9.2-1 lists the mapping relationship between the actors in industrial domain and the nodes in oneM2M domain. Those devices under the M2M gateway are non-oneM2M devices which can be mapped into the M2M area network. In this case, the M2M gateway which is mapped to the MN shall implement the interworking proxy application entity (IPE).

Table 9.2-1 – Mapping relationship 1

Use case No.	Actors in the use case	oneM2M node
1, 2, 3, 4, 5, 6, 7	Application and M2M service platform	IN
1, 2, 3, 5, 6, 7	M2M gateway	MN
	M2M devices (PLC/DCS, sensors) and intra-factory network	Non-oneM2M device in M2M area network
4	Wireless fidelity (WiFi) gateway	MN
	Power tools and underlying network (WiFi)	Non-oneM2M device in M2M area network

Figure 9.2-1 illustrates deployment mapping using IPE.

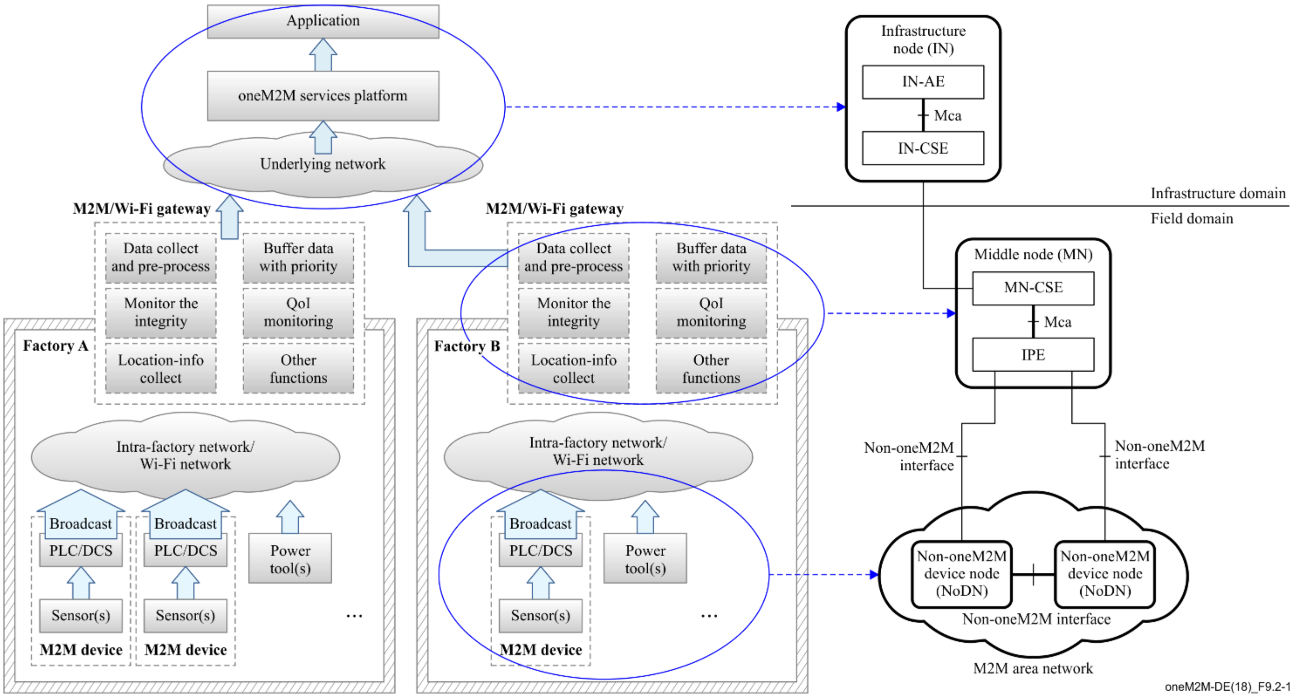


Figure 9.2-1 – Deployment mapping using IPE

9.3 Deployment mapping using peer-to-peer communication

Table 9.3-1 lists the mapping relationship between actors in industrial domain and nodes in the oneM2M domain. In this case, all of the actors are assumed to be oneM2M compliant nodes.

Table 9.3-1 – Mapping relationship 2

Use case No.	Actors in the use case	oneM2M node
1, 2, 3, 4, 5, 6, 7	Application and M2M service platform	IN
1, 2, 3, 5, 6, 7	M2M gateway	MN
	M2M devices (PLC/DCS, sensors)	ADN/ASN
4	WiFi gateway	MN
	Power tools	ADN/ASN

Figure 9.3-1 illustrates deployment mapping for peer-to-peer communication. In the existing factory production line, M2M devices with PLC/DCS will support peer-to-peer communication, which means data can be exchanged among nodes directly. Peer-to-peer communication between CSEs of different ASNs is currently not supported by oneM2M architecture deployment (see clause 6.1 of [b-ITU-T Y.4500.1]).

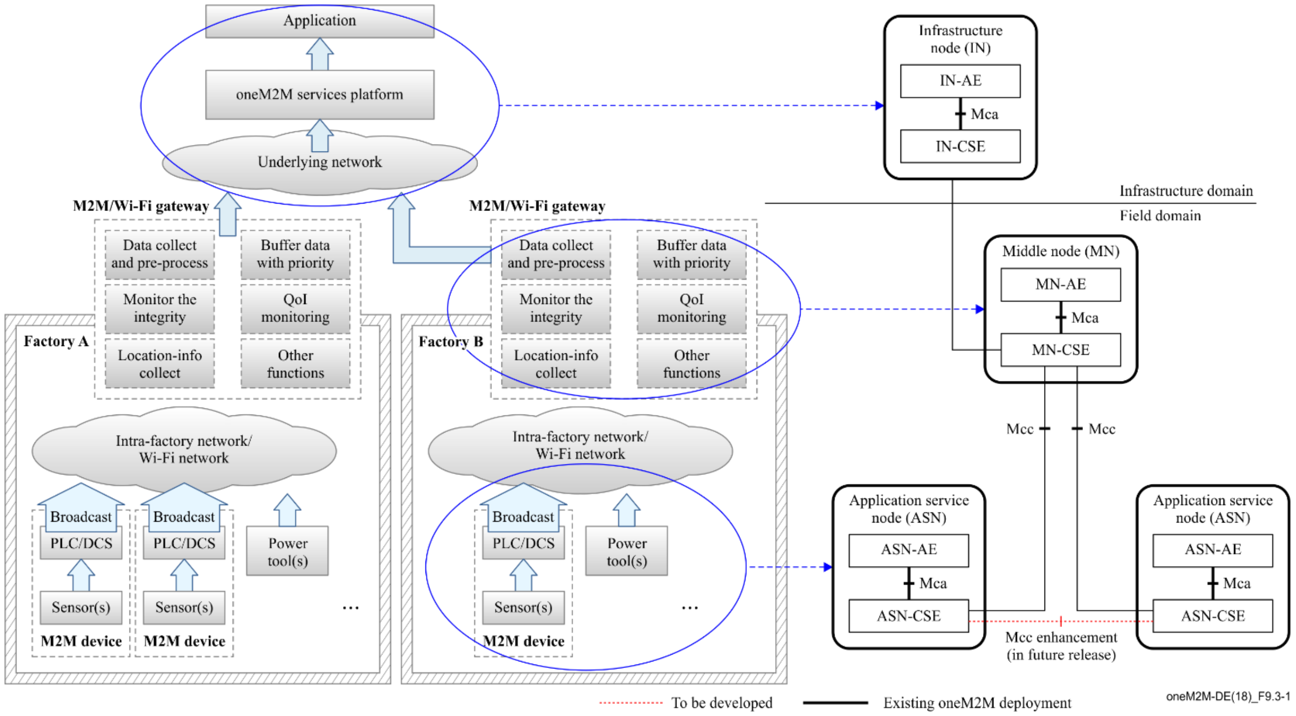


Figure 9.3-1 – Deployment mapping using peer-to-peer communication

9.4 Conclusion

Since M2M devices with PLC/DCS need to support peer-to-peer communication, the existing reference enhancement between CSEs of different ASNs shall be considered in the future oneM2M releases.

10 Security analysis

10.1 Introduction

The use cases in clause 7 have described three categories of smart manufacturing scenarios which include "inside the factory", "collaborated factories" and "factories connected to remote server/centre". In these scenarios, devices are deployed in multiple networks which construct the factories (e.g., enterprise network, control system network, supervisory network, field network) and these networks are connected by various types of boundary nodes. The factories also increase connectivity to the outside for remote facility, data centre or remote smart manufacturing services. Therefore, cyber attackers have plenty of opportunities to access a manufacturer's trade secrets and sensitive production data, or attack against important industrial systems (e.g., control system).

Compared to other vertical domains, the industrial domain has some specific characteristics which may impact the security requirements, such as:

- **Real-time**: this characteristic is also known as "timely response to events" in industrial domain. For some devices inside the factory, their tasks need to be processed within a certain period of time, especially the sensing and actuating devices in industrial automation and control system (IACS).
- **Various protection motivations and different device capabilities**: in different industrial systems, the motivation for protection may vary (from normal to high) depending on the importance of asset or data. The capabilities of device may also vary, which means they may provide different resources or complexity of protection mechanisms. Therefore, a mixed mode of protection solutions is usually required in practical industrial scenarios.

– **Co-existence of multiple networks**: in the above three categories of industrial scenarios, industrial Ethernet/fieldbus and enterprise wireless local area network (WLAN) are deployed for inside factory scenarios; public Internet is the necessary network for connecting one factory to another or to remote center.

Considering these characteristics in the industrial domain, some international standard organizations have made industrial security standards. For example, IEC have made the 62443 series as the standards for IACS and defined four security levels (SLs) based on various security motivations and device capabilities [b-IEC 62443]. (SL0 is defined as no security requirements; from SL1 to SL4, the motivation for protection, consumed resources and complexity of protection mechanisms increase; for SL3 and SL4, IACS specific skills are possibly utilized.)

The foundational security requirements in the industrial domain are detailed in clauses 10.2 to 10.6.

10.2 Identification and authentication

All users (humans, devices and software processes) need to be identified and authenticated before allowing them to access certain systems for operating devices or acquiring data.

The identification and authentication function defined in [b-oneM2M TS-0003] is in charge of identification and mutual authentication of CSEs and AEs. If all the actors in the industrial domain are oneM2M compliant nodes, defined mechanisms in [b-oneM2M TS-0003] are possibly used to identify and authenticate any access request. Methods include using passwords, tokens or location (physical or logical), and are not limited to other feasible methods.

10.3 Use control

10.3.1 Introduction

Once the user is identified and authenticated, the industrial domain has to restrict the allowed actions to the authorized use of the targeted devices/resources. If a requested operation is covered by the permissions accorded by the access control policies (ACPs), the operation is executed, otherwise it is rejected.

The requirement of use control also includes session lock which is required to prevent access after a period of inactivity and the limitation of concurrent sessions which is required for denial of service (DoS) prevention.

10.3.2 Authorization

Similarly, with the definition of authorization function in [b-oneM2M TS-0003], services and data access in the industrial domain are authorized to authenticated entities/users according to provisioned ACPs and assigned roles. Identity-based ACP (access control list in [b-oneM2M TS-0003]), role-based access control (RBAC) in [b-oneM2M TS-0003] and rule-based AC are the common authorization mechanisms.

An additional requirement of supervisor override exists in the industrial domain. While automated common authorization mechanisms are sufficient in most scenarios, in the event of emergencies or other serious events, a manual override of automated authorization mechanism is needed, especially in control systems.

10.3.3 Session lock and concurrent session control

Session lock: The industrial control system may require the prevention of further access by initiating a session lock after a configurable time period of inactivity or by manual initiation (although the previous action has been authorized according to the ACP).

Concurrent session control: The industrial control system may require the limitation of the number of concurrent sessions per interface, for any given user to a configurable number of sessions. A resource starvation DoS might occur if a limitation is not imposed.

10.4 Data confidentiality

10.4.1 Introduction

The sensitivity and the importance of data in industrial domain may be diverse (see clause 10.6 for classification of application data in industrial domain). In the case of some control system-generated data, for example, (whether at-rest or in transit), this kind of data may be of a confidential or sensitive nature, therefore data storage and communication channels should be secure.

Based on the various protection motivations (depending on the sensitivity of the data) and the different device capabilities, different industries may require different levels of encryption strengths for each data category. The use of cryptography is required to match the value of data, the time period during which the data is confidential and industrial constraints. The industrial control system should utilize encryption and hash algorithms such as advanced encryption standard (AES), secure hash algorithm (SHA) series, and key size based on generally accepted practices and recommendations [b-NIST (SP)800-57].

In addition to the common cryptography mechanisms, there are other security solutions needed to meet specific industrial requirements which are detailed in clauses 10.4.2 and 10.4.3.

10.4.2 Light-weight encryption

Standard applications of data encryption algorithms do not always meet real-time processing requirements. Within a packet, data at different positions may have various levels of importance, and consequently different security level needs. Therefore, using a low security level of data encryption at specific positions may be used to avoid unnecessary processing overhead. For example, the best practice might be for a light-weight encryption procedure to be used for efficient and highly automated devices used in control systems.

A simple encryption procedure is provided by ITU-T [b-ITU-T X.1362], which significantly reduces the consumed time for encryption and meanwhile protects data confidentiality and integrity. Such light-weight encryption algorithms shall be considered for protecting industrial data, especially for low-cost devices.

10.4.3 Session-based encryption

To exchange very sensitive data (such as manufacturer's trade secrets and sensitive production data), a session key shall be used for secure sessions. The defined security association establishment procedure in [b-oneM2M TS-0003] results in a transport layer security (TLS) or datagram transport layer security (DTLS) session which protects the data via a secure session establishment. Such a secure connection shall be established to protect the confidential and sensitive data in industrial domain.

10.5 System integrity

10.5.1 Introduction

The capability to protect system integrity is required in the industrial domain, especially for the protection of communication integrity and session integrity.

10.5.2 Communication integrity

Many attacks are based on the manipulation of data in transmission. Manipulation in the context of a control system could include the change of measurement values communicated from a sensing device to a receiver, or the alteration of command parameters sent from a control application to an actuating device.

The message integrity code (MIC) is defined in [b-oneM2M TS-0003] to provide integrity protection for the exchange of messages across reference points. Such cryptographic mechanisms shall be provided to protect the integrity of data in transmission for the industrial domain.

10.5.3 Session integrity

Besides protecting data in transmission, the integrity of sessions also needs protection in the industrial domain. This integrity focuses on the protection at the 'session versus packet' level (such as prevent session hijacking, insertion of false information into a session). The intent is to establish confidence at each end of a communication session in the ongoing identity of the other party. Use of session integrity may lead to significant overhead and therefore the use should only be considered when real-time communication is required.

10.6 Restricted data flow

Some important industrial systems (e.g., control systems) may be disconnected from an enterprise network or public network using unidirectional gateways, stateful firewalls and demilitarized zones (DMZs) to manage the flow of data.

As the co-existent multiple networks in the industrial domain are connected by boundary nodes, these boundary nodes such as gateways, proxies, firewalls shall provide proper capabilities to restrict or prohibit network access in accordance with provisioned security policies and an assessment of risk.

10.7 Conclusion

Considering the specific security requirements and the best practices of the industrial domain enhancement of security solutions shall be considered in the future oneM2M releases.

11 OPC-UA interworking

11.1 Introduction of OPC-UA

11.1.1 Background

This clause introduces the background of open platform communications-unified architecture (OPC-UA). The earlier adoption of OPC classic for industrial automation and enhancement to OPC-UA will be described.

In early stage of industry automation, there are no standards for interfacing with devices/tools/applications in industry. Thus, each vendor must develop their own proprietary servers. It is costly, inefficient and risky. There are numerous incompatible protocols. Configuration and maintenance is very complex. Island of automation is everywhere and not connected. Thus, there is no interoperability with other system. OPC (object linking and embedding (OLE) for process control) standard provides a standard interface for applications to communicate and exchange data and objects, and vendors only need to develop to one standard OPC interface. This will reduce cost, protect investment, and increase productivity.

The OPC standard has developed several specifications. [b-OPC Classic] OPC data access (DA) is the first one, then OPC A&E (alarms and events), OPC historical data access (HDA), etc. OPC DA has been widely adopted but has some disadvantages, such as no integration between different specifications: DA, HDA, A&E; poor connection reliability; poor security; not firewall friendly, etc.

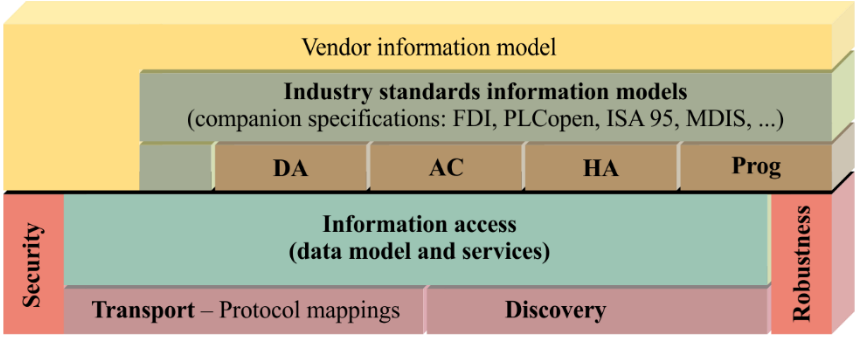
Then the latest OPC UA has been developed and released in 2008. [b-OPC UA website] OPC UA is applicable to manufacturing software in application areas such as field devices, control systems, manufacturing execution systems (MES) and enterprise resource planning (ERP) systems. These systems are intended to exchange information and to use command and control for industrial processes. OPC UA defines a common infrastructure model to facilitate this information exchange.

OPC UA is easy for configuration and maintenance. It has increased visibility, reliability, security and performance. It has platform neutrality feature and it's compatible with legacy products. Now OPC UA is very widely used in industry, and become a very important part of Industry 4.0.

11.1.2 Key features

This clause introduces the key features of OPC-UA, such as the features of the communication model, provided services and resource representation.

Figure 11.1.2-1 shows the layering architecture of OPC UA. OPC UA utilizes a client/server-based communication model. At the bottom layer, discovery supports to find the availability of OPC servers on local PCs and/or networks; transport includes various protocol mappings for different requirements such as speed or firewall-friendly. Information access is achieved based on structured data models and provided services by OPC UA. Additionally, OPC UA consider security and robustness for both access procedure and during transportation/session. At the top layer, the functionalities of successful OPC classic technologies (e.g., DA) are integrated; information modeling turns data into information, supports companion industry standards information models (e.g., ISA 95) and vendor specific extensibility.


Vendor information model
Industry standards information models
(companion specifications: FDI, PLCopen, ISA 95, MDIS, ...)
DA
AC
HA
Prog
Security
Information access
(data model and services)
Robustness
Transport – Protocol mappings
Discovery

oneM2M-DE(18)_F11.1.2-1

Figure 11.1.2-1 – OPC UA layering architecture

The following are key features of OPC UA:

Functional equivalence

Building on the success of OPC classic, such as DA, AC (alarms and conditions), HA (same as HDA), program (Prog) which are integrated into OPC UA, OPC UA is functionally equivalent to OPC classic, but more capable. Integration between OPC UA products and OPC classic products is easily accomplished.

Platform independence

OPC UA functions on any of the following and more:

– Hardware platforms: traditional PC hardware, cloud-based servers, PLCs, micro-controllers (advanced reduced instruction set computing (RISC) machine (ARM, etc.)

– Operating systems: Microsoft Windows, Apple OSX, Android, or any distribution of Linux, etc.

OPC UA provides the necessary infrastructure for interoperability across the enterprise, from machine-to-machine, machine-to-enterprise and everything in-between. Asset management system, production control system, purchasing system, human machine interface (HMI) visualization system, etc., can be integrated together by OPC UA.

Information modeling

The standard information model is developed and deployed to address industry domains specifics. It describes standardised *nodes* of a *server*'s *AddressSpace*. These *nodes* are standardised types as well as standardised instances used for diagnostics or as entry points to server-specific *nodes*. Thus, the information model defines the *AddressSpace* of an empty OPC UA server.

Support companion standards such as ISA-95 common object model, since in current automation systems, devices from many different manufacturers have to be integrated. Data/information from the plant floor is abstracted through information models, and flow up to enterprise systems.

Security

OPC UA clients present credentials to OPC UA servers (x509 certs on both sides). OPC UA servers require authentication and authorization, optional message signing and encryption.

Robustness/reliability designed and built in

OPC UA overcomes the inherent problems associated with failed communications and failed clients and servers. Sequence numbers, keep-alive, resynchronizing, and support for redundancy are highlighted.

Service sets

The interface between OPC UA clients and servers is defined as a set of services. These services are organized into logical groupings called service sets. The following **service sets** are supported by OPC UA [b-OPC UA Part 4]:

– **Discovery**: discover servers (FindServers, GetEndpoints, RegisterServer)

– **SecureChannel**: open/close secure communication (lower level – protocol dependent)

– **Session**: open/close session

– **Attribute**: read/write data (including history)

– **Subscription**: subscribe to data (receive data)

– **MonitoredItem**: subscribe to data (specifying which data to subscribe to)

– **View**: browsing allows clients to navigate up and down the hierarchy through the AddressSpace

– **Query**: querying to select nodes based on certain filter criteria

– **NodeManagement**: add/delete nodes and references

– **Method**: method calls

11.1.3 Protocol stack

This clause introduces the protocol stack of OPC-UA for which establishment of TCP connection is required as the foundation.

OPC UA stack includes the following layer (see Figure 11.1.3-1):

– Message transport layer

– Message security layer (security channel layer)

– Message encoding layer (serialization layer)

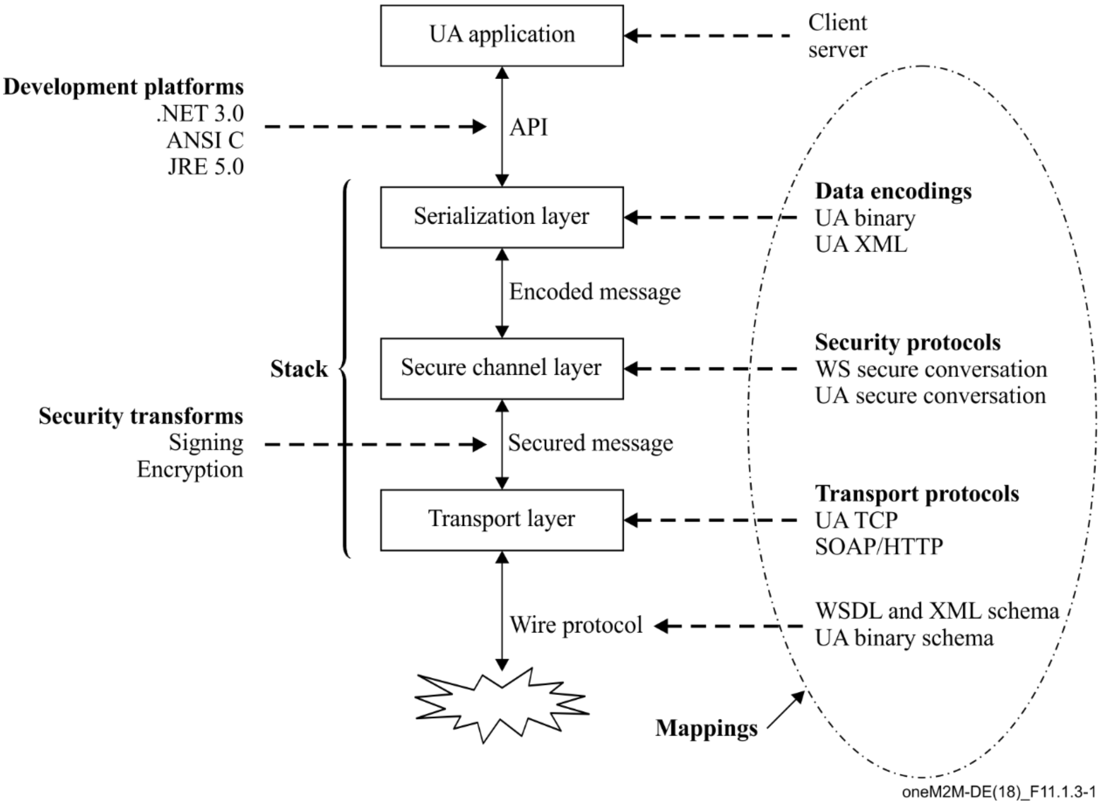


Figure 11.1.3-1 – The OPC UA stack overview

OPC UA stack profiles define standard combinations (see Figure 11.1.3-2):

– XML web services (XML or UA binary)

– Native binary (UA TCP, UA secure conversation and UA binary)

– Native with SOAP/HTTP

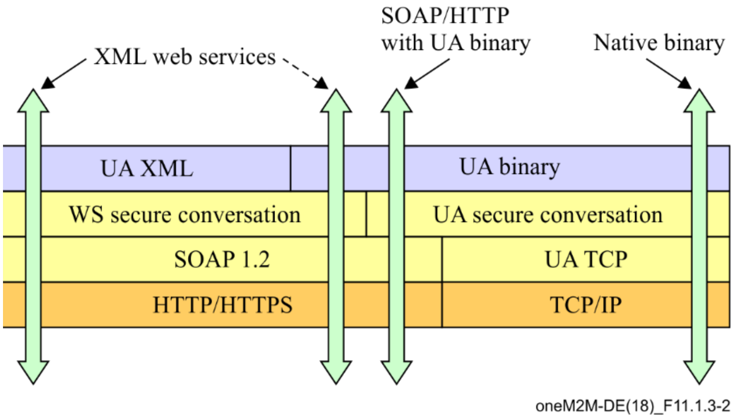


Figure 11.1.3-2 – The OPC UA stack profile

Advantages and disadvantages of different profiles are summarized as below:

– XML web services stack profile

- excellent tool support, firewall friendly, CPU and bandwidth intensive;
- recommended applications: enterprise integration, MES, other general business/information applications.

– Native binary stack profile

  - best performance; limited tool support, requires firewall configuration, single TCP port per server;
  - recommended applications: embedded devices, PC-based control, SCADA.

– Native binary with SOAP stack profile

  - mid-range performance, firewall friendly, requires UA secure conversation implementation;
  - recommended applications: MES/ERP, factory integration across the Internet.

11.1.4 Information model

This clause introduces how OPC-UA abstracts real objects, defines the related properties and relates objects to one another. Some examples are given to show semantic enablement supported by OPC-UA information model, as well as the usage of OPC-UA information model to abstract industrial elements or processes (e.g., defined by ISA-95).

OPC UA provides a framework that can be used to represent complex information as objects in an AddressSpace which can be accessed with standard web services. These objects consist of nodes connected by references.

Figure 11.1.4-1 depicts how OPC UA abstracts real objects (physical objects accessible by the OPC UA server application, e.g., PLCs which support OPC UA). AddressSpace is the foundation of OPC-UA servers which are the sources of industrial data. Real objects and their components are abstracted and represented by a set of nodes in the AddressSpace. Their definitions (by various node classes, related attributes or properties) and their references to relate to one another are also described. A view is a subset of the AddressSpace. Views are used to restrict the nodes that the server makes visible to the client to simply data acquisition and limit the access control of existing data in the servers.

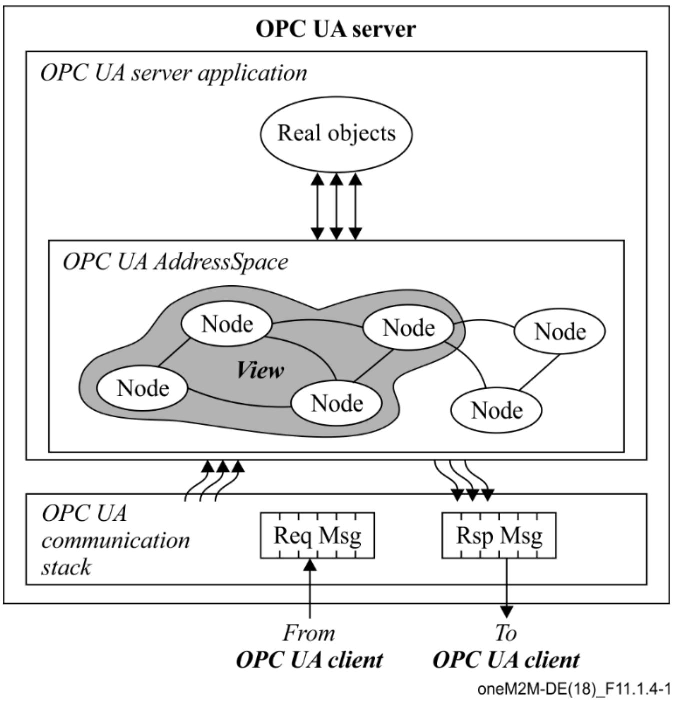


Figure 11.1.4-1 – OPC UA AddressSpace

OPC UA defines eight node classes to describe different kinds of real objects and their relationships (see Figure 11.1.4-2). The node classes include four types (ObjectType, VariableType, DataType, and ReferenceType) and four instances (object, variable, method and view) [b-OPC UA Part 5].

– The types are used to describe real objects and related properties (e.g., ObjectType, VariableType), represent data exposed in the AddressSpace (e.g., DataType) and allow industry specific data types to be used, or relate real objects (or their components and properties).

– Some of the instances such as object and variable are the respective instances of ObjectType and VariableType node classes. View is defined to show the visibility of selected real objects in OPC UA servers to data consumers (OPC UA clients). Method defines kinds of functions exposed by OPC UA server and these methods can be called by authorized OPC UA clients.

– References are defined to interconnect real objects or show their properties, such as HasTypeDefinetion to show the physical type of an object, HasComponent to show the construction of a physical object, AsymmetricReference to show the relationships between an object and another one with directional interconnection. OPC UA also supports the concept of sub typing (by HasSubtype and type definitions). This allows a modeller to take an existing type and extend it.

For example, a variable node represents a value that can be read or written. The variable node has an associated DataType that can define the actual value, such as a string, float, structure, etc. It can also describe the variable value as a variant. A method node represents a function that can be called. Every node has a number of attributes including a unique identifier called a NodeId and non-localized name called as BrowseName.

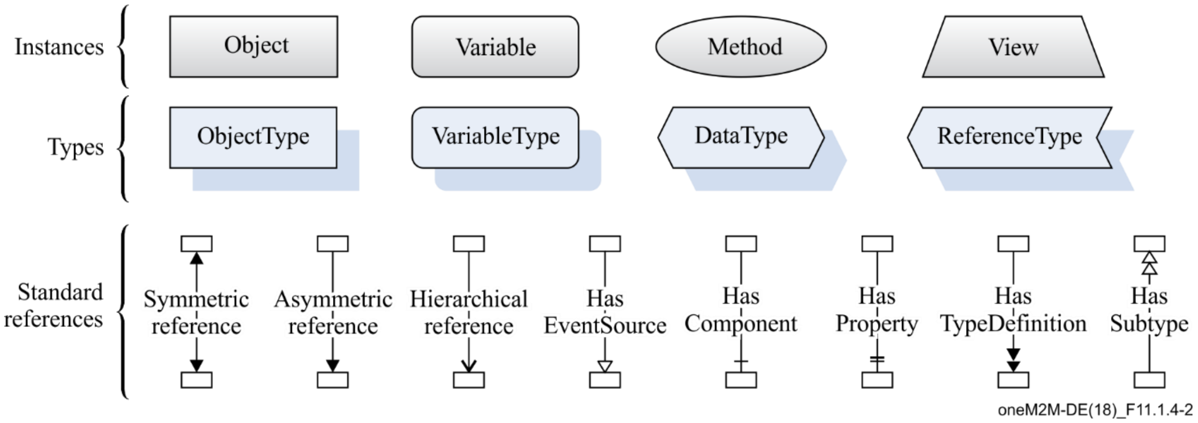


Figure 11.1.4-2 – Description of instances, types and references

An example of abstracting boiler used at the factory floor by OPC UA information model is show in Figure 11.1.4-3. The left part of the figure shows the instances which represent the physical characterises and functions of the boiler, meanwhile the right part of the figure shows the defined types using OPC UA information model.

The boiler (Boiler1) has three components show in the left part, a pipe (Pipe1001) to input the feedwater, a drum (Drum1001) to contain the water and another pipe (Pipe1002) to output the stream. Both the boiler and its components are abstracted as objects, and HasComponent ReferenceType is used to describe the construction. The direction of the flow to show the working procedure of this boiler (feed water to drum, and then drum generates stream) is described by an AsymmetricReference (FlowTo). Even the pipes have their own components, such as Pipe1001 has component "Flow Transmitter" as FT1001, Drum1001 has component "Level Indicator" as LI1001.

Both the flow transmitter and level indicator are the types of field devices. Their physical types are abstracted as the right part of the figure. Flow transmitter (FT) type and level indicator (LI) type are all extended subtypes of FieldDeviceType, and from the HasTypeDefinition reference type, the type definition could be queried. Vendor information such as VersionNo, UserDocumentation could be further searched by defining another subtype of existing type "FT type" and containing the information as a new subtype "Vendor FT type". The maintenance of field devices in the factory floor could be realized by relating the vendor to their products, such as HasVendor is an AsymmetricReference and it links the vendor's contact and documentation information to the devices. Additionally, the references can cross server borders which mean vendor information could be stored in an ERP system as long as the vendor information is also described by OPC UA information model.

OPC UA companion standards that permit industry groups (such as ISA/International Society of Automation-95, PLCopen) to define how their specific information models are to be represented in OPC UA server AddressSpaces.

ISA-95 defines its own common object models based on their standards [b-OPC UA for ISA-95]. The defined four classes of objects include: personnel information, role-based equipment information, physical asset information, and material information. Each of these object classes further include subtypes and have instances. The defined ISA95Object, ISA95Class, ISA95Property and ISA95Reference could be mapped to corresponding OPC UA definitions, as shown in Figure 11.1.4-4. Industry specific data types (e.g., all data types defined by ISA-95) could be used and mapped to OPC UA data types.

When industrial data is constructed by OPC UA information model, different classes of nodes convey different semantics and rich information could be linked to each object to build the foundation of semantic enablement for the industrial domain.

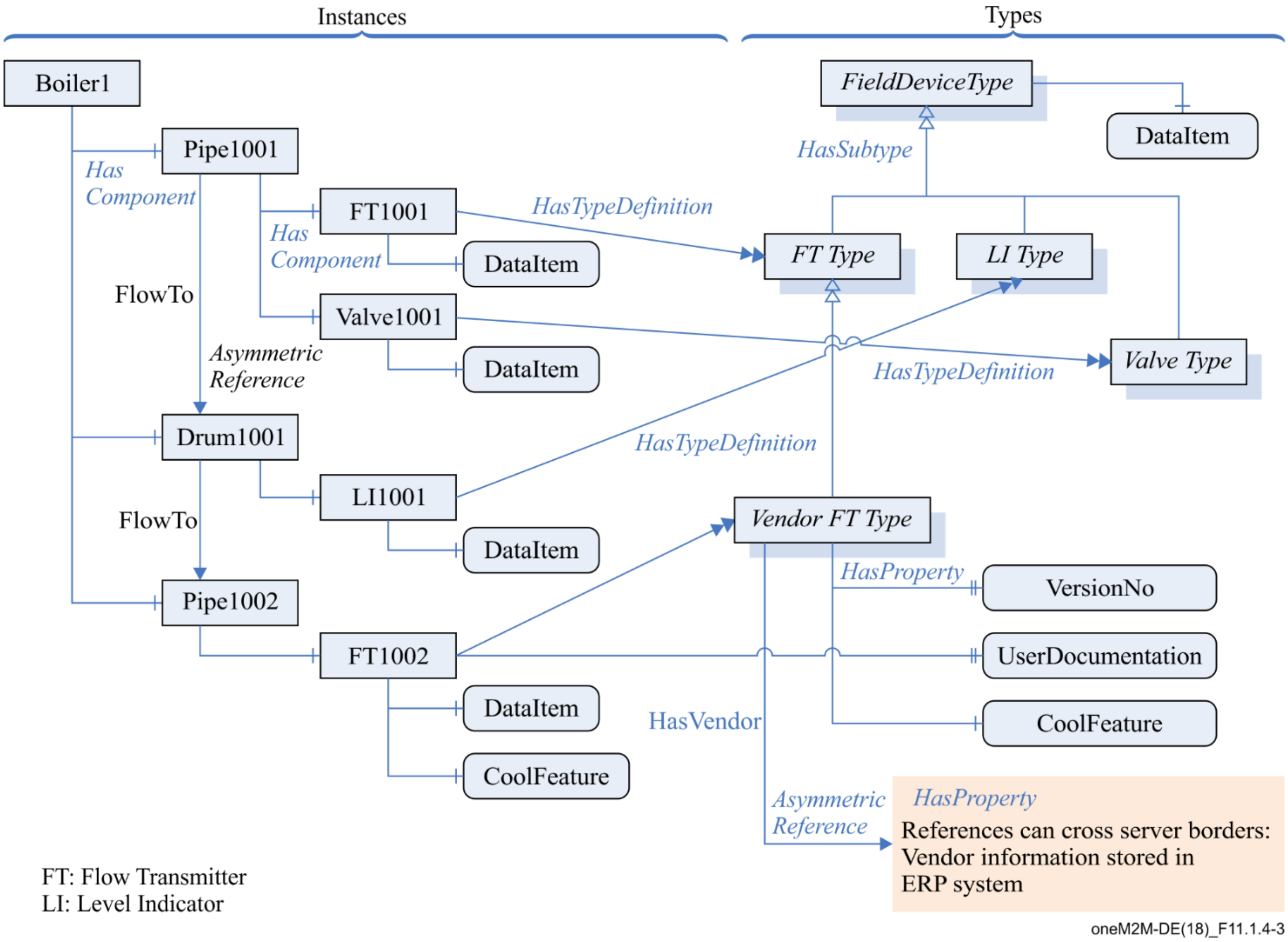


Figure 11.1.4-3 –An example of abstracting boiler at factory floor

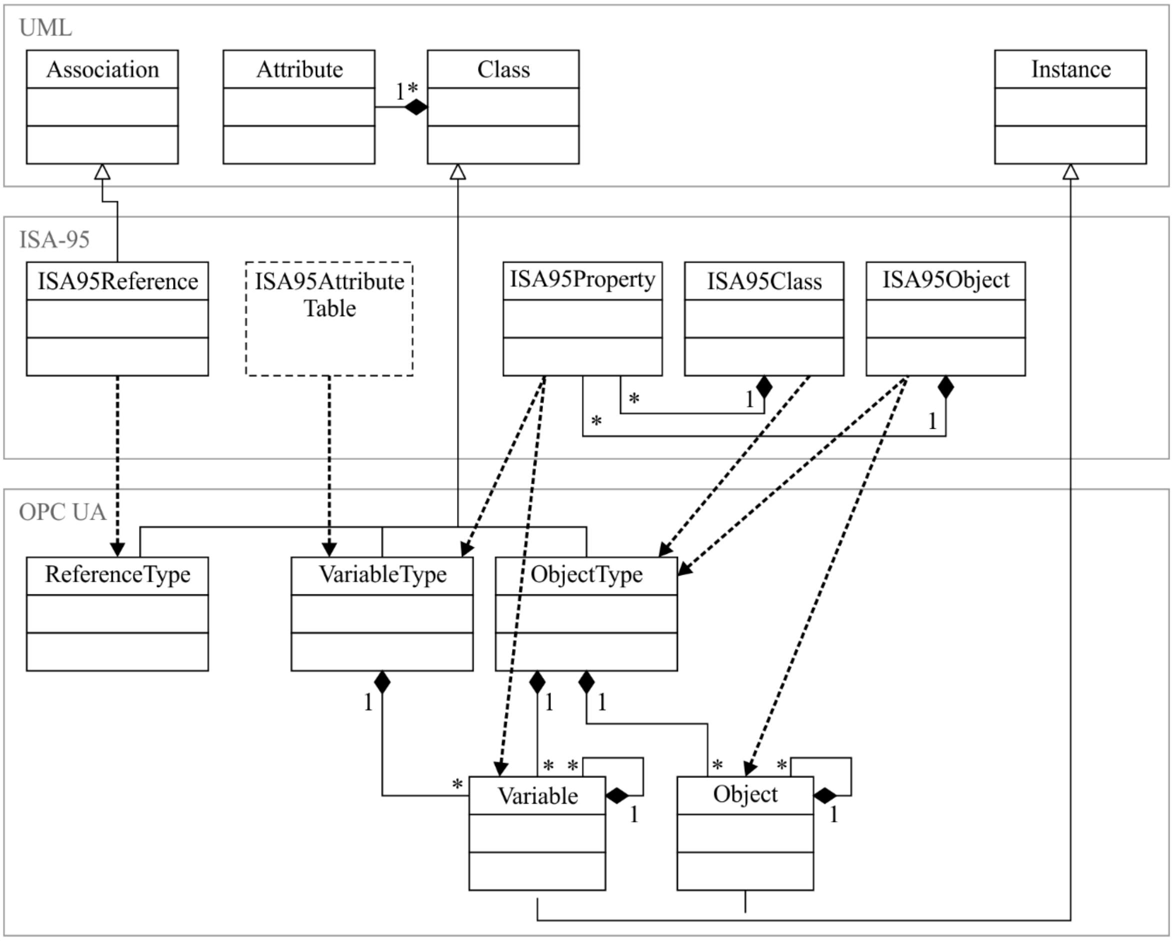


Figure 11.1.4-4 – Rules of mapping ISA-95 information models to OPC UA

11.1.5 Security

This clause introduces security considerations of OPC-UA.

The considerations in OPC-UA security include: threats and objectives, security model, services, mappings and profiles. Since OPC-UA specifies a communication protocol, the focus is securing the data exchanged between applications.

Threats and objective

Industrial domain is consisted of various network types and threats may exist in the communication through these networks, as show in Figure 11.1.5-1. By identifying the threats to the system (Table 11.1.5-1), identifying the system's vulnerabilities to these threats and providing countermeasures, industrial automation system security is achieved. The objectives have also been refined through years of experience (Table 11.1.5-2).

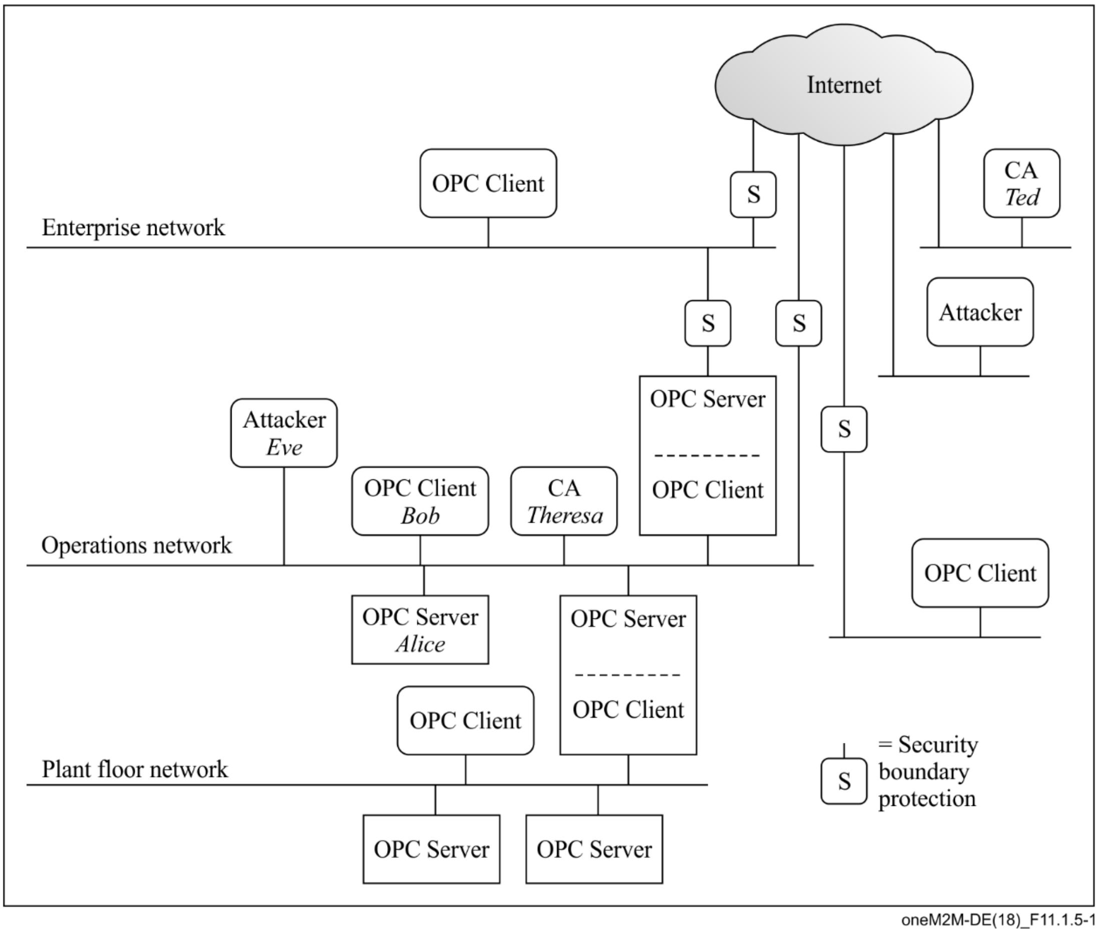


Figure 11.1.5-1 – OPC-UA network model

Table 11.1.5-1 – Security threats defined by OPC-UA

Threat	Description
Message flooding	A large volume of messages, or a single message that contains a large number of requests
Eavesdropping	Unauthorized disclosure of sensitive information
Message spoofing	Forged messages from a client or a server
Message alteration	Messages may be captured or modified and forwarded to OPC UA clients and servers
Message replay	Messages may be captured and resent at a later stage without modification
Malformed messages	Craft a variety of messages with invalid message structures or data values and send them to OPC UA clients or servers
Server profiling	Deduce the identity, type, software version, or vendor of the server or client
Session hijacking	Take over the session from an authorized user
Rogue server	Builds a malicious OPC UA server or installs an unauthorized instance of OPC UA server
Compromising user credentials	Illegally obtains user credentials

Table 11.1.5-2 – Objectives defined by OPC-UA

Objective	Description
Authentication	The process of verifying the identity of an entity such as a client, server or user
Authorization	Right or permission granted to an entity to access a system resource
Confidentiality	Protection from data being read by unintended parties
Integrity	Assurance that information was not modified during transmission
Auditing	Includes tracking of all activities and actions including security related activities
Availability	Assure that no system services have been compromised to become unavailable or severely degraded

Security architecture

OPC UA security architecture is shown in Figure 11.1.5-2.

The session is implemented by a UA application (application layer), and is responsible for high-level logical connection between the client and server. The session allows user-access to server (user authentication and authorization) and can only be created if a secure channel is already established.

The secure channel is implemented by a communication stack (communication layer) and is responsible for the low-level logical connection between the client and server. The secure channel secures outgoing messages and verifies incoming messages.

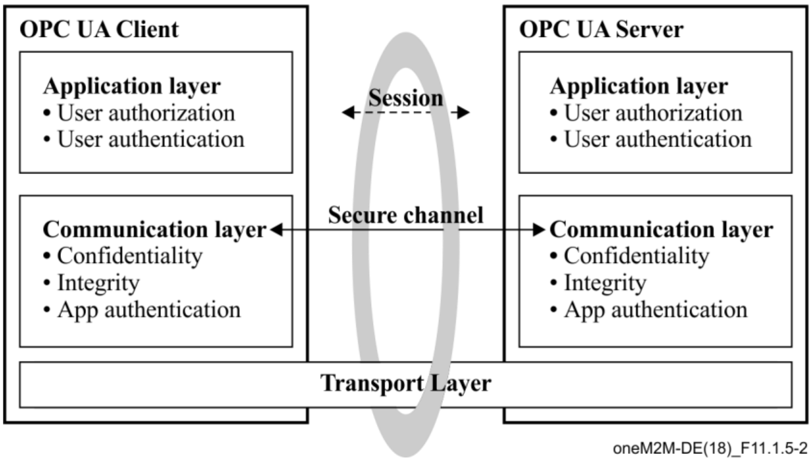
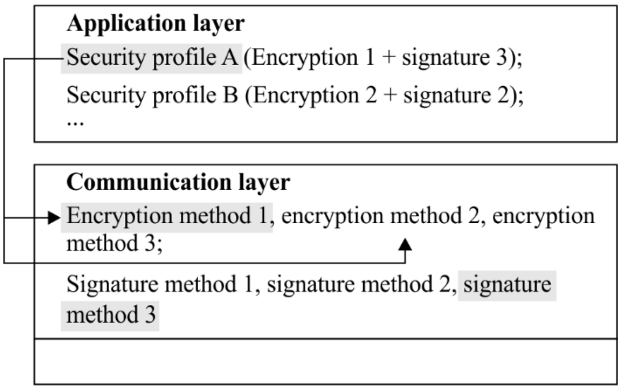


Figure 11.1.5-2 – OPC-UA security architecture

OPC UA also defines profiles (security policy) to provide security flexibility and extendibility. An example of OPC-UA security profile is shown in Figure 11.1.5-3.


Application layer
Security profile A (Encryption 1 + signature 3);
Security profile B (Encryption 2 + signature 2);
...
Communication layer
Encryption method 1, encryption method 2, encryption method 3;
Signature method 1, signature method 2, signature method 3

oneM2M-DE(18)_F11.1.5-3

Figure 11.1.5-3 – An example of OPC-UA security profile

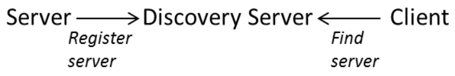
11.1.6 Technical comparison

This clause compares the services provided by OPC-UA and oneM2M, the resource representation of both sides and any other distinct differences.

Communication model

Table 11.1.6-1 demonstrates the comparison between oneM2M and OPC UA with respect to communication model.

Table 11.1.6-1 – Comparison of communication model of oneM2M and OPC UA system

	oneM2M	OPC UA
Registration need	Each entity in oneM2M must perform registration procedure which establishes a relationship between CSEs/AE allowing them to exchange information. Registree-Registrar relationship produces routing path when transferring messages.	To establish a connection between an OPC UA server and a client, each server needs to register with the discovery server, then a client could utilize discovery service set to discover OPC UA server through the discovery server (as below).  Server Discovery Server Client Register server Find server
Communication flow	Based on RESTful stateless request-response paradigm. Request message flow occurs from the originator CSE to the hosting CSE which generates hop-by-hop data delivery.	OPC UA client and OPC UA server directly communicate (peer-to-peer) without involvement of any other OPC UA entity. Request message occurs from OPC client and the paired OPC server then answers with Response messages. An industrial element could be a combined OPC UA client and server to realize data exchange between different industrial systems (e.g., SCADA requests data as a client and a machine provides data as a server, then SCADA continues to provide data as a server to another client in MES). But OPC UA client and server will not forward any request and response.

Resource Representation & Access

Table 11.1.6-2 demonstrates the comparison between oneM2M and OPC UA with respect to resource representation.

Table 11.1.6-2 – Comparison of resource representation of oneM2M and OPC UA system

	oneM2M	**OPC UA**
Functionality exposed by resources	oneM2M defines common M2M services which can be used for any vertical application service domains. oneM2M common services are exposed via oneM2M defined resource types which comprises of <container>, <group>, <locationPolicy>, <subscription>, <request>, <delivery> etc. Based on the device capability, oneM2M entity can be represented as CSE or AE. oneM2M resource model provides common service layer level interoperability.	OPC UA is designed for specific use in the industrial domain. It is a communication protocol other than a horizontal service platform like oneM2M. The exposed services are mainly for data exchange. For each OPC UA client, the exposed services on targeted server could be found by defined services sets (e.g., session service, attribute service, subscription service). The client can further call methods defined on paired server. E.g., by attribute service set, clients are allowed to read and write attributes of nodes on paired servers.
Resource access control	oneM2M control the access rights by defining <accessControlPolicy>, which stores a representation of privileges. It is associated with resources that shall be accessible to entities external to the hosting CSE. It controls "who" is allowed to do "what" and the context in which it can be used for accessing resources.	OPC UA control the access range of clients to resources located on severs by view service set. E.g., For device maintenance application, an electric engineer has different access right with another mechanic engineer to the attributes on the same machine. The corresponding views of them to browse attributes on the same machine (OPC UA server) differ.
Resource structure	Hierarchical resource structure <CSEBase> represents the oneM2M CSE and is the root for all the resources. oneM2M resources have a parent-child relationship. Each resource in oneM2M entity can be accessed via both structured (hierarchical) URI as well as un-structured (flat) URI.	Hierarchical resource structure In OPC UA address space, nodes represent the root for all resources. OPC UA resources also have a parent-child relationship, and have rich references to other nodes.

Subscription and notification

As subscription and notification is one of the most important service used in the industrial domain, Table 11.1.6-3 demonstrates the comparison between oneM2M and OPC UA with respect to subscription and notification service/functionality.

Table 11.1.6-3 – Comparison of subscription and notification functionality of oneM2M and OPC UA system

	oneM2M	OPC UA
Subscription and notification functionality	oneM2M defines subscription and notification CSF and realize this functionality by <subscription> to subscribe resource or events, and notify based on defined policies. Group management can support group-based subscription and notification, as well as aggregated notification from group members. oneM2M supports tracking the change of attribute(s) and direct child resource(s) of the subscribed-to resource.	OPC UA defines a subscription service set together with a monitored item service set to realize flexible subscription and publication functionality. OPC UA supports monitoring attributes for value changes and monitoring objects for events. The definition of monitoring is richer, such as support triggering a publication after a change of attribute value is monitored comparing with a pre-defined threshold. Group-based (batch) subscription and publication is under consideration but not yet supported. OPC UA supports aggregation of publication based on defined aggregation policies (instead of based on group), such as pre-defined monitor period, or data pre-process (average, maximum or minimum).

11.2 Scenarios for interworking

This clause defines the possible scenarios for oneM2M interworking with OPC-UA. Several cases of interworking will be described.

11.2.1 Overview of interworking scenarios

The scenarios for oneM2M and OPC-UA interworking are summarized as below. The oneM2M system may support several of these scenarios simultaneously:

– OPC-UA system interact with oneM2M infrastructure domain (clause 11.2.2).

– OPC-UA systems in the field domain interact with each other via oneM2M infrastructure domain (clause 11.2.3).

– OPC-UA system interact with oneM2M field domain via oneM2M infrastructure domain (clause 11.2.4).

– OPC-UA system directly interact with oneM2M field domain (clause 11.2.5).

11.2.2 OPC-UA system interact with oneM2M infrastructure domain

In Figure 11.2.2-1, OPC-UA system is deployed in the field domain, the OPC-UA system interacts with the oneM2M infrastructure domain. The M2M applications access oneM2M infrastructure domain to utilize the data from OPC-UA system. For example, the use case "An Industrial Use Case for On-demand Data Collection for Factories" collected in TR-0018 describes the data in the factory is required to be integrated to M2M platform for advanced data analytics to enhance manufacturing. The production data from OPC-UA system could be mapped to oneM2M data at oneM2M infrastructure domain (IPE of the IN supports the OPC-UA interworking functionality), or OPC-UA data could also be mapped to oneM2M system at the edge gateway of a factory (IPE of a MN in the oneM2M field domain supports the OPC-UA interworking functionality).

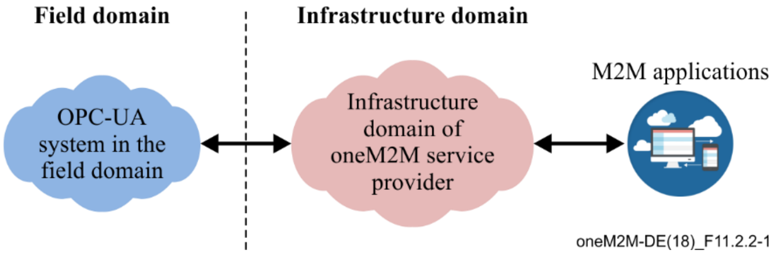

Field domain
Infrastructure domain
M2M applications
OPC-UA system in the field domain
Infrastructure domain of oneM2M service provider
oneM2M-DE(18)_F11.2.2-1

Figure 11.2.2-1 – OPC-UA system in the field domain interacts with oneM2M infrastructure domain

11.2.3 OPC-UA systems in the field domain interact with each other via oneM2M infrastructure domain

In Figure 11.2.3-1, two OPC-UA systems are assumed to be deployed in different locations in the field domain. The OPC-UA systems could interact with each other via oneM2M infrastructure domain. For example, the use case "Data Process for Inter-factory Manufacturing" collected in TR-0018 describes data collaboration among factories (at different sites). To optimize productivity through collaborative production, an OPC-UA system inside a factory exchanges the production data with another OPC-UA system inside a different factory via M2M platform.

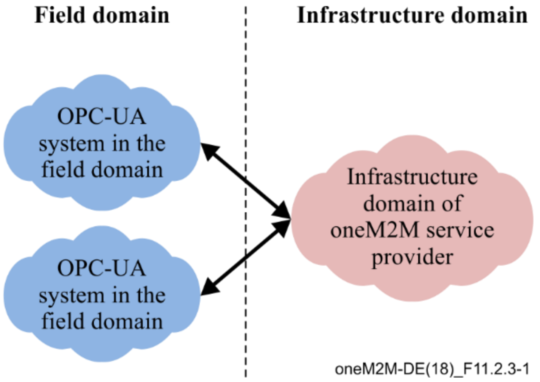

Field domain
Infrastructure domain
OPC-UA system in the field domain
OPC-UA system in the field domain
Infrastructure domain of oneM2M service provider
oneM2M-DE(18)_F11.2.3-1

Figure 11.2.3-1 – Two OPC-UA systems in the field domain interact with each other via oneM2M infrastructure domain

11.2.4 OPC-UA system interact with oneM2M field domain via oneM2M infrastructure domain

In Figure 11.2.4-1, OPC-UA system is deployed in the field domain, a oneM2M system is also deployed in the field domain but in a different location. The OPC-UA system interacts with the oneM2M field domain via oneM2M infrastructure domain. For example, optimization of production and supply chain management (SCM) in real-time is required for smart manufacturing. OPC-UA is widely adopted in the production system in factories, and a smart logistic system is possibly realized by oneM2M standards. The two systems have the needs for interaction via M2M platform and require the feedback of analysis results toward procurement planning and logistic planning supplied by domain applications.

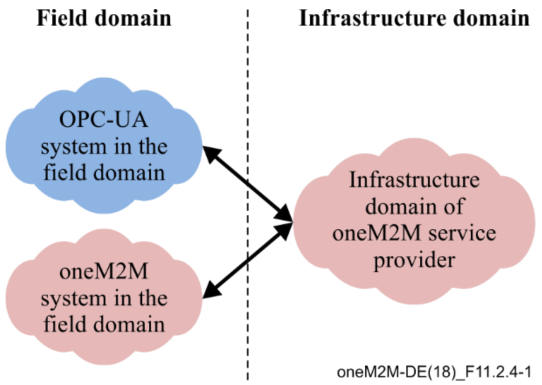

Field domain
Infrastructure domain
OPC-UA system in the field domain
oneM2M system in the field domain
Infrastructure domain of oneM2M service provider
oneM2M-DE(18)_F11.2.4-1

Figure 11.2.4-1 – OPC-UA and oneM2M system in the field domain interact with each other via oneM2M infrastructure domain

11.2.5 OPC-UA system directly interact with oneM2M field domain

In Figure 11.2.5-1, OPC-UA system is deployed in the field domain, and a oneM2M system is deployed in the same location in the field domain. The OPC-UA system could directly interact with oneM2M field domain without oneM2M infrastructure domain. For example, OPC-UA is used in the production system in a factory, and an inside-factory logistic system is constructed by new M2M device types which could realize oneM2M standards. The OPC-UA system could directly interact with the local oneM2M system without going through the oneM2M infrastructure domain.

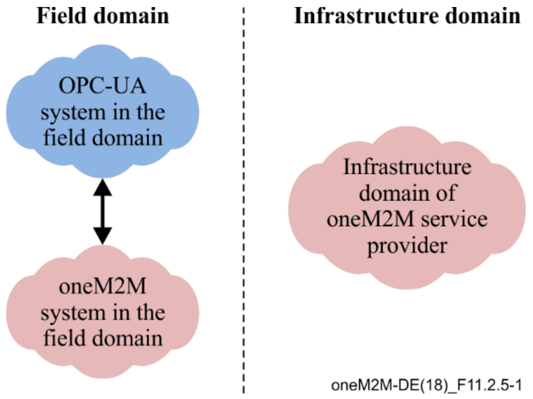

Field domain
Infrastructure domain
OPC-UA system in the field domain
oneM2M system in the field domain
Infrastructure domain of oneM2M service provider
oneM2M-DE(18)_F11.2.5-1

Figure 11.2.5-1 – OPC-UA system in the field domain directly interact with oneM2M system in the field domain

11.3 Possible solutions to address interworking

This clause studies the possible solutions to realize oneM2M interworking with OPC-UA. The architecture of interworking will be proposed with the usage of IPE. Solutions of resource mapping and procedure mapping based on OPC-UA information model will be studied.

11.3.1 Introduction

This clause describes solutions for oneM2M and OPC UA interworking and provides:

- Functional architecture for interworking
- Resource model mapping
- Procedure mapping

Firstly, functional architecture with main entities is described. Then possible resource model mappings are discussed. Next, procedure mapping is described, i.e., interaction between main entities e.g., discovery or registration, subscription and notification. This functional architecture and possible interworking solutions can address the defined interworking scenarios between oneM2M and OPC UA in clause 11.2.

11.3.2 Functional architecture for interworking

The OPC UA interworking scheme is based on IPE which is a specialized AE for interworking. As depicted in Figure 11.3.2-1, the IPE is characterized by the support of a non-oneM2M reference point (OPC UA interface) and by the mapping the non-oneM2M (OPC UA) data models to oneM2M resources exposed via the Mca reference point.

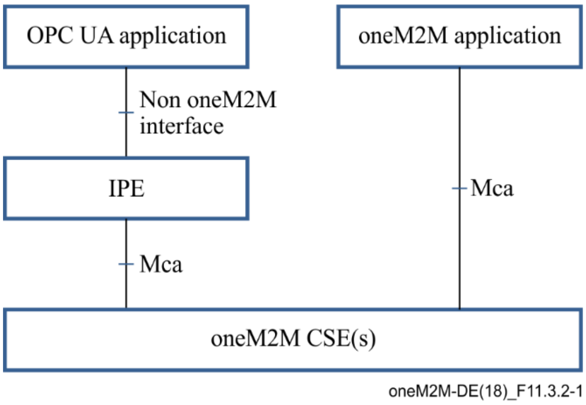


Figure 11.3.2-1 – Functional architecture with IPE

The main entities and their characteristics are as follows:

- **IPE**: a specialized AE, characterized by the support of a non-oneM2M reference point, and by the capability of mapping the OPC UA data model to the oneM2M resources exposed via the oneM2M Mca reference point.
- **IWK** (interworking) **function**: a function of IPE which translates the OPC UA entities into oneM2M resources e.g., OPC UA specific AE, container.
- **OPC UA device**: NoDN (non-oneM2M device node) which plays either OPC UA client or OPC UA server roles or both. The OPC UA server hosts OPC UA resources. OPC UA client accesses the OPC UA server to read or write OPC UA attributes, e.g., for monitoring and controlling industrial devices where OPC UA is installed.

Figure 11.3.2-2 demonstrates a possible deployment model which describes the main entities of OPC UA and oneM2M interworking via reference points. The deployment shows an example of deploying IPE in a MN, but it is also possible that the deployment of IPE locates in an IN. The uniform interworking functions provided by the IPE (no matter locates in MN or IN) could address the defined interworking scenarios between oneM2M and OPC UA in clause 11.2.

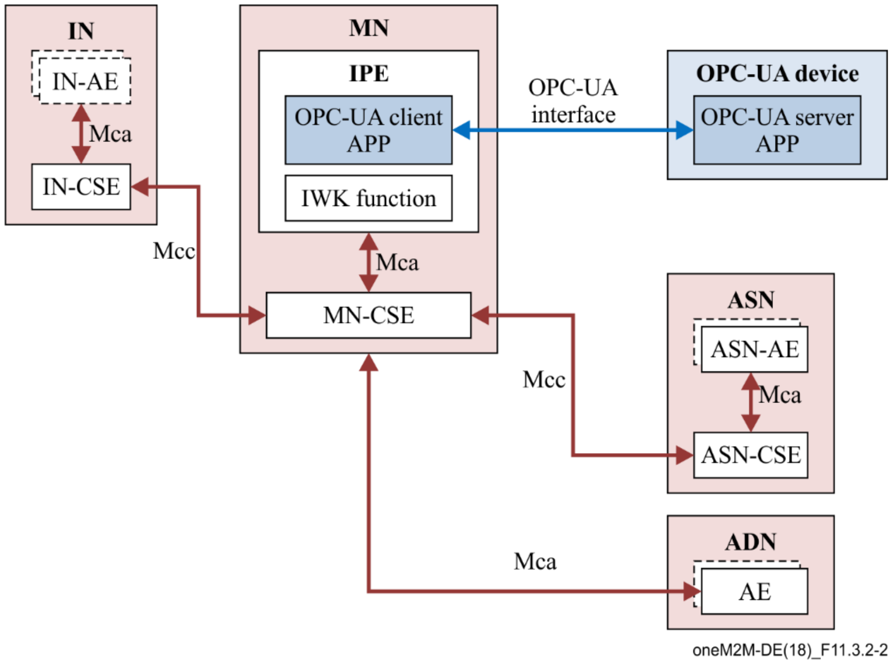


Figure 11.3.2-2 – Deployment model based on functional architecture

11.3.3 Resource model mapping

11.3.3.1 Introduction

As introduced in clause 11.1.4, the resource structure of OPC UA server is hierarchical and organized by its "AddressSpace". (see Figure 11.1.4-1). The AddressSpace is modelled as a set of nodes which represent real objects and their components and are accessible by OPC UA clients using OPC UA services.

Therefore the possible mapping between oneM2M and OPC UA entities is considered based on several levels of the standard AddressSpace structure of OPC UA servers. Then the mapping solution are analyzed and recommendations are proposed.

11.3.3.2 Generic entities mapping

OPC UA defines the entities by using the concept of "object", but the objects located in the OPC UA servers are only accessible through "AddressSpace" where objects are represented by a set of nodes as depicted in Figure 11.3.3.2-1. Several node types (classes) are defined by OPC UA as introduced in clause 11.1.4. And "Object" node type is to abstract the real objects (e.g., an industrial device and its components) and to related one node to others through defined "references" in this object node to include semantic information (e.g., construction of a device, device characteristics and capabilities).

OPC UA clients communicate with OPC UA servers to acquire industrial data. All the data on a OPC UA server is organized in the server's AddressSpace (as Figure 11.3.3.2-1) and could be browsed and further accessed by OPC UA clients. The most essential node in the AddressSpace is the Object node and its subtypes such as "variables", "methods" and "references". The content of real objects including current and historic records could be accessed through "Read/Write" commands, and could be subscribed based on defined policies (e.g., events, or selected monitored list) through invoking the supported "methods".

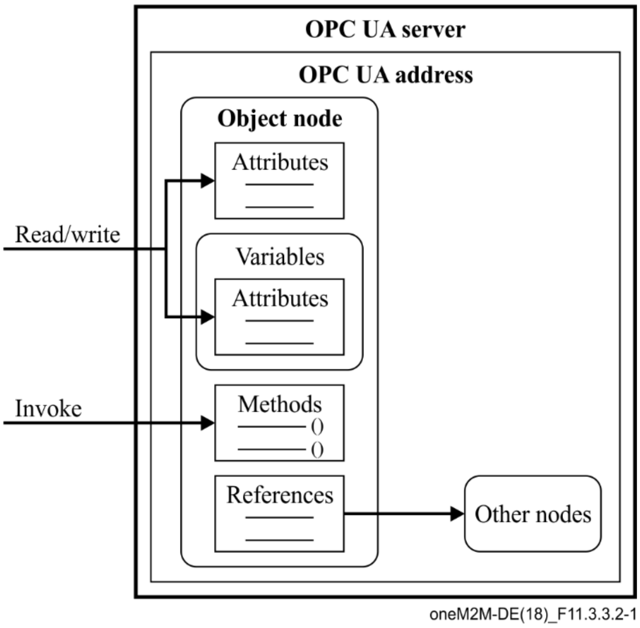


Figure 11.3.3.2-1 – OPC UA node model

Standardized nodes as entry points into AddressSpace are shown in the dashed rectangle in Figure 11.3.3.2-2. They are the top levels standardized for all servers, and are summarized as below.

– **Root**: the browse entry point for the AddressSpace.

– **Views**: the browse entry point for views, will use references to display only a specific subset of the data which clients have interests (the entire AddressSpace is the default view). As in Figure 11.3.3.2-2, "BoilerView" defines a dedicated subset of boiler related objects and attributes.

– **Objects**: the browse entry point for Object nodes, the "Objects" node can also reference View nodes using organizes references (one type of references). As in Figure 11.3.3.2-2, several objects representing physical areas are defined, e.g., "Area 1" means a geographical area, and "BoilerArea" means a dedicated logical area or group containing all boilers. A boiler named "Boiler1" is located in both "Area 1" and "BoilerArea", and it could be browsed by interested OPC UA client application through defined "BoilerView".

– **Types, ObjectType, VariableType, ReferenceType and DataType**: the browse entry point for node types. As introduced in clause 11.1.4, there are four node types defined in OPC-UA. And in Figure 11.3.3.2-2, the boiler "Boiler1" references a "BoilerType" Object type which could include semantic information of this device type.

– **Server:** to show the server type of the OPC UA server.

In Figure 11.3.3.2-2, arrows represent references between OPC-UA entities.

Based on the OPC UA structured AddressSpace, the following levels of abstraction are modelled i.e., "Root" standard object, standard objects beneath Root and other resources including methods, attributes and references, etc.

The possible mapping for these different levels of model is:

– OPC UA "Root" standard object – oneM2M <AE> resource.

– OPC UA standard objects beneath Root – oneM2M <container>/ <flexContainer> resource.

– Other resources (e.g., methods, attributes, references) – oneM2M <flexContainer> / <contentInstance> resource.

Then the generic mapping OPC UA entities to oneM2M resources are shown in Figure 11.3.3.2-3. Further analysis will be presented in the next clause.

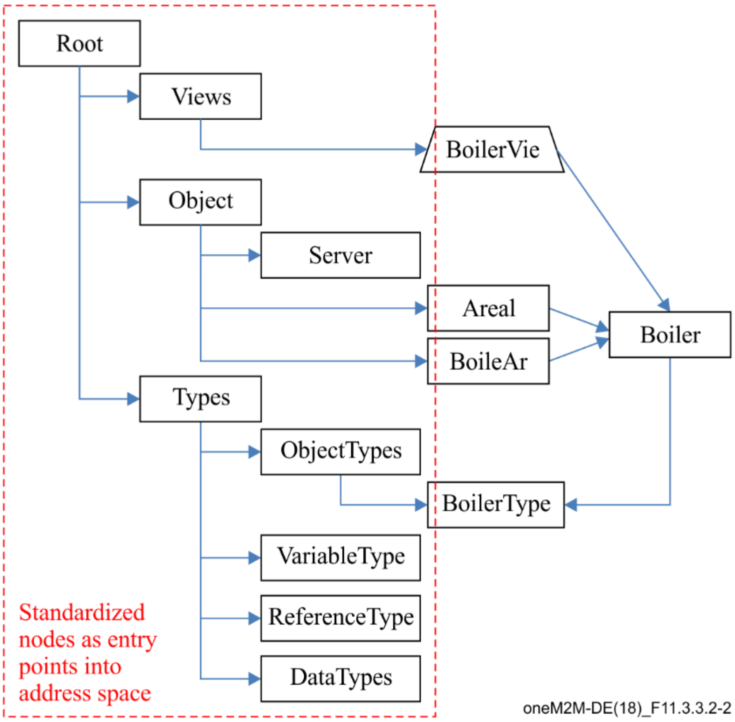


Figure 11.3.3.2-2 – Standard address space structure of an OPC UA server

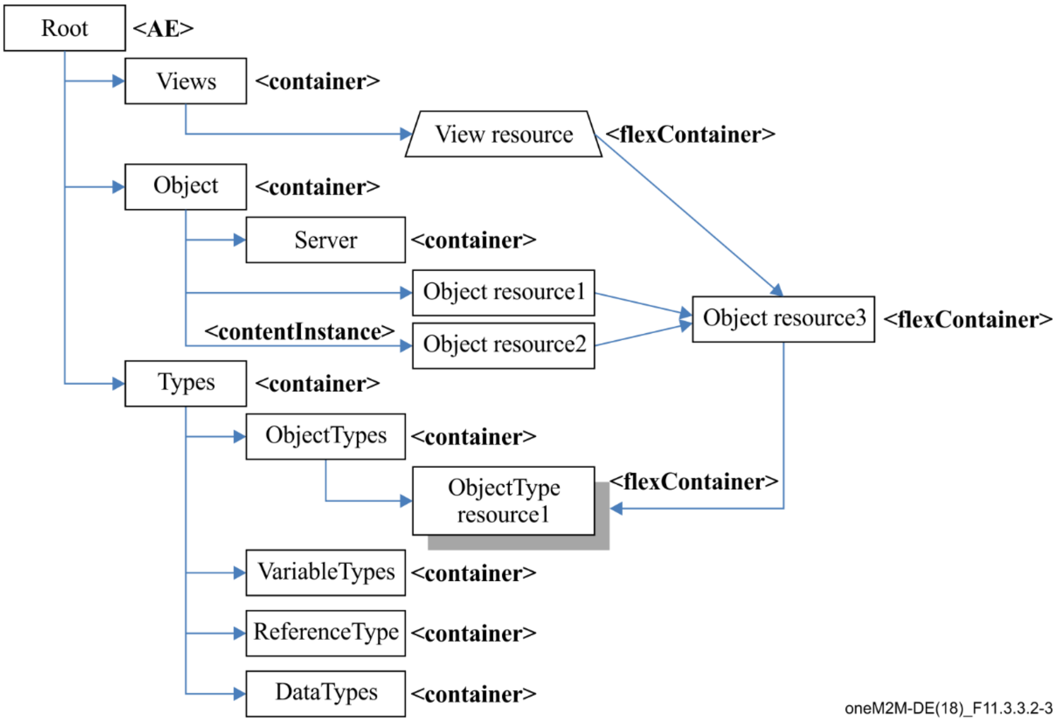


Figure 11.3.3.2-3 – Generic entities mapping

11.3.3.3 Analysis and recommendations

Based on the utilization of OPC UA in the industrial domain, the main scenario of OPC UA and oneM2M interworking would be industrial data exchanging between OPC UA servers and oneM2M system or data exchanging between OPC UA servers through oneM2M system. All of these OPC UA server are structured by the address space, therefore the top level standardized object "Root" as the entry point of any OPC UA server's address space is mapped to <AE>, which would need creation of an <AE> resource by the hosting CSE (located in the interworking node such as a MN) after necessary authentication and proper access rights definition for the <AE> resource.

Then to further browse the OPC UA server's address space, several standard objects beneath Root (e.g., views, objects, types) are defined as the subtypes of Root object to act as the standardized entries for main OPC UA functionalities, such as variables browsing and subscription. Therefore these standard objects beneath Root are suggested to be defined as the child resources of <AE> resource, which could be <container> or <flexContainer> resource.

Next, any other OPC UA resources e.g., attributes, methods, references are represented as a content sharing resource in oneM2M. But these resources may require different access control privileges on their attributes. For example, "Variable" node type has several attribute definitions, and the "UserAccessLevel" attribute is defined to limit the access of the attributes taking user access rights into account. Similarly, "Method" node type also has several attribute definitions, and the "UserExecutable" attribute is defined to limit the access of the methods taking user access rights into account. But "reference" node type usually does not need specific access rights for different users. Flexible mapping of these OPC UA resources are suggested depending on the necessity of user dependent access rights. Additionally, "Method" node type in OPC UA has "InputArguments" and "OutputArguments" attributes defining the arguments used when calling the Method. These contents have one option to be adopted directly as the child resources of <flexContainer>, while another option is to be mapped as attributes of oneM2M <flexContainer> resource, which is more efficient (Similarly with Alljoyn interworking, defining "input" and "output" parameters of [allJoynMethodCall] resource by using the [customAttribute] attribute of oneM2M <flexContainer> resource type.). Content sharing resource in the oneM2M such as <flexContainer> or <contentInstance> shall be selected for mapping based on the possibility of defining <accessControlPolicyIDs> as their child resources or other considerations such as better usability of oneM2M resources.

The existing resource type definition in oneM2M is preferred to be used for mapping OPC UA data models to oneM2M resources. And below options are considered for the possible mapping.

Table 11.3.3.3-1 – Analysis of resource mapping

Options		PROs	CONs
Mapping OPC UA standard objects beneath Root	**Option 1**: Map standard objects beneath Root to <container> resource	Simple mapping	Some of the mandatory attributes of <container> resource could not have corresponding attributes from OPC UA, needs to be filled by e.g., creator of the resource
	Option 2: Map standard objects beneath Root to <flexContainer> resource	Less mandatory attributes; customizable attributes could directly indicate associated content of the OPC UA standard objects	Need detailed definition of <flexContainer> resource in further interworking specifications

Table 11.3.3.3-1 – Analysis of resource mapping

Options		PROs	CONs
Mapping other resources (e.g., methods, attributes, references)	**Option 3**: Map all other resources (e.g., methods, attributes, references) to <flexContainer> resource	Simple mapping	Some of the OPC UA resources e.g., reference node type does not need dedicated <accessControlPolicyIDs>, and creation of <contentInstance> resources could be sufficient
	Option 4: Map part of other resources to <flexContainer> resource, and part of other resources to <contentInstance> resource	Flexible resource mapping and possible less memory occupation on IPE	Complex mapping: need to define which specific OPC UA resource types shall be mapped to which oneM2M resource types

Recommendations

For the level of OPC UA standard objects beneath Root mapping, both Option 1 and Option 2 are feasible mapping solutions on IPEs.

For the level of other OPC UA resources mapping, Option 4 is suggested for more flexibility of resource mapping due to the variety of remaining OPC UA resources types.

Additionally, below considerations are suggested when mapping resources.

When mapping the OPC UA Root object to oneM2M <AE> resource, some mandatory AE attributes don't have corresponding attributes from OPC UA, and these attributes could be mandatory to be supported in oneM2M. Thus, it is recommended that these attributes be provided by the creator of such a resource e.g., CSE or filled by the platform.

11.3.4 Procedure mapping

As introduced within clause 11.1.6, there exist technical differences on OPC UA and oneM2M service procedures. In order to support oneM2M entities to cooperate with OPC UA devices in the above interworking scenarios in clause 11.2, the following connection establishment and data collection procedures from OPC UA device are considered.

11.3.4.1 Connection establishment

To interwork with OPC UA system, oneM2M needs to establish the communication session in between at the beginning, and represent data on OPC-UA devices as oneM2M resources.

11.3.4.1.1 Initialization

In the initialization stage, oneM2M IPE functionality will be installed on MN in the field domain or IN in the infrastructure domain. Figure 11.3.4.1.1-1 provides a possible initialization procedure of related entities involving MN-IPE. MN-IPE performs the registration with MN-CSE, and an oneM2M ASN-AE or ADN-AE registers through IN-CSE on the MN-CSE for communication with an OPC-UA device. The consequent discovery procedure is based on the initialization.

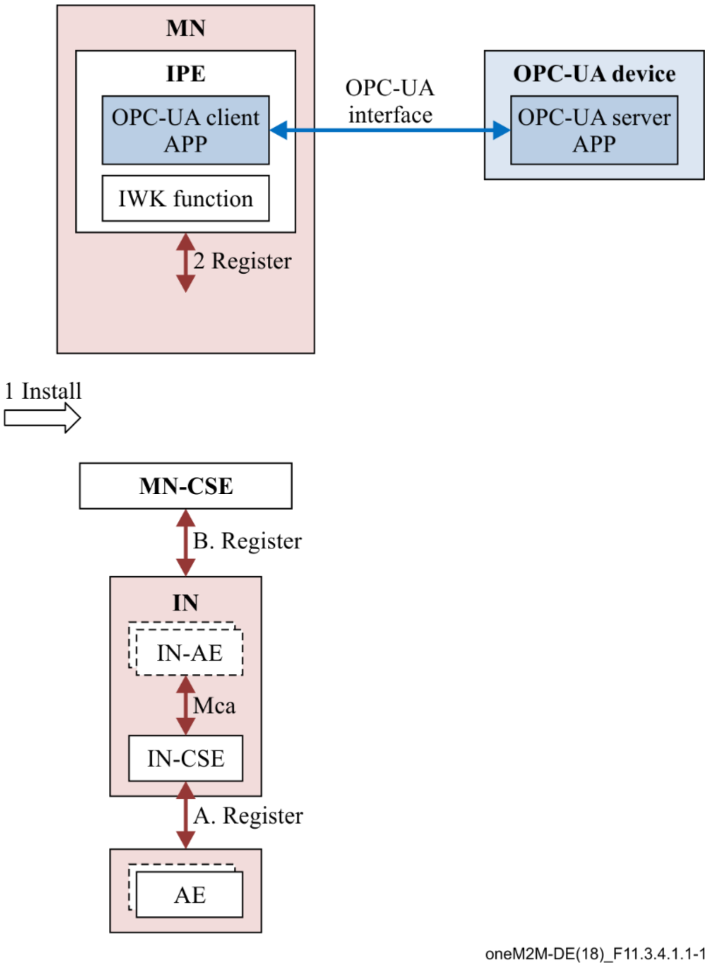


Figure 11.3.4.1.1-1 – Initialization procedures

Below are the possible steps for initialization as depicted in the figure:

1) IPE is installed on MN through one of out of band mechanisms
   - triggered by the end user action, e.g., a user subscribing physically to the service; or
   - based on SP business logic.
2) IPE registers on MN-CSE.

Subsequently, the oneM2M AE can communicate with IPE residing in MN which provides OPC UA interworking function by:

1) Registers itself with IN-CSE.
2) And IN-CSE registers on MN-CSE.

11.3.4.1.2 Discovery

Then a possible solution for oneM2M entities to discover OPC UA device in request-response mode via oneM2M IPE is shown in Figure 11.3.4.1.2-1.

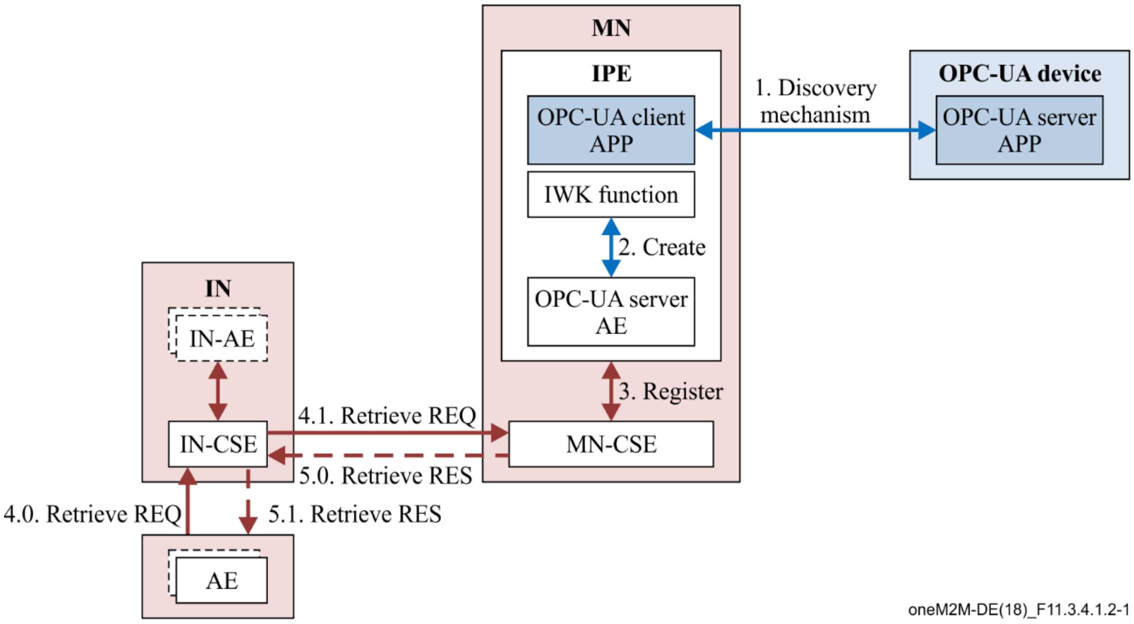


Figure 11.3.4.1.2-1 – Discovery procedures

1) IPE discovers OPC UA devices utilizing OPC UA discovery mechanisms. In a typical case, OPC UA client may use the DiscoveryUrl by out-of-band method (i.e., entry into a GUI) to discover the corresponding server straightly. Or it is also possible that LocalDiscoveryServer and/or GlobalDiscoveryServer could be involved in more complex cases. Then OPC UA client can build the session to its server via SecureChannel service set and session service set.

2) As soon as OPC UA discovery is done, IWK function within IPE will map the discovered OPC UA server and related resources into oneM2M resources and creates them (mapping from the above clause 11.3.3 as AE resources) on MN.

3) Each OPC UA server AE then registers with MN-CSE for the discovered OPC UA servers.

4) (4.0/4.1) After that an AE can request to discover OPC UA server AE resources by sending a RETRIEVE REQ via IN-CSE with the appropriate filter criteria to MN-CSE where they are registered.

5) (5.0/5.1) MN-CSE replies a RETRIEVE RES to IN-CSE with the OPC UA server AE's information which in turn is sent to the AE.

11.3.4.2 Data collection from OPC-UA device

11.3.4.2.1 Simple reading procedures

To realize data collection from OPC UA devices, a direct way is to send RETRIEVE request from the originator AE to IPE as the receiver, as shown in Figure 11.3.4.2.1-1.

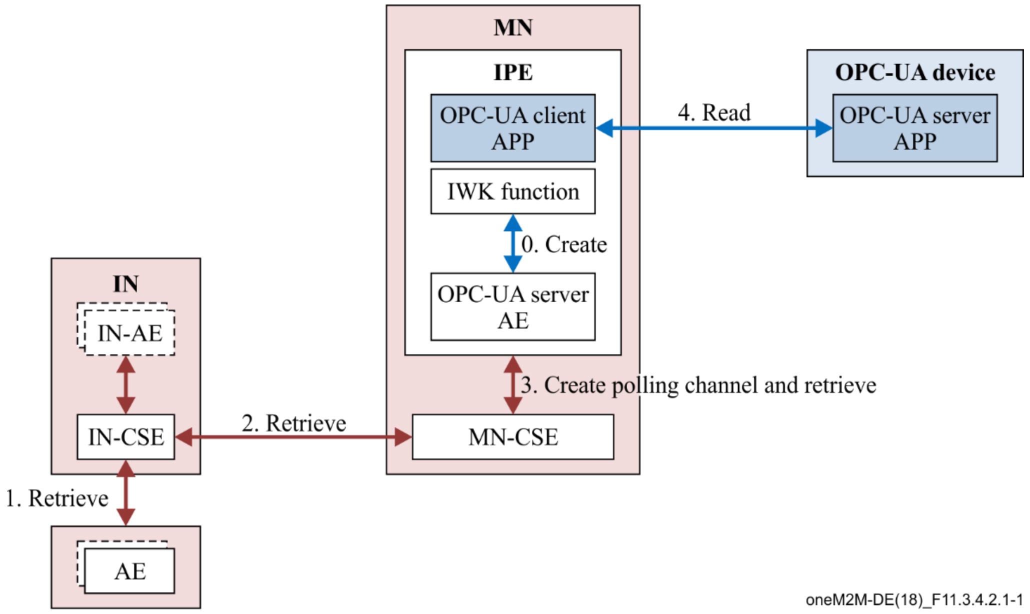


Figure 11.3.4.2.1-1 – Simple reading procedures

0) In the discovery procedures, IWK function in IPE has already mapped OPC UA server with related resources to oneM2M AE.

1) The originator AE tries to collect the OPC UA data from IN-CSE by Retrieve message.

2) IN-CSE forwards the Retrieve to MN-CSE.

3) MN-CSE reaches IPE via polling channel Create and Retrieve, where Retrieve is mapped to OPC UA "read" message.

4) IPE acts as OPC UA client to read the required data from its server residing in the OPC UA device. The data may include one or more attributes of OPC UA devices. With structured attribute values, whose elements are indexed as in an array, the transferred data can be the entire set of indexed values, specific areas or individual elements.

11.3.4.2.2 Subscription and notification procedures

The other way of data collection is for IN-CSE to create a subscription to get notification on the change of the corresponding OPC UA resources in the address space (of OPC UA servers).

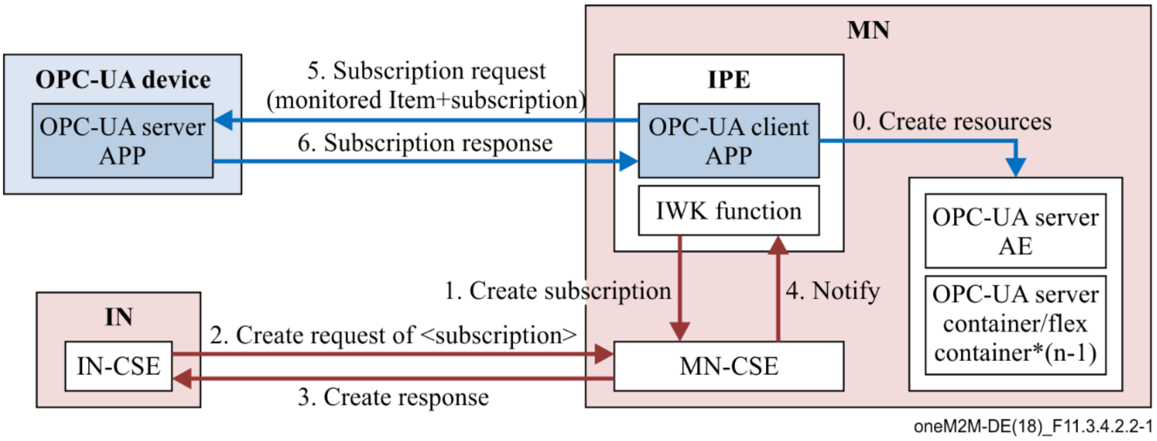


Figure 11.3.4.2.2-1 – Interworking procedure for subscription

0) oneM2M subscription tracks the change of attribute(s) and direct child resource(s) of the subscribed-to resource, while OPC UA can support richer subscription and notification functionality (via defined monitored item and subscription service sets) to get the changes of

whole AddressSpace of an OPC UA server. As a result, IPE needs to create n-1 layer(s) of <container>/<flexContainer> child resource to match OPC UA device's n-layer resource structure.

1) IPE will further create subscribed-to <container>/<flexContainer> resource at CSE for getting notifications about request of <subscription> resource.

2) IN-CSE sends a CREATE request of <subscription> resource targeting to the <container>/<flexContainer> resource under an OPC UA server AE.

3) MN-CSE replies the CREATE Response message to the IN-CSE.

4) MN-CSE sends a NOTIFY Request message to IPE of <subscription> resource being created by the IN-CSE.

5) OPC UA client in IPE maps the NOTIFY request to an OPC UA Subscribe Request message to OPC UA server, and the server utilizes MonitoredItem and subscription services to subscribe OPC UA data or events.

6) OPC UA server sends an OPC UA reply with Subscription Response message to OPC UA client in IPE.

Notice that in step 0, the subscribed-to resource could be <container>, <flexContainer> or <contentInstance >. However, <contentInstance> is not able to have a child resource of any of the three kinds of resources above, and it is either not able to have a <subscription> child resource created for tracking the changes on this resource type. This would disable the subscription of any OPC UA resources who has child resources possibly to be subscribed-to- resources. Therefore, OPC UA resources (e.g., Object type) containing changeable attributes or having variables as child resources which might be subscribed by industrial applications, should be considered to be mapped to <container> or <flexContainer> resources. Other unchangeable resources for describing the functionalities (e.g., Method type), or relationships with other OPC UA nodes (e.g., reference type) could still be mapped to <contentInstance> as well as <container> or <flexContainer>. As a result, those resources mapped to <contentInstance> cannot be subscribed, but the data collection can only be achieved via simple reading procedures.

Consequently, OPC UA device could transfer data contained in NotificationMessages to MN-IPE and notify the modified resource to IN-CSE as shown in Figure 11.3.4.2.2-2.

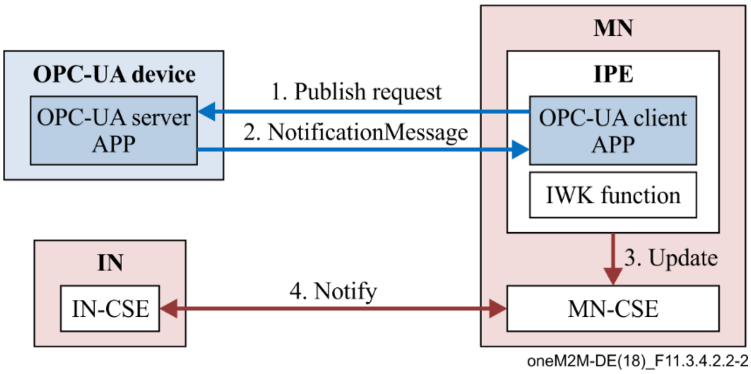


Figure 11.3.4.2.2-2 – Interworking procedure for notification

1) OPC UA client sends a Publish request message to OPC UA server.

2) OPC UA server generates a NotificationMessage including the subscribed-to value(s) and/or event(s) to reply to OPC UA client.

3) IPE maps OPC UA NotificationMessage to an UPDATE request message to MN-CSE.

4) MN-CSE sends a NOTIFY request message to the IN-CSE, and IN-CSE answers a NOTIFY Response message to MN-CSE.

11.4 Possible impacts on oneM2M TSs

This clause concludes the study of oneM2M and OPC-UA interworking and proposes the possible impacts to oneM2M current specifications based on the consideration of the focus of oneM2M. The possible impacts may be addressed for resource/procedure mapping of OPC-UA interworking and semantic support for the industrial domain.

Based on the OPC-UA interworking study from this chapter, possible impacts on oneM2M TSs may lead to the corresponding transparent interworking developements, semantic enhancements, and security aspects.

– OPC-UA transparent interworking would be developed as the next phase of TR. Target TS could be [b-oneM2M TS-0003] proximal IoT interworking or a new TS.

  To support OPC-UA resource model mapping as described in clause 11.3.3, <flexContainer> needs to be further specified in the TS and summarized within [b-ITU-T Y.4500.1] clause 9.6.1.2.2 including opcuaView, opcuaObject, opcuaObjectType, etc. As a conclusion from clauses 11.3.3 and 11.3.4, OPC UA Method and ReferenceType nodes should be mapped to <contentInstance> resource.

  Simultaneously, towards OPC-UA procedure mapping in clause 11.3.4, new messages format of each discovery, simple reading, subscription and notification service procedure would be provided in the TS. Mappings should be defined in details, e.g., OPC UA Views mapped to oneM2M <accessControlPolicy>, OPC UA queue attributes mapped to oneM2M CMDH buffer.

– Moreover, semantic enhancements could be proposed as industrial domain information model. As a matter of fact, industry equipments include a large number of static devices (e.g., boilers, pipes, valves) and dynamic machines (e.g., compressors, pumps, pressure filters, disintegrators), and it would be difficult to list them all. And OPC UA holds companion specifications with ISA-95 at the enterprise and manufacturing systems level, which defines the terminology and information and operations models used in the integration of business systems e.g., MES, ERP, SCM, and AutomationML/PLCopen at the supervisory and control level, which are the commonly used data models in SCADA, HMI, PLCs for industrial automation, as shown in Figure 11.4-1 both for intra-factory and inter-factory scenarios. Hence, it is suggested for oneM2M to analyze data models from abovementioned different industrial standards, and propose semantics for industrial data collection and data analysis.

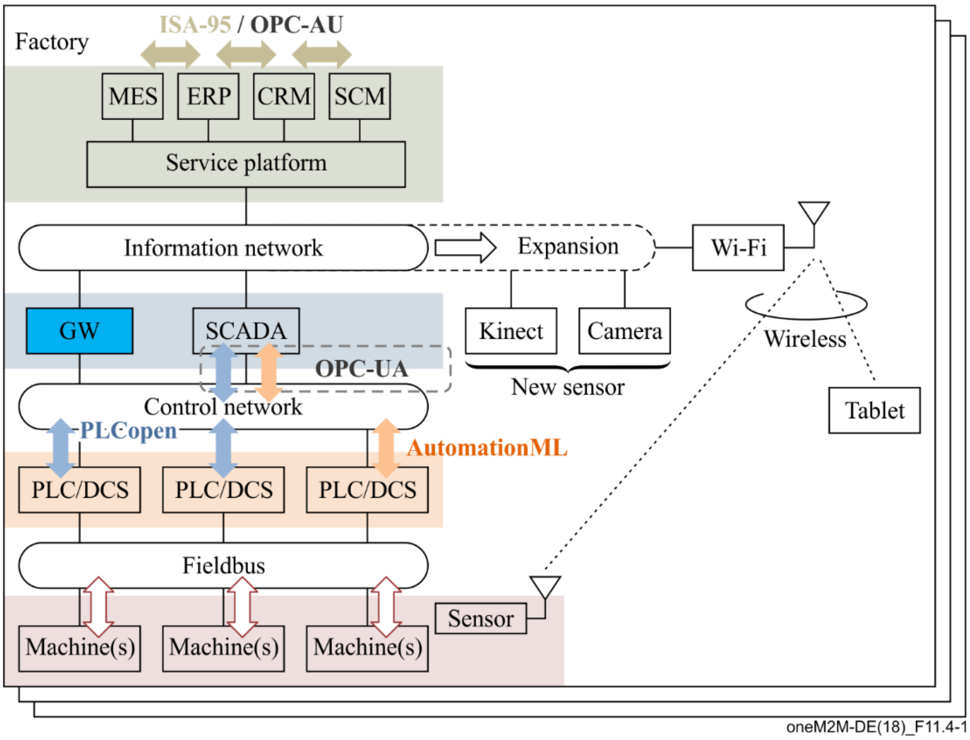


Figure 11.4-1 – OPC UA elements for semantics considerations

– Security would be another area for further study. As shown in Figure 11.1.5-2, OPC UA security architecture is composed of secure session and secure channel. Although secure session could be achieved by oneM2M existing AEs and CSEs, it is still necessary to consider from oneM2M side how to support secure channel establishment, e.g., to convert broken OPC UA pieces into understandable security messages (e.g., credentials to be treated as UA binary payload for oneM2M protocol) possibly on CSE.

12 Conclusion

The use cases of the industrial domain mainly include the communication and interaction of intra-factory and inter-factory scenarios, in which, effective collaboration between factories is achieved based on the connectivity provided by M2M technologies, and collected field data from all factories is used to make accurate decisions and timely responses. These use cases need the oneM2M system to support the requirements such as collecting field data from factories and supporting new data types (e.g., time series data), monitoring the status of underlying network to satisfy the QoS of applications, and classifying application data into various security levels.

To support the above use cases and requirements of industrial domain by oneM2M common service layer, industrial domain systems are integrated with oneM2M architecture. Additionally, the enhancement of oneM2M architecture shall be considered, such as introducing new resource types for implementing time series data and enhancing existing reference between CSEs of different ASNs to support peer-to-peer communication for manufacturing requirements. These functionalities need to be taken into account in future oneM2M specifications in order to deploy industrial service based on oneM2M system.

Additionally, specific security requirements are summarized. For the best practice of deploying industrial services, the enhancement of existing security solutions shall be considered in future oneM2M specifications, such as supporting end to end security and classifying application data into various security levels.

Bibliography

[b-ITU-T X.1362] Recommendation ITU-T X.1362 (2017), *Simple encryption procedure for Internet of things (IoT) environments*.

[b-ITU-T Y.4500.1] Recommendation ITU-T Y 4500.1 (2018), oneM2M - Functional architecture.

[b-ITU-T Y.4500.2] Recommendation ITU-T Y 4500.2 (2018), oneM2M - Requirements.

[b-ITU-T Y.4500.11] Recommendation ITU-T Y 4500.11 (2018), oneM2M Technical Specification TS-0011, Common Terminology.

[b-IEC 62443] IEC 62443-series, *Industrial communication networks – Network and system security*.

[b-IEC TC] IEC TC, *News*.
<http://www.iec.ch/tcnews/2014/tcnews_0214.htm>

[b-IEC TC 65] IEC TC 65, *Industrial-process measurement, control and automation*.

[b-IEEE P2413] IEEE P2413 website.
<http://grouper.ieee.org/groups/2413/>

[b-IEEE P2413 presentation] IEEE P2413 presentation (2014), *Standard for an Architectural Framework for the Internet of Things (IoT).*

[b-IEEE P2413 report] IEEE P2413 report (2014), *oneM2M Specification Comment Collection*.

[b-oneM2M DR] oneM2M DR, *Drafting Rules*.
<http://www.onem2m.org/images/files/oneM2M-Drafting-Rules.pdf>

[b-oneM2M TS-0003] oneM2M Technical Specification TS-0003, *Security Solutions*.

[b-Bosch's Blog] *Article First European testbed for the Industrial Internet Consortium in Bosch's ConnectedWorld Blog*.
<https://blog.bosch-si.com/industry40/first-european-testbed-for-the-industrial-internet-consortium/>

[b-IIC Engineering] IIC document Engineering (2014), *The First Steps*.
<https://www.iiconsortium.org/pdf/IIC_First_Steps_2014.pdf>

[b-IIC Engineering Update] IIC report Engineering Update (2014).
<https://www.iiconsortium.org/January-2015-IIC-Progress-Report.pdf>

[b-IIC website] IIC website.
<http://www.iiconsortium.org/>

[b-Industrie 4.0] *Reference Architecture Model Industrie 4.0,* (2015) *(RAMI4.0).*
<https://www.vdi.de/fileadmin/vdi_de/redakteur_dateien/gma_dateien/5305_Publikation_GMA_Status_Report_ZVEI_Reference_Architecture_Model.pdf>

[b-NIST (SP)800-57] NIST Special Publications (SP)800-57 (2014), *Guidelines for Derived Personal Identity Verification (PIV) Credentials*.

[b-OPC Classic] OPC technologies: OPC Classic.
<https://opcfoundation.org/about/opc-technologies/opc-classic/>

[b-OPC UA for ISA-95] OPC Unified Architecture for ISA-95 Common Object Model Companion Specification Release 1.00, Oct 2013. Available at:
<https://opcfoundation.org/developer-tools/specifications-unified-architecture/isa-95-common-object-model/>

[b-OPC UA Part 4] OPC UA Part 4 – Services 1.03 Specification, July 2015.

[b-OPC UA Part 5] OPC UA Part 5 – Information Model 1.03 Specification, July 2015.

[b-OPC UA website] OPC technologies, OPC UA.
<https://opcfoundation.org/about/opc-technologies/opc-ua/>

[b-SMLC presentation] SMLC presentation (2014), *Smart Manufacturing: Enterprise Real-Time, Networked Data, Information & Action*. Available at: <https://pdfs.semanticscholar.org/presentation/9155/8d1b80dda52dd3164154b7e79009e88d0024.pdf>

[b-SMLC website] SMLC website. <https://www.smartmanufacturingcoalition.org/faq/>