Factory-X Use Cases

Application Guide

Factory-X

The Digital Ecosystem

Contact: info@factory-x.org

IMPRINT

Publisher

Factory-X
https://factory-x.org
info@factory-x.org

Project Office Factory-X

Open Industry 4.0 Alliance Implementation GmbH
Willy-Brandt-Platz 6
81829 Munich, Germany

Version History

August 12th, 2024	V1.1	– Initial version including technical view
September 2nd, 2024	V1.2	– Purpose and scope, System integration considerations
March 12th, 2025	V1.3	– Change of Figure 8 and layout
April14th, 2025	V1.4	– Replace FX Port by MX-Port, delete3.5

Authors

ET cross use cases: Ulrich Löwen, Siemens
UC1: Johannes Hoos (Festo), Mirko Löffler (Siemens), Kevin Diebels (Trumpf), Björn Sautter (Festo)
UC2: Johannes Kalhoff (PhönixContact), Frank Wolter (Siemens)
UC3: Christoph Attila Kun (BASF), Carl-Philipp Ding (BASF)
UC4: Fernando Martinez Calderon (Siemens), Smitha Rayala (SAP), Joshi Nirav (DMG MORI)
UC5: Markus Lieret (Siemens)
UC6: Birgit Obst (Siemens)
UC7: Klaus Bauer (Trumpf), Nils Schacht (inovex), Marcel Elygomes (Trumpf)
UC8: Plamen Kiradjiev (German Edge Cloud)
UC9: Marc-Andre Dittrich (DMG MORI)
UC10: Lucas Fochler (Estainium)
UC11: Dominik Rohrmus (Siemens)

PART OF MANUFACTURING-X

Funded by the European Union NextGenerationEU

Supported by: Federal Ministry for Economic Affairs and Climate Action on the basis of a decision by the German Bundestag

Table of content

1 General

Factory-X will develop a **digital ecosystem**. In the understanding of the Factory-X use cases this digital ecosystem is a socio-technical system in which various stakeholders (companies, technical systems, people, associations, etc.) work together to achieve an overarching objective. The overarching objective is the vision of Factory-X (according to the GVB): a new way of working together to provide the best production facilities that enable factory operators to produce the best products for their customers.

The use cases develop three different **views** on the digital ecosystem Factory-X as socio-technical system:

- **Business view**: this view addresses the collaboration of legal entities. This collaboration is subject to certain rules that are developed per use case in collaboration with TP3 and TP5. The PMs and POs of the Factory-X use cases are responsible to develop the business view.
- **Usage view:** this view considers concrete software applications that exchange data. These software applications are used to achieve business goals. Factory-X considers the use cases outlined in the GVB. These usage views lay the foundation for the requirements to be considered by TP4. The PMs and POs of the Factory-X use cases are responsible to develop the usage view.
- **Technical view**: this view considers data exchange between software applications using technical concepts provided by TP4, so called shared services. The requirements for these shared services are developed by the use cases in collaboration with TP4. The shared services are designed and implemented prototypically by TP4, and the use cases will use these implementations to implement and validate their use cases. To develop the technical view in the use cases, close collaboration between the PMs and POs on the one hand and the TAs and SAs on the other hand is necessary.

2 Purpose and scope

This document is intended to establish and document a common understanding of the objectives, and the prototypical implementation based on demonstrators within the Factory-X use cases. For this purpose, an application perspective is taken. The document describes (logical) concepts and is agnostic regarding technical implementations and implementation technologies. The aim of the document is to be understandable even for non-software architects.

In the course of the project, it may happen that some of the terms defined, and concepts described in this document will also be discussed outside of the Factory-X use cases. It must then be ensured that these different considerations are consistent with one another. In case of an inconsistency, it is necessary to take concrete measures to resolve these inconsistencies within the overall project.

The scope of the document refers to content that is planned in the individual use cases in accordance with the GVB. Other aspects that are of interest across use cases may be added later.

3 Terms and definitions

3.1 Business application

Software application using shared services

The scope of a business application is defined by a Factory-X use case.

A business application is designed and developed in a Factory-X use case by one or more business application provider and used in a Factory-X use case by one or more business application user. It is designed for easy system integration to interact with legacy applications and other business applications. To operate a business application, a business application operator is necessary. The business application operator can be the business application provider or the business application user or another 3rd party company.

app Icon used in usage views: where “app” is the name of the business application.

3.2 Legacy application

Software application not using shared services

A legacy application is neither designed nor developed in a Factory-X use case. It is enabled by system integration to interact with business applications and other legacy applications without the need for internal changes to the legacy application. To operate a legacy application, a legacy application operator is necessary.

app Icon used in usage views: where “app” is the name of the legacy application

Note: As long as shared services are not yet precisely defined, the characterization of whether a specific software application is a business application or a legacy application can still change.

Note: These definitions for business and legacy applications still leave freedom to whether a software application is considered a legacy application or a business application. The Factory-X use cases are free to use this freedom for their purposes.

3.3 Shared service

Software functionality provided through interfaces

A shared service is designed and developed by Factory-X TP4.

A shared service is operated by a shared service operator (operating company) and can be called by different business applications. A purpose to design and develop shared services is the reuse across use cases.

A shared service can be provided by a shared service provider in compliance with the requirements of TP 4 and the governance body.

Shared services are not considered neither in the business nor usage view, but in the technical view.

3.4 Capability

This term is initially not used by Factory-X use cases. Some use cases use the term in the GVB. Each use case decides whether in a first step a capability is considered as a shared service or a business application.

4 Business view of Factory-X use cases

4.1 Value network

Network of business roles and value creation relationships between business roles

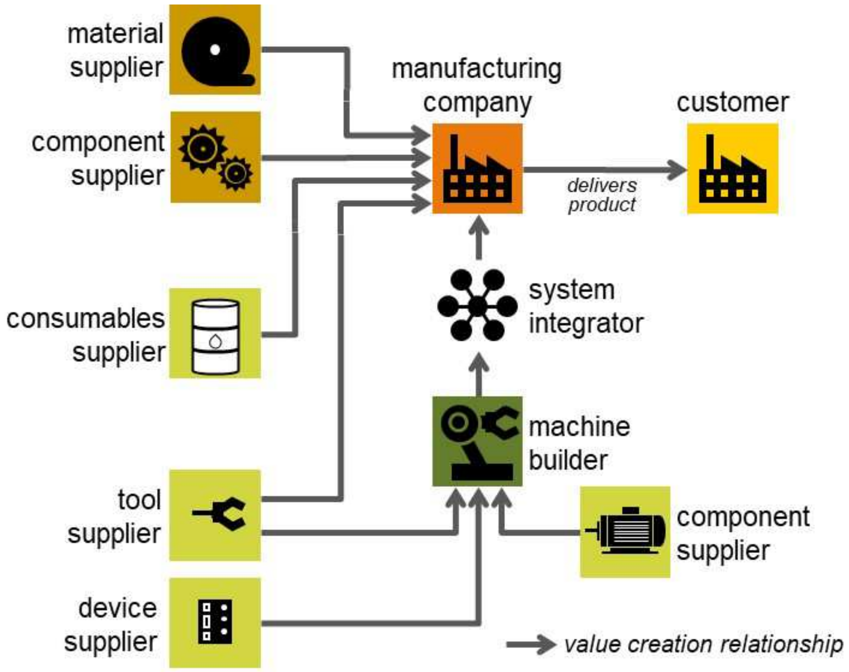

Figure 1: Example of a value network

From the perspective of a manufacturing company there are two different value chains:

- Value chain regarding the **product** of the manufacturing company, which includes all deliveries from suppliers that are **integrated** into the manufacturing company's product. These suppliers are illustrated by brown colored icons, the manufacturing company by an orange-colored icon and the customer of the product by a yellow-colored icon. The consideration of this value chain is the application scope of Catena-X with focus on automotive industries.
- Value chain regarding the **production system** of the manufacturing company, which includes all deliveries from suppliers that are needed to **build** and **operate** the manufacturing company's production system. These suppliers are illustrated by green or yellow green icons. Factory-X expands the application scope of Catena-X by this value chain. A key driver regarding this supply chain are new business models for machine builders and component suppliers, e.g., additional revenue streams during operation of a machine or a component.

A value creation relationship is a directed relationship between two business roles. A value creation relationship typically includes a value proposition of one business role for which typically the other business role is willing to pay.

The value creation relationships should be labeled. In Figure 1 an example is shown for the value creation relationships between the manufacturing company and its customers, which is labeled by “delivers product”.

4.2 Business role

Set of characteristics of a legal entity to exhibit a set of required behaviours

Note: a specific legal entity can assume different business roles

Examples for business roles:

- Manufacturing company, machine builder, supplier and customer, which typically also assume the role business application user
- Business application provider, business application operator, business application user, which are not shown in Figure 1

- Shared services provider, shared service operator, governance body, certifier, which are not shown in Figure 1 and currently not considered by the Factory-X use cases
- System integrator, tool provider, and development services providers, which are supportive roles, and which are only partially shown in Figure 1.

Manufacturing company: Legal entity being responsible for the design, development, and manufacturing of a physical product with a view to bringing it to market

Icon used in value networks:

Machine builder: Specific manufacturing company, where the physical product is a machine

Icon used in value networks:

Supplier: Specific manufacturing company, where the physical product is a component

Icons used in value networks, if the delivery of the supplier is integrated into a manufacturing company's product:

Icons used in value networks, if the delivery of the supplier is needed to build and operate the manufacturing company's production system:

Customer: Legal entity requiring a physical product in exchange for money or other resources

Icon used in value networks:

Business application provider: Legal entity providing or licensing a business application commercially

Icon used in value networks:

Business application/legacy operator: Legal entity operating a business/legacy application commercially

Icon used in value networks:

Business application user: Legal entity operating a business application commercially

Icon used in value networks:

Data provider: Legal entity providing data commercially

Icon used in value networks:

Service provider: Legal entity providing services

Icon used in value networks (depending on the provided kind of service, for example, consulting services, data driven services, hybrid services, physical services):

Shared service provider: Legal entity providing or licensing a shared service commercially

Icon used in value networks:

Shared service operator, operating company: Legal entity operating a shared service commercially

Icon used in value networks:

Governance body: Legal entity specifying the functionality and compliance of shared services, specifying the compliance of business applications, and certifying certifier

Icon used in value networks:

Certifier: Legal entity creating a certificate for shared services and business applications regarding compliance

Icon used in value networks:

System integrator: Legal entity supporting a business role regarding system integration of products and solutions of other business roles

Icon used in value networks:

Tool provider: Legal entity providing a software tool to develop a software application

Icon used in value networks:

Development services provider: Legal entity supporting a business application provider or shared services provider in the development of the business applications or shared services

Icon used in value networks:

Integration services provider: Legal entity supporting another legal entity in the integration of business applications, legacy applications or shared services

Icon used in value networks:

Electrical grid operator (specifically for UC 2.9): Legal entity operating an electrical grid

Icon used in value networks:

Recycler (specifically for UC 2.11): Legal entity providing the recycling of a physical entity

Icon used in value networks:

4.3 Example of a business view

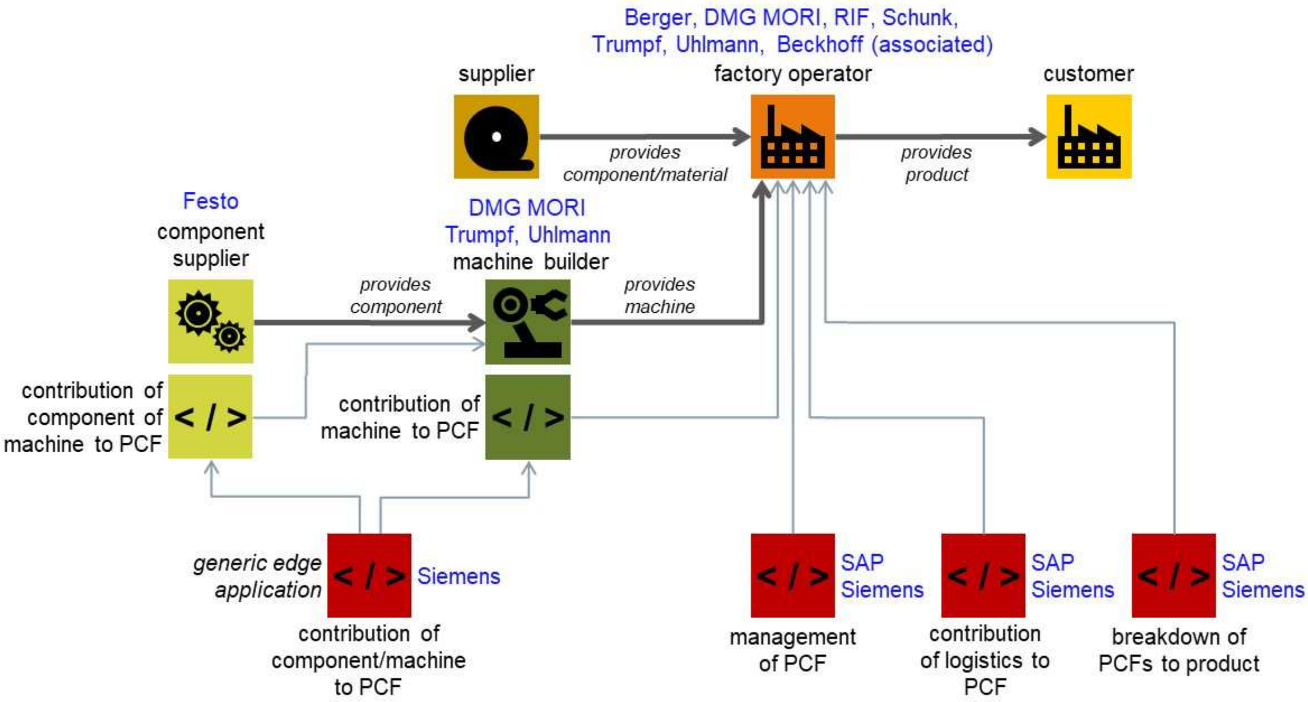

Figure 2: Example of a business view

Each light grey arrow in Figure 2 is labeled with “provides business application”.

The colored icons including their black labels are the business roles, which are assumed in the specific Factory-X use case by the legal entities illustrated with blue text.

5 Usage view of Factory-X use cases

Description of the interaction of business applications, legacy applications, and further actors, for example, humans, machines, components of machines

5.1 Description of the basic elements of a usage view

In a first step, the following basic elements are described:

- Considered business and legacy applications
- Data flow (or more generally: interactions) between two business and legacy applications. If the source or destination of a data flow is not known or not relevant, a dummy legacy application will be introduced.
- Each business and legacy application is associated to a business role, which operates the software application, in the business view. This is illustrated by a dashed line that includes all software applications operated by the same business role. This dashed line is annotated with an icon of the corresponding business role.

  In a first step, no distinction is made between operating a software application by its own and using software as a service.

  In addition, in a first step, nested business applications are not considered.
- If there is a data flow (or more generally: interaction) between two software applications, which are operated by the same business role, the data flow is illustrated by a red arrow: ↔
- If there is a data flow (or more generally: interaction) between two business applications, which are operated by different business roles, the data flow is illustrated by a blue arrow: ↔ If there is a data flow (or more generally: interaction) between business applications operated by different business roles, there is typically a value creation relationship between these two business roles in the usage view. Legacy applications should not be involved in cross-company data exchange.
- The value creation relationships between the business roles according to the business view can be integrated into the illustration of the usage view.

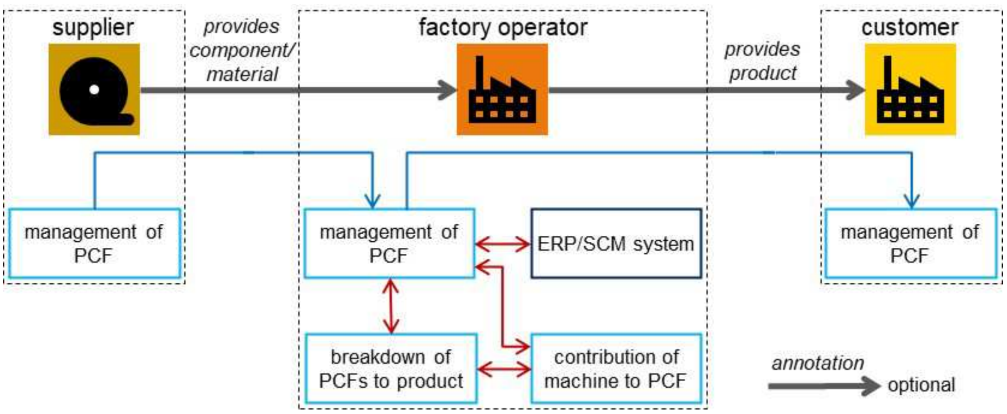

Figure 3: Example describing the basic elements of a usage view

5.2 Extension of description of basic elements of a usage view

Based on the basic description as illustrated in Figure 3 the following extensions are added to the description of the usage view:

- The data flow (or more generally: interaction) between software applications will be annotated by using yellow bordered rectangles, into which an explanatory text can be integrated. The following aspects should be considered when annotating:
  - Description of payload, i.e., data which is exchanged. It is recommended to structure this payload based on assets, i.e., to associate the information exchanged between software applications to assets.
  - Characteristics of data exchange, e.g., amount of data, frequency
  - Requirements regarding data sovereignty, e.g., data exposure, data provenance, usage control

  Typically, all this information cannot be integrated into the graphic but will be rather described in separate text blocks.
- The software applications will be annotated by using a cyan bordered rectangle indicating, whether the software application is deployed on cloud, on premise or on edge.
- Furthermore, for each software application the following aspects should be described in separate text blocks:
  - Purpose of the software application in 1-2 bullet points
  - Functionality of the software application in 5-10 bullet point. The focus is on the software application itself, not on the overall behavior of the software application in its context.
  - Interoperability/portability requirements in 2-5 bullet point addressing the following question: what does it mean from a user perspective when different legal entities offer the same business application?
- There can be integrated appropriate additional elements, e.g., human actors or machines, which interact or exchange data with a software application.

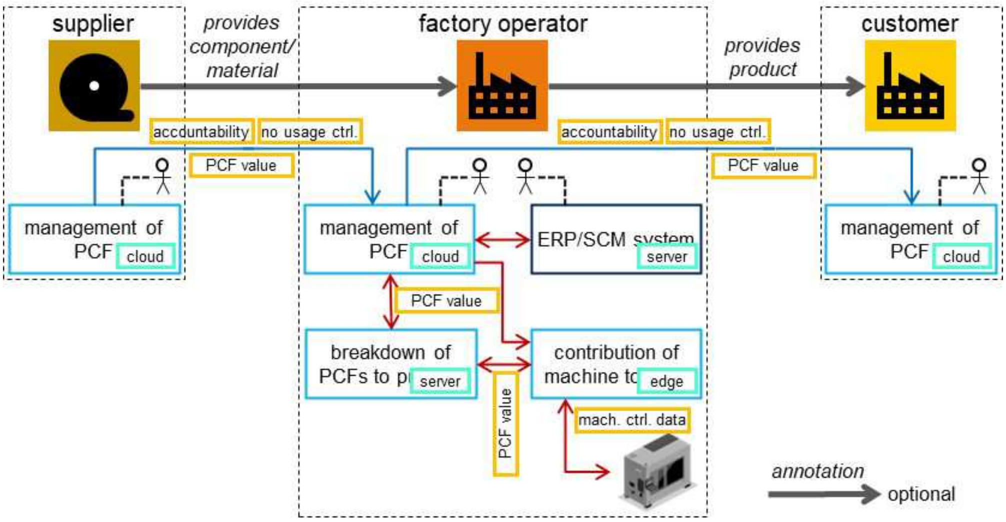

Figure 4:Example of an extension of Figure 3

- When creating the usage view, different scenarios can be described, which should usually be illustrated in different figures. Examples are
  - In one scenario the interaction of software application during operation of the software applications can be described as shown in Figure 3 and Figure 4. In another scenario the system integration of software applications, i.e., the workflow, how the software applications are provided and integrated to a running system, can be described.

- Figure 3 illustrates a scenario, where the machine user is also the machine owner, whereas Figure 5 illustrates a scenario, where the machine user is not the machine owner.

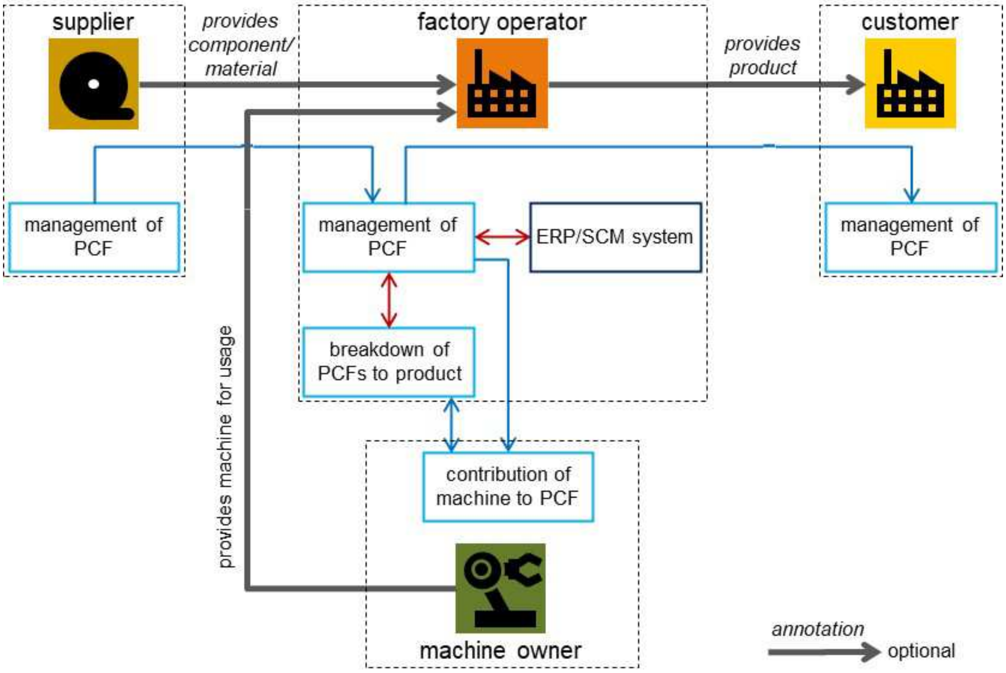

Figure 5: Scenario, where machine user is not machine owner

6 Quality Guidance for Business and Usage View

Creation of the views according to the guiding principles described in this document

The views describe a common view of all companies involved in an Factory-X use case

Note: Business and usage view described on a common methodology based on IEC 63283-2 Smart manufacturing – Use cases

7 Technical view of Factory-X use cases

The usage view is a high-level description of a use case and shows basic interaction relationships between business and legacy applications. The objective of the technical view is to describe these interactions in the form of concrete workflows and thereby detailing the requirements for shared services provided by TP4. The technical view will be developed iteratively, and the individual use cases will decide on the concrete approach, for example whether they first aim for good coverage of the scenarios or rather a breakthrough regarding the shared services of TP4. Note that the technical view should be distinguished from an implementation view, which is not discussed by the Factory-X use cases.

7.1 Step 1: Detailing the workflows

In a first step, the interactions between business and legacy applications described in the usage view are specified more precisely. This specification focuses on business logic and is thereby the task of the POs and PMs of the use cases; the TAs and SAs of the use cases do not necessarily have to be involved since technical architectures are not yet discussed here. This detailing includes the following design decisions for each interaction between a business/legacy application 1 and a business/legacy application 2, for illustration see Figure 6:

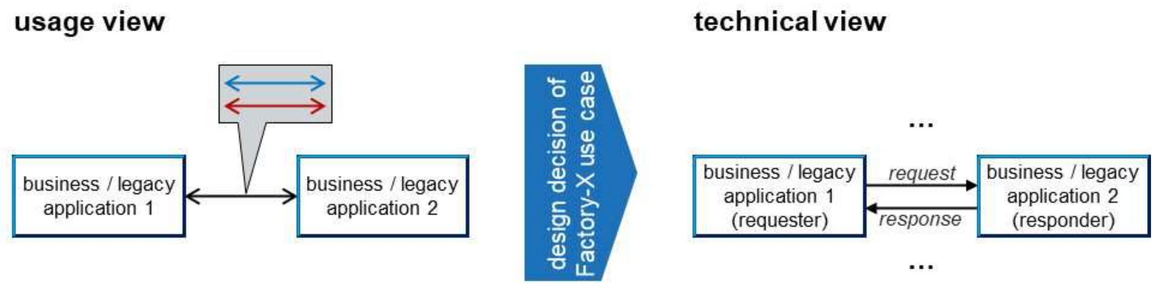

Figure 6: Breaking down an interaction in a sequence of transactions

- For business/legacy application 1 and business/legacy 2 there is an agreement on a **common** (logical) **object model**. Examples for such object models are (the object models of) ECLASS, AAS, OPC-UA, AutomationML, etc. There are currently no specifications in Factory-X as to how such a common object model should be formally described.
- Each interaction is broken down in **sequence of transactions.** Each transaction has a **requester** and **responder**, where the requester requests the execution of an operation “result = f(input, output)” by the responder (result, input and output follow the agreed common object model). Examples for operations are read, write, execute, create, delete, etc. As a consequence, business/legacy application 2 has to provide a set of operations result = f(input, output). Note that in one transaction of an interaction a business/legacy application can assume the role of a requester and in another transaction of the same interaction the role of a responder.
- Any other actors that may already be described in the usage view, such as human actors (personas) or technical entities (for example machines or sensors), must be completed.
- Any scenarios that may already be described in the usage view must be completed.

The POs and PMs of the use cases have agreed to create this description based on BPMN (business process model and notation, see [1]). In addition, due to the complexity of the models, it is recommended to create this description using a modeling tool (and not a “drawing” tool). The Factory-X use cases are methodically supported here by the consortium partner “Scheer”. In addition, the consortium partner “Scheer” provides a freely accessible modeling tool, for details see [2].

From a top-down view, it is recommended to structure the technical view in the form of scenarios. For each scenario, an organization should be made in pools and swim lanes. A pool should be created for each legal entity in the usage view (indicated by the dashed lines). A separate swim lane should be created for each interacting actor in the legal entity, for example human actor (persona), business/legacy application, technical entity, etc. For illustration see Figure 7.

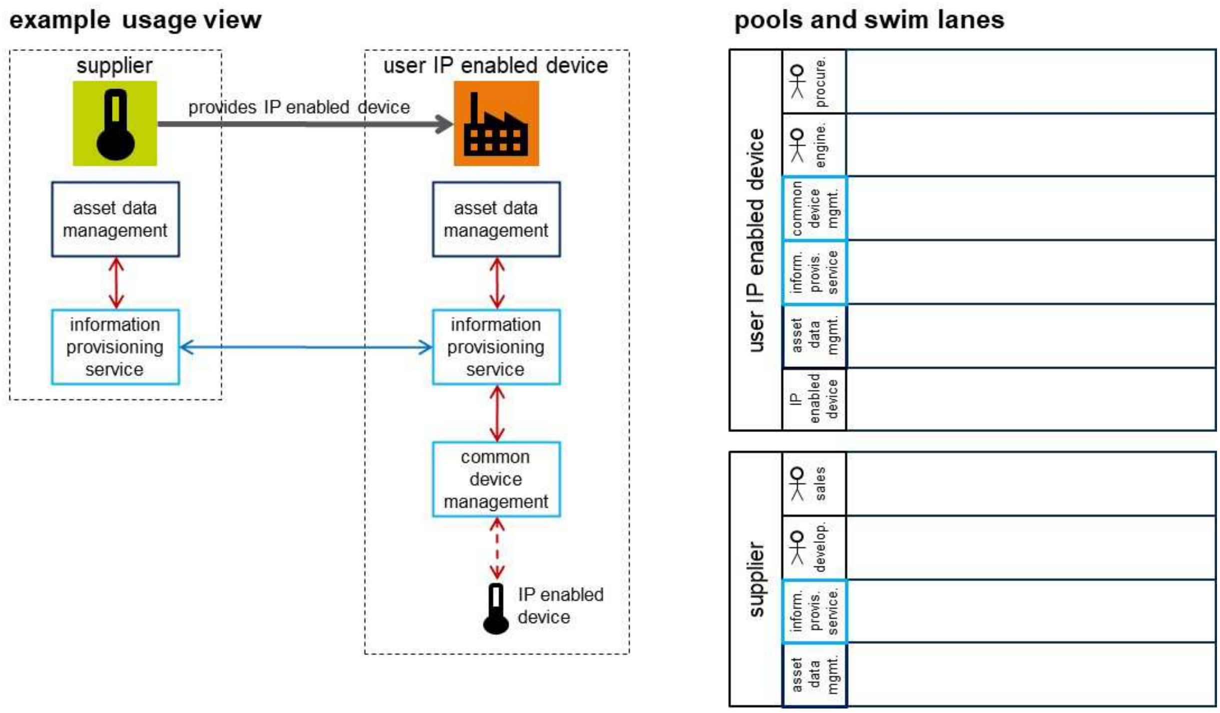

Figure 7: Organization in pools and swim lanes

7.2 Step 2: Agreement on a uniform implementation

The POs and PMs of the Factory-X use cases agreed on a design decision that interactions between business/legacy applications should be implemented in a uniform manner using a concept provided by TP4. Furthermore, legacy applications should be integrated in a uniform manner using a concept provided by TP4.

TP4 will develop such a concept, describe how to apply the concept, and provide prototypical implementations of the concept. This concept is named **MX-Port**.

Figure 8 illustrates the application of MX-Port in the technical view of the use cases.

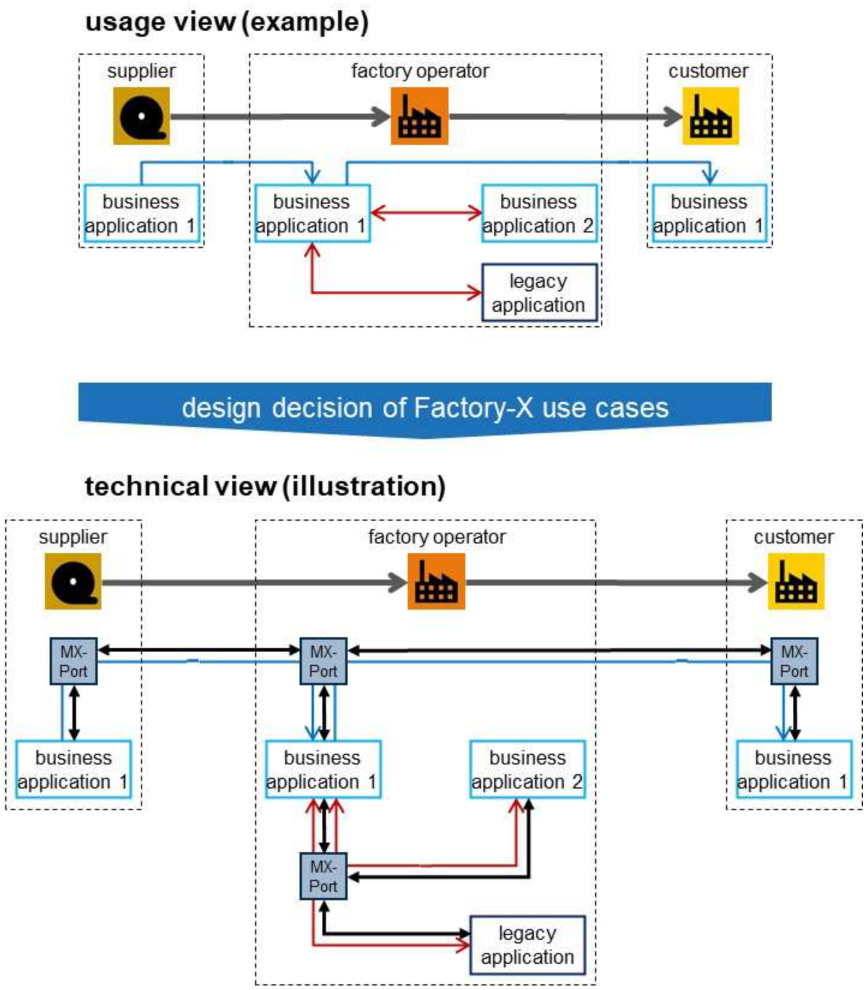

Figure 8: Illustration of application of MX-Port in technical view of use cases

The following guiding principles shall apply:

- The PMs and POs of the Factory-X use cases are free to implement selected individual interactions specifically without using MX-Port. This will be represented in the usage view with dashed arrows. Figure 9 illustrates an example, where the PO of an Factory-X use case decided to implement the interaction between the business application and machines individually without using MX-Port.

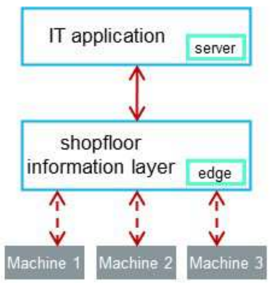

Figure 9: Illustration of an individually implementation of an interaction

- A legal entity is free in the usage of MX-Port, for example using only one or several MX-Ports.
- There could be different MX-Ports with different functionality, for example a MX-Port without data usage control for data exchange within a company. This will be detailed during the Factory X project in alignment with TP4.
- An MX-Port can be implemented using different technologies, for example EDC, AAS server, OPC-UA server.

- A business application can integrate an MX-Port. It is the responsibility of the PMs and POs of the Factory-X use cases to define the added value of a business application compared to an MX-Port. Note that TP4 governs MX-Port. Figure 10 illustrates design alternatives for using MX-Port.

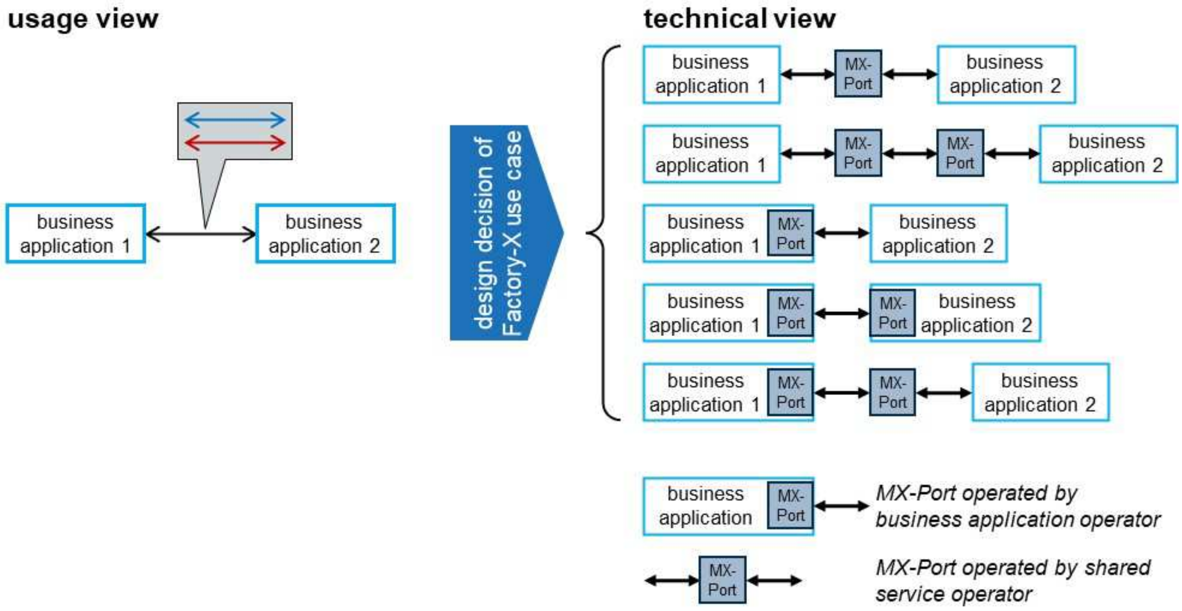

Figure 10: Illustration of design alternatives for using MX-Port

7.3 Step 3: Detailing regarding MX-Port

Figure 11 details the MX-Port concept illustrating a transaction between a requester and responder based on MX-Port.

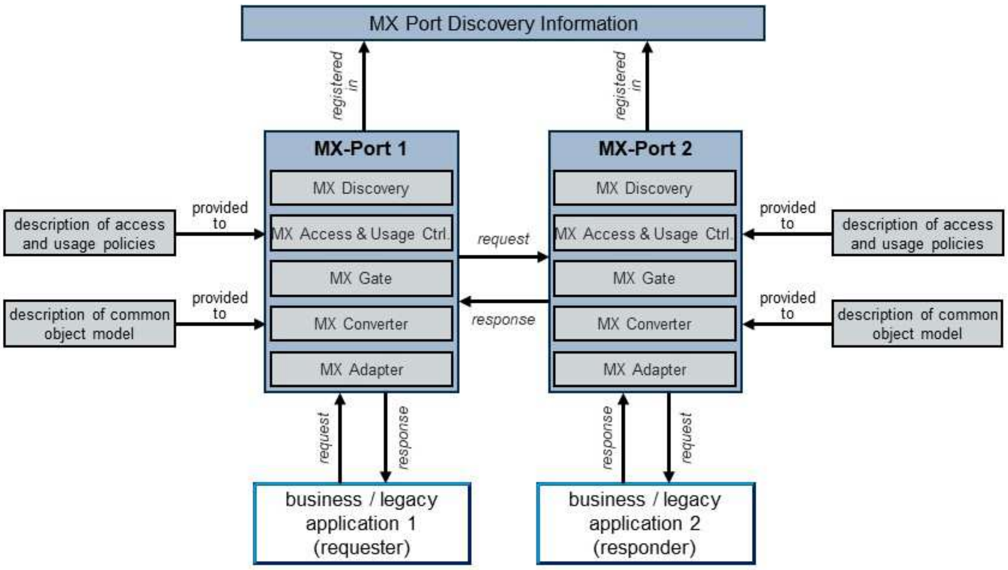

Figure 11: Overall conceptual structure of MX-Port

- MX Discovery: The purpose is to unify the discovery of interfaces.
    - MX Discovery of an MX-Port provides capabilities to discover another MX-Port. MX-Port 2 must be registered in an MX Port Discovery Information (operated by some (logical) operating company), where it can be found by MX-Port 1. MX-Port 1 does not necessarily have to be registered in the same MX Port Discovery Information.

- MX Port Discovery Information can be implemented in different ways. Factory-X must be open in this regard.

- MX Access & Usage Control: The purpose is to unify access and usage control of interfaces.
    - MX Access & Usage Control of an MX-Port provides capabilities to define access & usage control policies for the MX-Port. For each MX Access & Usage Control of an MX-Port the governing organization must define the access and usage policies.
    - Factory-X must be open regarding the implementation of access and usage control.
- MX Gate: The purpose is to unifies access to information.
    - MX Gate of an MX-Port provides a set of operations *result = f(input, output)*. Business application must specify the operations of its MX-Port. Maybe that the implementation of *f* requires internally a conversion to some other object model or a connection to some internal adapter or protocols, which is the purpose of MX Converter and MX Adapter.
    - Each use case defines for each MX-Port, which operations *f* are conceptually required to be provided by MX Gate of the corresponding MX-Port
- MX Converter: The purpose is to unify the representation of information.
    - MX Converter of an MX-Port provides capabilities to convert information from one representation into another representation.
- MX Adapter: The purpose is to adapt to the specific interfaces of a business or legacy application.
    - MX Adapter of an MX-Port provides a standardized interface for the specific capabilities of a business or legacy application.

Figure 12 illustrates the workflow of a transaction between a requester A1 and a responder A2, where A1 requests for execution of an operation *f(input, output)*, which returns a result *result* of A2:

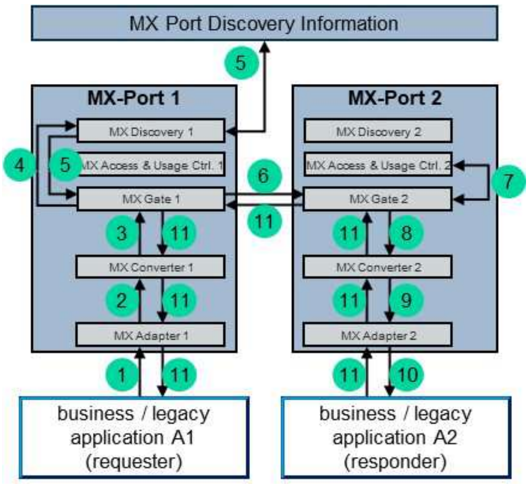

Figure 12: Illustration of workflow of a transaction

1. A1 requests for execution of operation *f* of A2, which is manifested by some A1 specific request
2. MX Adapter 1 transfers the specific request of A1 to interface of MX Converter 1 and invokes operation *f* of MX Converter 1, which is manifested by $f_{Converter\ 1}$ of MX Converter 1. This also includes that MX Adapter 1 provides MX Converter 1 with all required *input* in a specific format $input_{Converter\ 1}$.
3. MX Converter 1 converts $input_{Converter\ 1}$ to information *input* of operation *f* as agreed between MX Gate 1 and MX Gate 2 in the common (logical) object model, provides *input* to MX Gate 1, and invokes operation *f* of MX Gate 1.

4. MX Gate 1 uses MX Discovery 1 to request for MX Port 2.
5. MX Discovery 1 looks up in MX Port Discovery Information and returns MX-Port 2 to MX Gate 1.

If no is MX-Port 2 found, the request of A1 is canceled, otherwise for all MX-Ports 2 found the following steps are executed:

6. MX Gate 1 provides *input* to MX Gate 2 and uses MX Gate 2 to request for access to operation *f* of MX Gate 2.
7. MX Gate 2 uses MX Access & Usage Control 2 to enforce access & usage control of requested operation *f*.
8. If approved, MX Gate 2 provides *input* to MX Converter 2 invokes operation *f* of MX Converter 2, otherwise the request of A1 is canceled.
9. MX Converter 1 converts *input* of operation *f* to $input_{Adapter\ 2}$ supported by A2 and invokes operation **f** of MX Adapter 2 (manifested by $f_{Adapter\ 2}$ of MX Adapter 2).
10. MX Adapter 2 transfers request to interface of A2 and invokes operation *f* of A2 (manifested by some A2 specific operation).
11. Execution of operation *f* of A2 returns *result* and *output* (via corresponding conversions) to A1.

Figure 13 shows a special case where business applications 1 and 2 are operated by the same software application operator and where in step 5 in Figure 12 the request from MX Gate returns MX Gate as the result.

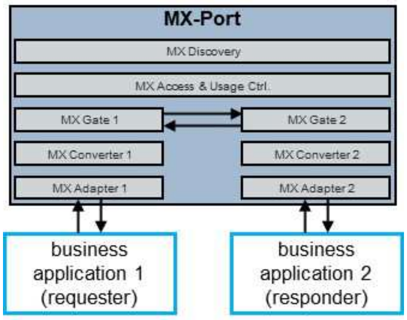

Figure 13: Illustration of special case

An MX-Port can serve multiple requesters or multiple responders, because the requester specifies $f_{BA\ responder\ xy}$ and thus determines which responder is requested. For illustration see Figure 14.

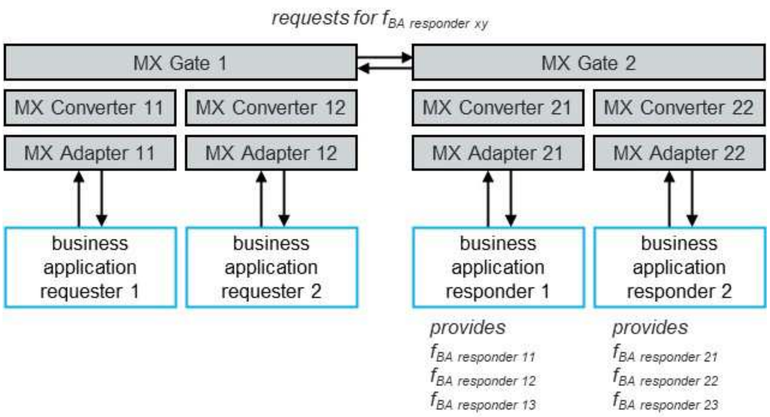

Figure 14: Illustration of serving several requesters and responders

7.4 System integration considerations

A business application provider can decide to integrate an MX-Port (or parts of an MX-Port) into a business application, which is illustrated in Figure 15. Parts of an MX-port within a business application are shown in "white", parts of an MX-port outside a business application are shown in "gray". The parts of an MX-port within a business application are **logical** objects (and not implementations; implementation aspects are not considered in the technical view). This means that a business application that integrates constituents of an MX-port only needs to provide an interface compliant to the MX-Port on its north side. For system integration, the parts outside of the MX-port must be provided by a 3rd party.

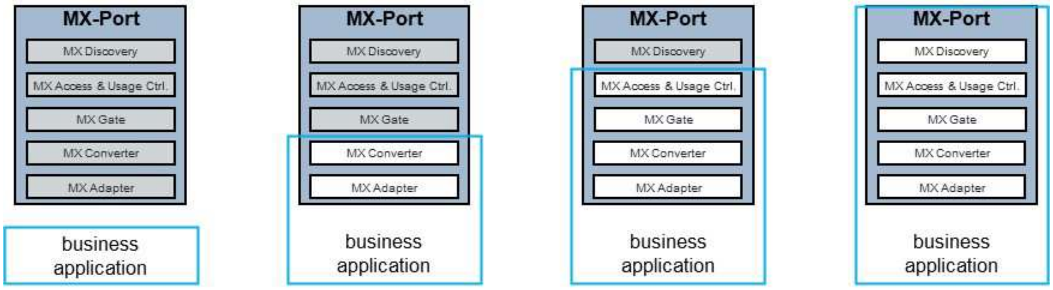

Figure 15: MX-Port can be (partly) integrated in a business application (examples)

A legacy application shall provide capabilities for system integration. A legacy application may include an MX Adapter or MX Converter, which is illustrated in Figure 16; if it also includes an MX Gate, it should be considered a business application. The parts of an MX-port within a legacy application are **logical** objects (and not implementations). For system integration, the parts outside of the MX-port must be provided by a 3rd party.

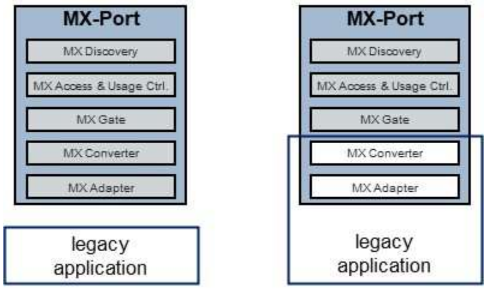

Figure 16: MX-Port can be (partly) integrated in a legacy application (examples)

During system integration, it must be considered whether an interaction between business/legacy applications takes place across legal entities or within a legal entity, see Figure 17:

- In the case of an interaction across legal entities, a separate MX-Port is required within each legal entity. The MX-Port is necessary that the legal entity can define and enforce the own interests regarding the interaction. The MX-Port can be located outside the business/legacy application or, in accordance with the options illustrated in Figure 15 and Figure 16, can be partially or fully integrated into the affected business/legacy applications.
- In the case of an internal company interaction, one or two MX-Ports can be used. This is a design decision that the Factory-X use cases must make. Here, too, the MX-Port can be located outside the business/legacy application or, in accordance with the options illustrated in Figure 15 and Figure 16, can be partially or fully integrated into the affected business/legacy applications.

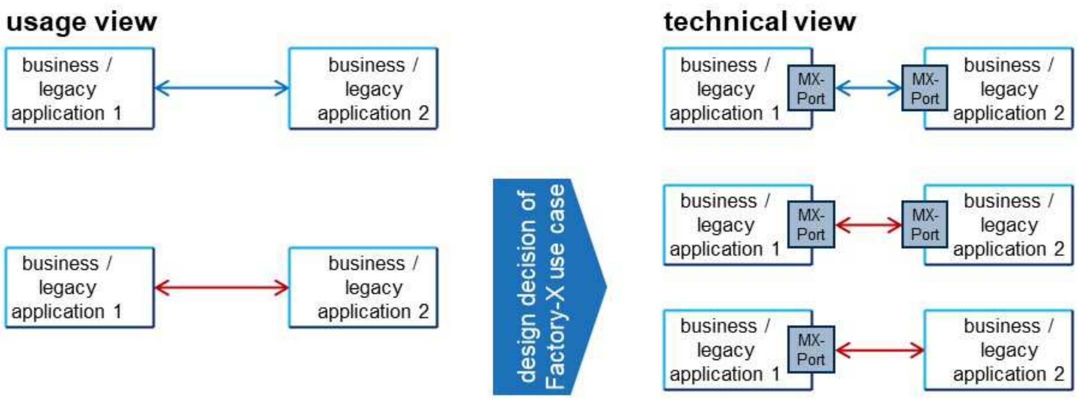

Figure 17: Distinction between cross-legal entity and intra-legal entity interaction

The variants of system integration shown in Figure 17are each illustrated by an example in Figure 18.

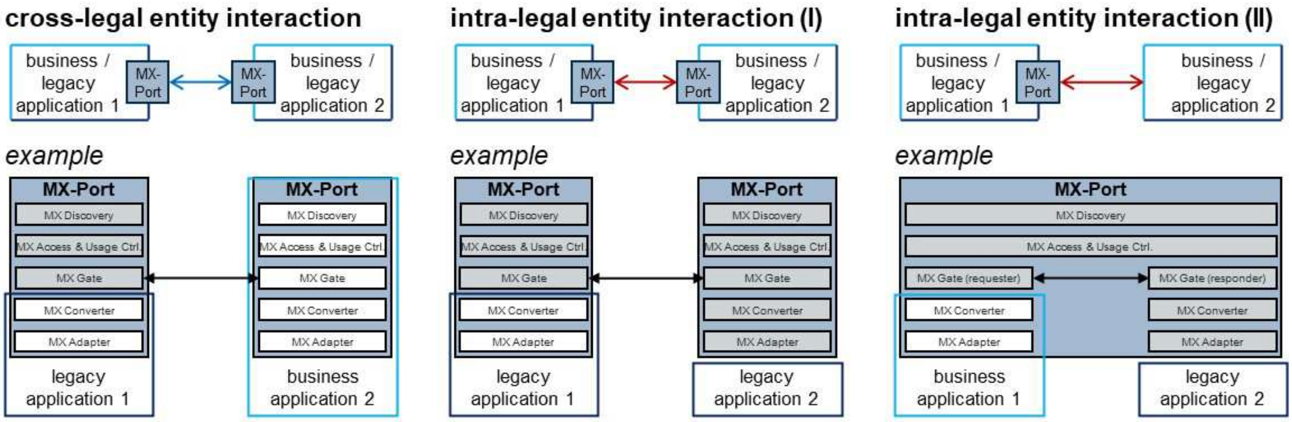

Figure 18: Illustration of variants of system integration using MX-Port

From the perspective of the business or legacy application, this results in the following options for system integration illustrated in a comparison in Figure 19:

- A legacy application is always connected via MX Adapter and MX Converter to the "south side" of an MX Gate, which is not part of the legacy application.
- A business application can be connected via MX Adapter and MX Converter to the "south side" of an MX Gate, which is not part of the business application.
- A business application can be connected via the own MX Gate to the "east/west side" of another MX Gate, which can be part of the business application or not

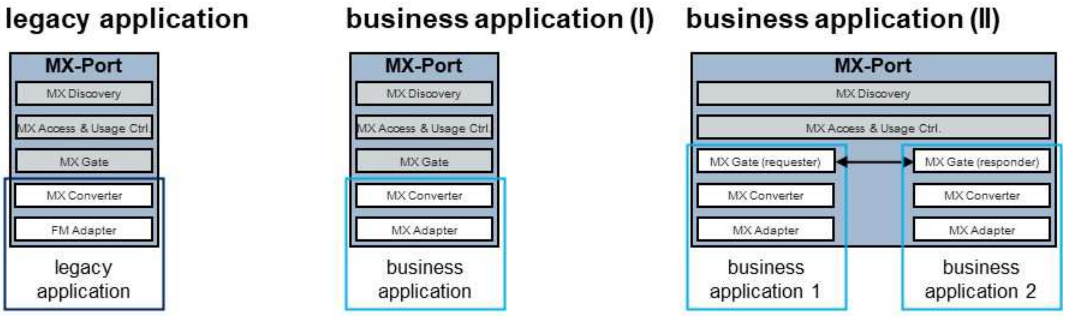

Figure 19: Illustration of options for system integration of business and legacy applications

7.5 Modularity regarding MX-Port

The MX-Port has the claim of modularity, both from a technical and a business perspective. This will be illustrated below using an example.

The example in Figure 20 shows a sensor supplier who, in addition to sensors, also offers an analysis application that evaluates measured sensor data. This analysis application is designed so that several sensors can be connected via IO-Link. The connected sensors can also be provided by another supplier of IO-Link connectable sensors. In addition, a higher-level IT application is considered, to which the analysis application provides selected analysis results.

Figure 20 also shows an example of a possible conceptual transformation of the usage view in a technical view. The sensors are considered as legacy applications that act as responders. They are therefore connected via an MX Adapter (IO Link) and MX Converter to a south-side MX Gate of the analyze application, which acts as responder. The specific business logic of the analyze application accesses the individual MX Gates from the left as shown in Figure 12 and thereby ensures that the sensor information is retrieved via the south-side of the MX Gates.

The analyze application itself also acts as responder towards the higher-level IT application. Therefore, the specific business logic of the analyze application provides the north-side MX Gate with the required information by feeding from south in the north-side MX Gate of the analysis application.

The higher-level IT application acts as requester. The specific business logic of this IT application feeds from the south in the MX Gate of the IT application.

Basically, the three MX-ports within the analyze application are configured by the operator of the analyze application and the MX-port of the higher-level IT application is configured by the operator of this application according to their needs.

MX Discovery and MX Access & Usage Control are not considered in more detail in this example.

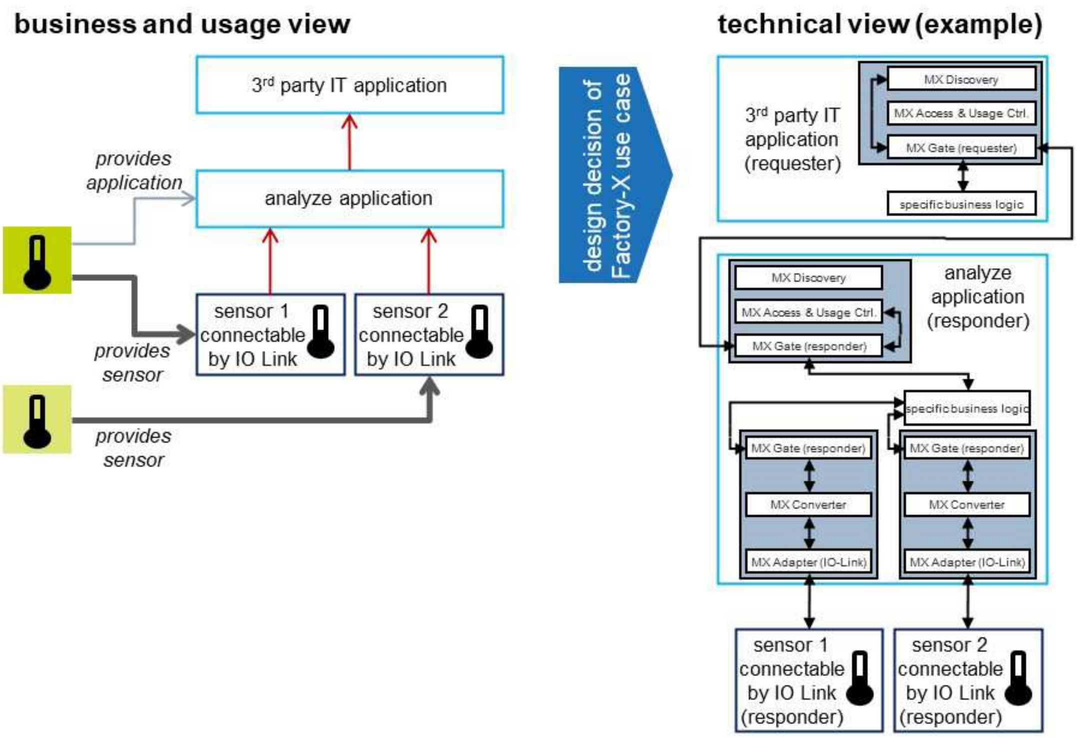

Figure 20: Modularity regarding MX-Port (example)

Figure 21 extends Figure 20 by integrating another sensor that is connected via OPC-UA. It is assumed that this is done by using an MX Adapter (OPC-UA).

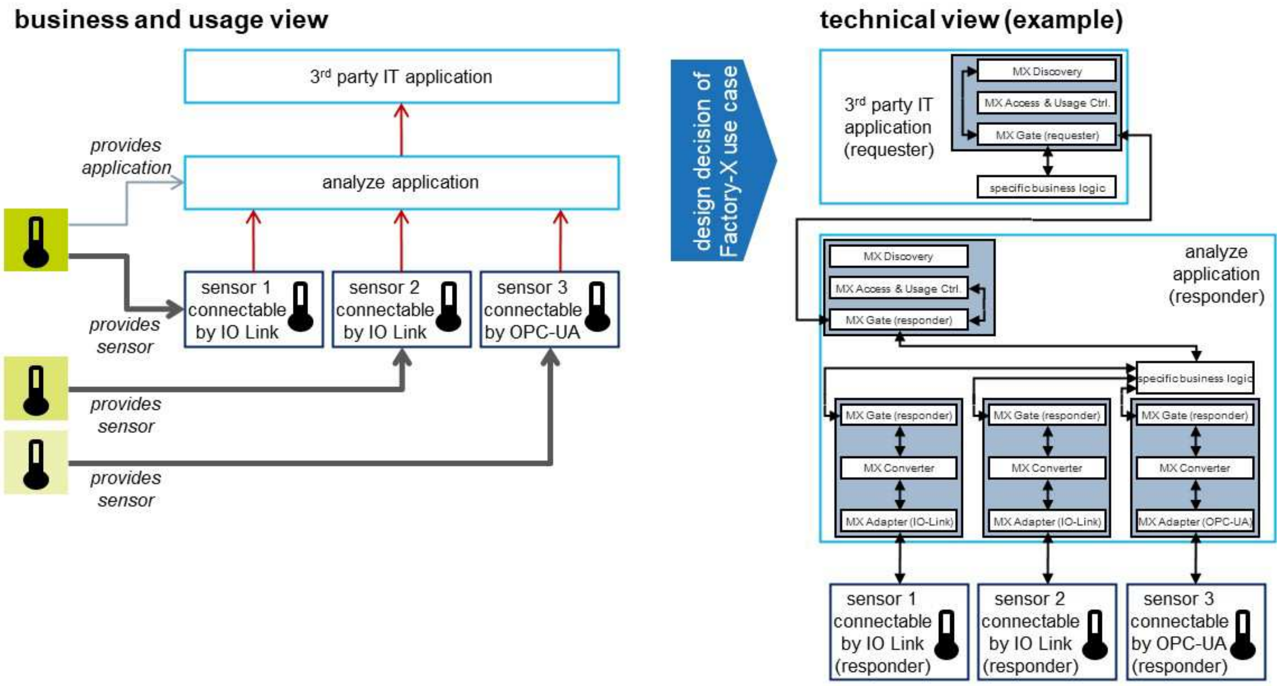

Figure 21: Modularity regarding MX-Port (example)

The constituents of an MX-Port can be integrated into a business application or be external from the perspective of a business (or legacy) application, see Figure 15. The entire MX-Port is configured by the operator of the business application according to its needs.

Figure 22 shows, in comparison to Figure 21, an example where the MX Adapter (IO-Link) and MX Adapter (OPC-UA) are provided externally (regarding the analyze application), for example by a 3rd party software provider.

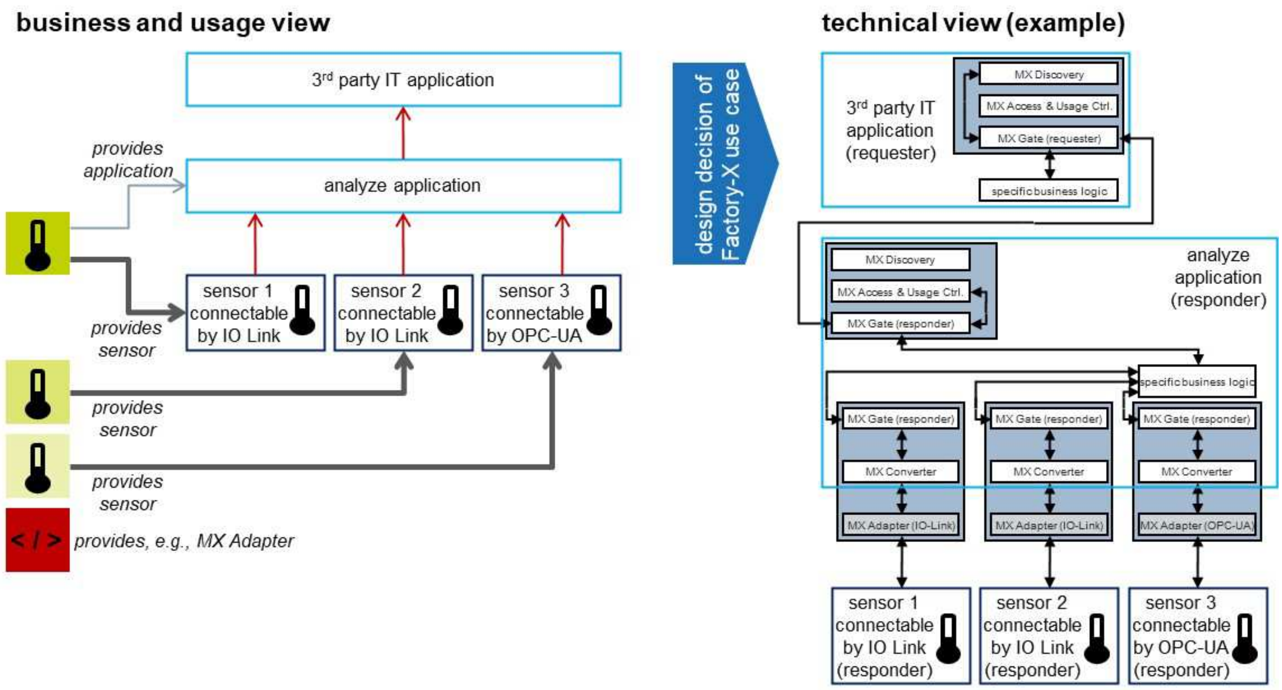

Figure 22: Modularity regarding MX-Port (example)

Figure 23 and Figure 24 show, in comparison to Figure 21, examples where the constituents of an MX-Port are provided externally (regarding the analyze application) from different software provider.

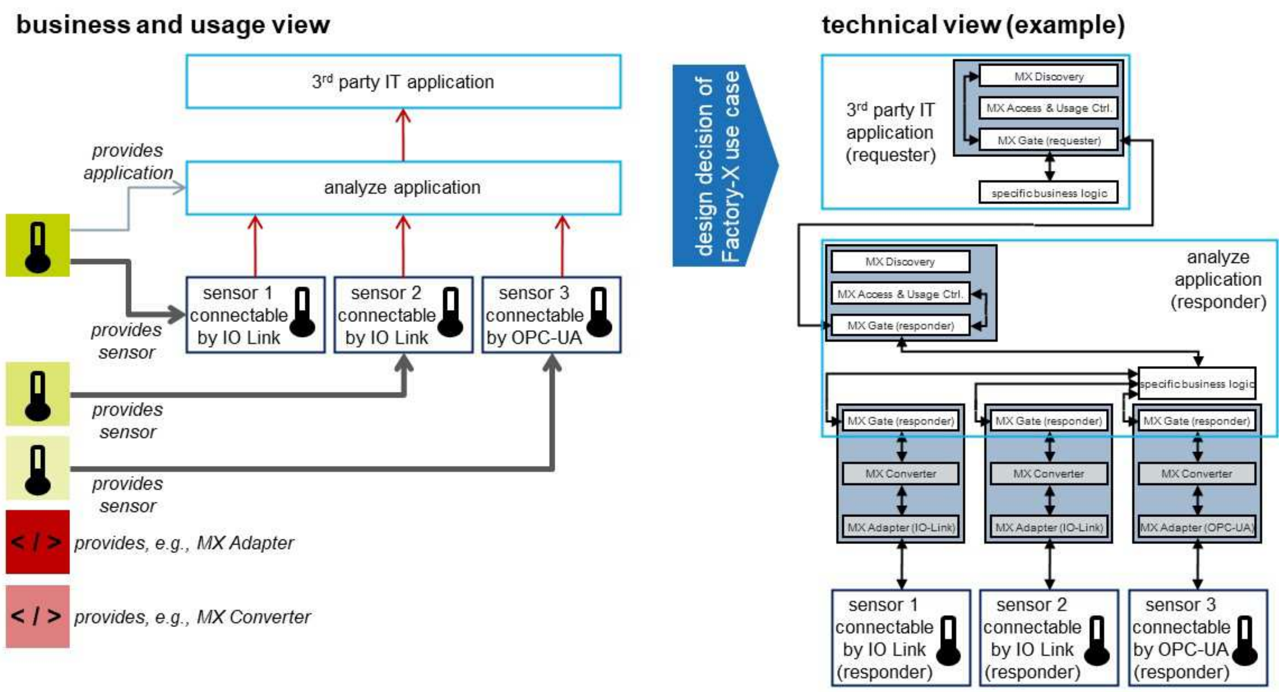

Figure 23: Modularity regarding MX-Port (example)

Since in Figure 24 the MX Gates of the legacy applications “sensor 1”, “sensor 2” and “sensor 3” in the role responder are provided externally, the analyze application (logically) provides internally an MX Gate in the role requester. In Figure 24 four logical MX-Ports are shown for the data provision by the sensors. However, since these MX-Ports are all operated by the same legal entity, just one MX-Port can be used instead. Another possibility is to assign the MX Gate in the role requester of the analyze application to the other MX-Port of the analyze application, see Figure 25.

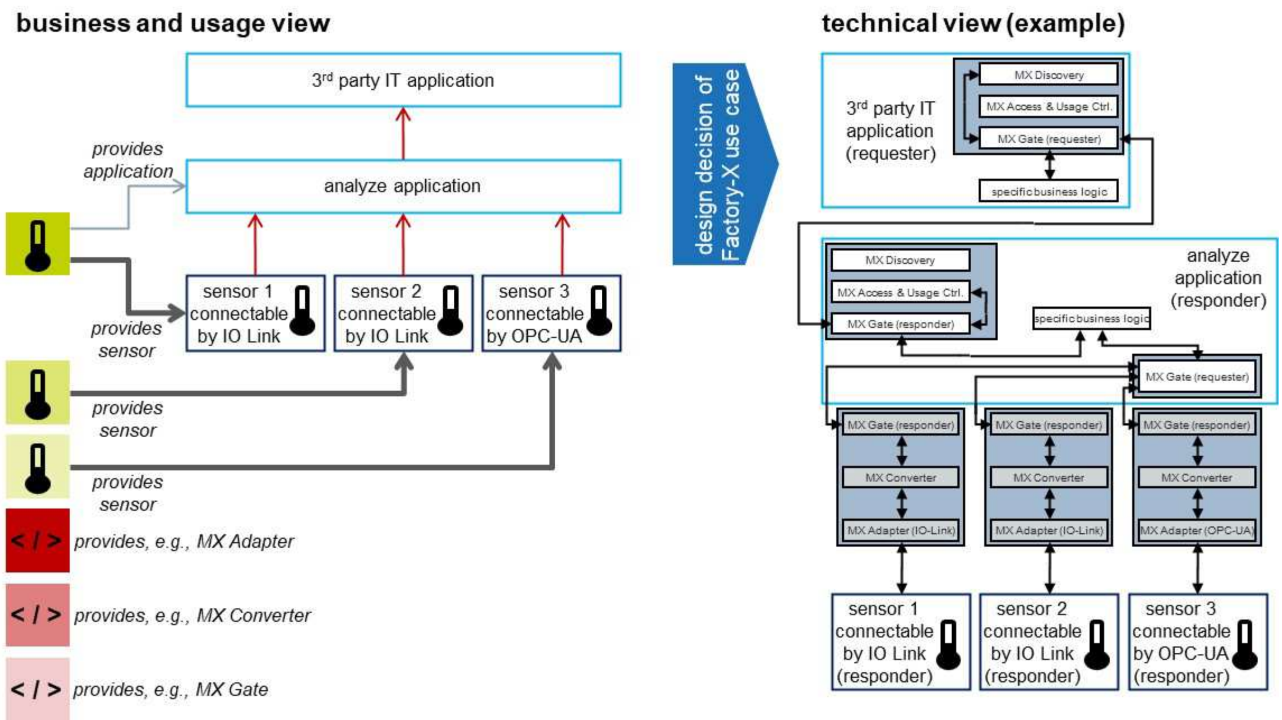

Figure 24: Modularity regarding MX-Port (example)

7.6 Cross-company data flow

Figure 25 extends the examples shown in Figure 20 - Figure 24 by illustrating data flow between different legal entities. Figure 25 shows three different legal entities sharing data:

- One legal entity operates the 3rd party IT application
- A second legal entity operates the analyze application and the two legacy applications sensor 1 and sensor 2.
- A third legal entity operates the legacy application sensor 3.

The dashed lines in Figure 25 indicate the scope of the three legal entities.

Each of these three legal entities requires an MX-Port including MX Discovery (so that the MX-ports of the other legal entities can be found) and MX Access & Usage Control (so that access and usage can be defined for the applications operated by the legal entity). For data to flow from one legal entity "A" to another legal entity "B", legal entity "A" must offer an MX Gate in a provider role in its MX-Port and the other legal entity "B" must include an MX Gate a requester role in its MX-Port.

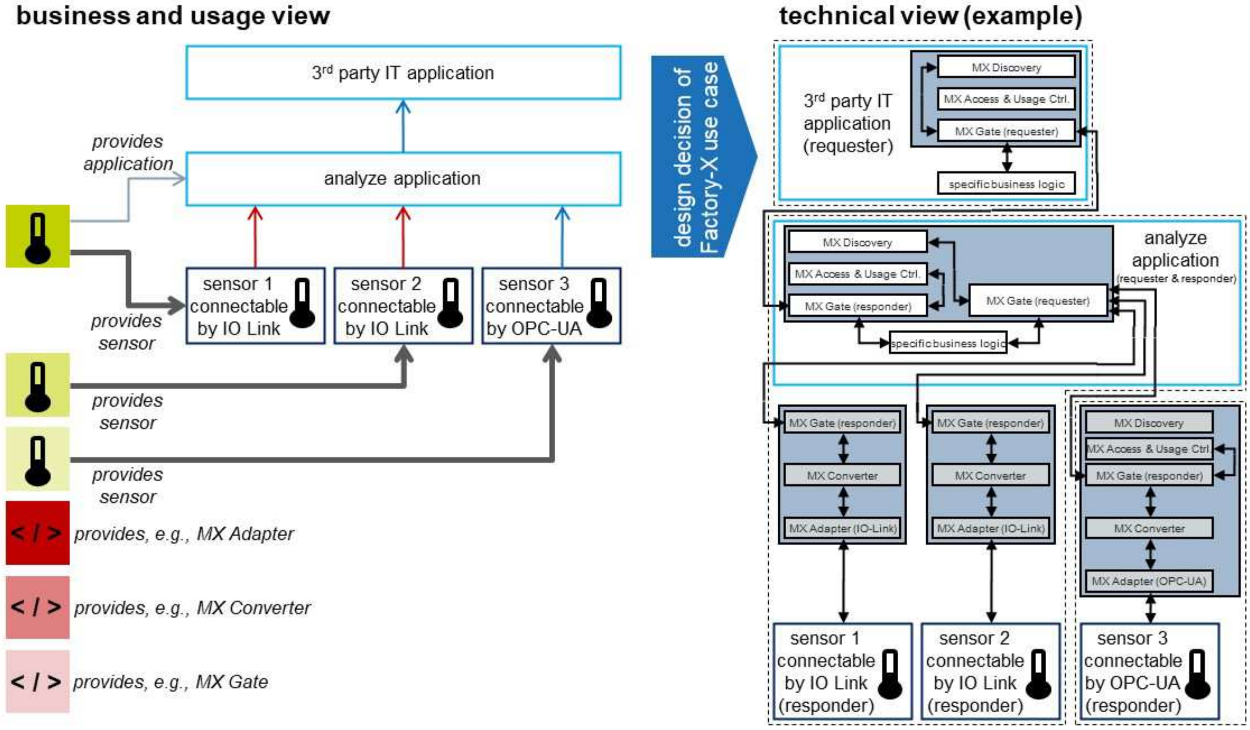

Figure 25: Cross-company data flow (example)

8 Annex A: ISO/IEC/IEEE 42010

The described approach is based on ISO/IEC/IEEE 42010, which was also applied by IIC and Plattform Industrie 4.0. Figure 26 illustrates the systematic overall approach. More details can be found in [3].

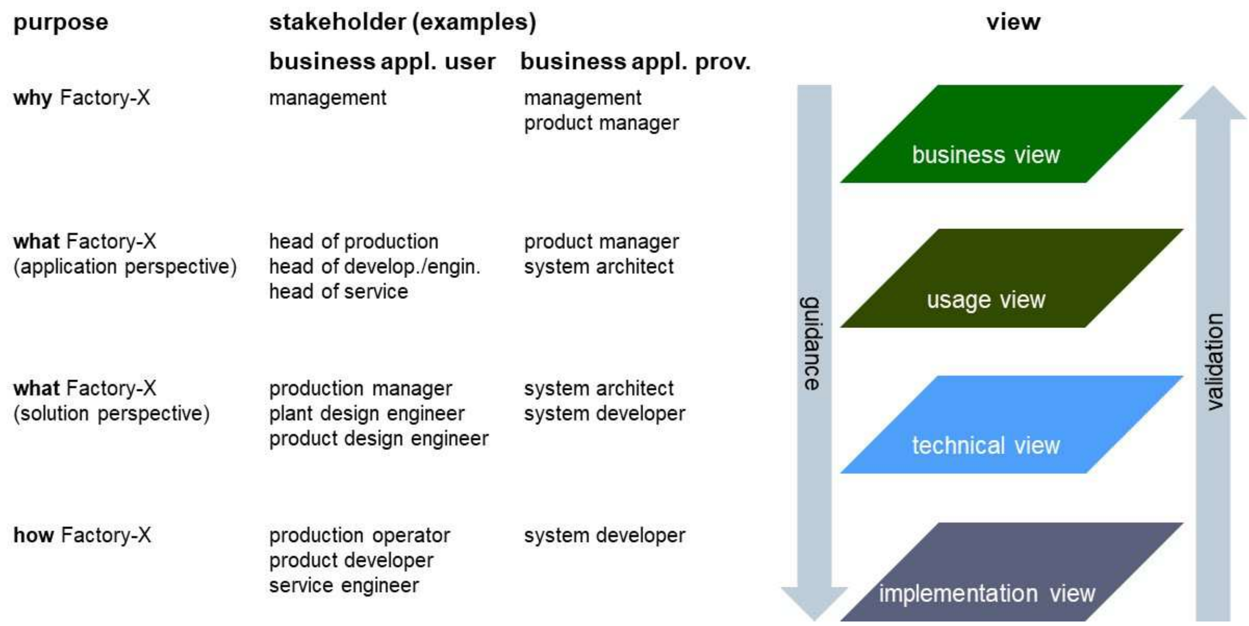

Figure 26: Systematic overall approach

9 Annex B: Business view of Factory-X

Figure 27 illustrates the business view of Factory-X from the perspective of the Factory-X use cases. Figure 27 shows the interaction of the business roles defined in "Business role" on a generic level, independent of a specific use case. Note that a legal entity can assume different roles in Figure 27 at the same time.

In the digital ecosystem Factory-X there is one governance body, which

- Specifies the functionality and compliance of shared services
- Specifies the compliance of business applications
- Certifies certifier

A certifier certifies the operation of shared services and business applications.

The digital ecosystem Factory-X is **open** in the sense that – apart from the governance body – all other business roles can be assumed by any legal entity.

Typically, a business application must be integrated into the IT/OT landscape of the user of the business application, whereby services from a system integrator can be used.

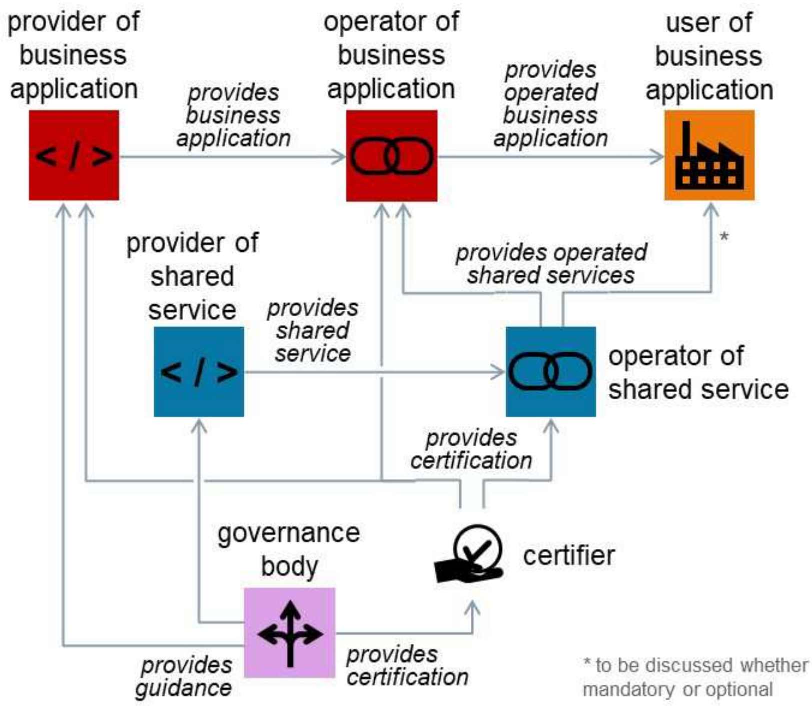

Figure 27: Business view of Factory-X

The individual business roles are based on the following business models:

- Provider of business application, who provides/licenses a business application commercially: The scope and standardization of business application is driven by companies/associations/etc. having an interest in the business application. The business application must be designed to be based on the mandatory shared services (Factory-X mandatory services not yet defined). To be Factory-X compliant a certification as a “provider of business application” is necessary (Factory-X certification not yet defined).
- Operator of business application, who operates a business application commercially: The operation of a business application is based on shared services operated by operators of the shared services. To be Factory-X compliant certification as an “operator of business application” is necessary (Factory-X certification not yet defined). Some of the shared services required by the business application can also be operated by the operator of business application. The operator of business application and the user of business application must agree on the operation of shared services required by the business application.
- Provider of shared service, who provides/licenses a shared service commercially: The Scope and standardization of shared services is driven by TP4. To be Factory-X compliant a certification as a “provider of shared service” is necessary (Factory-X certification not yet defined).
- Operator of shared service, who operates a shared service commercially: To be Factory-X compliant a certification as an “operator of shared service” is necessary (Factory-X certification not yet defined).
- User of business application, who uses a business application commercially: The use of a business application buys a business application from a provider of business operation (or via an operator of business application) and chooses an operator of the business application. The operator of the business application and the user of the business application must agree on the operation of the shared services required by the business application. It is to be discussed whether a certification of a user of business application is mandatory.

The Factory-X project only defines the guiding principles regarding commercialization, the commercialization itself is not in the scope of the Factory-X project.

10 References

[1] BPMN: *https://de.wikipedia.org/wiki/Business_Process_Model_and_Notation*

[2] Tooling (Factory-X internal): Modeler - Factory-X Overarching Project Information - Confluence (atlassian.net)

[3] Industrial Internet Consortium:2019, *The Industrial Internet Reference Architecture Technical Report, https://www.iiconsortium.org/pdf/IIRA-v1.9.pdf*