<AutomationML/>

The Glue for Seamless Automation Engineering

Application Recommendation: Toolchain

- *Recommendations how to introduce AutomationML in industrial toolchains of manufacturing industries*

Document Identifier: AR TC, V 1.0.0

State: May 2023

Version 1.0.0, May 2023
Contact: www.automationml.org

Content

List of figures

1 Introduction

1.1 Goal/ Mission

The AutomationML association is working in different working groups to find the best solutions for modelling components, layout information, electrical hardware description, etc. The work group (WG) *Toolchain and Industrialization* combines these solutions into one common data model suitable for typical industrial use cases. It also evaluates existing AutomationML interfaces of common tools and platforms. This WG collaborates with other WGs, tool developers and the industry to mature AutomationML into a standard that can be used coherently in existing development tool chains.

The main target is to empower AutomationML to be used in industry!

This document is following the mentioned mission above. The main goal is to state concrete and clear rules for the use of AutomationML in a specific set of industrial equipment engineering disciplines. A clear focus on the necessary content of the data exchange between this disciplines and existing rules and documents of the AutomationML association are the basement of this document. This Application Recommendation is focused on the different processes of engineering e.g., documentation and classification of documents is not part of this Application Recommendation.

1.2 Definitions

1.2.1 Plant creation process

The plant creation process is the design process of production systems. It includes a "systematic, goal-oriented" approach in successive phases and carries out used methods and tools to plan a factory from the first idea to the construction and ramp-up of production. A process can be defined as a result of functions or operations that convert a given input into an output. This is described in chapter 1.2.2.

The reference process of the Automation Markup Language (AutomationML or AML) divides the factory planning process into five steps (see Figure 1).

It relates to the modelling scope later as in addition to the construction of the plant, it also considers its commissioning and the usage of the plant under normal conditions, including monitoring and maintenance.

The AutomationML reference process has a significantly larger modeling scope. In addition, the model focuses very much on the planning areas of interdisciplinary factory planning, which in addition can be extended in a very high range of detail and is particularly interesting for discipline-oriented modelling.

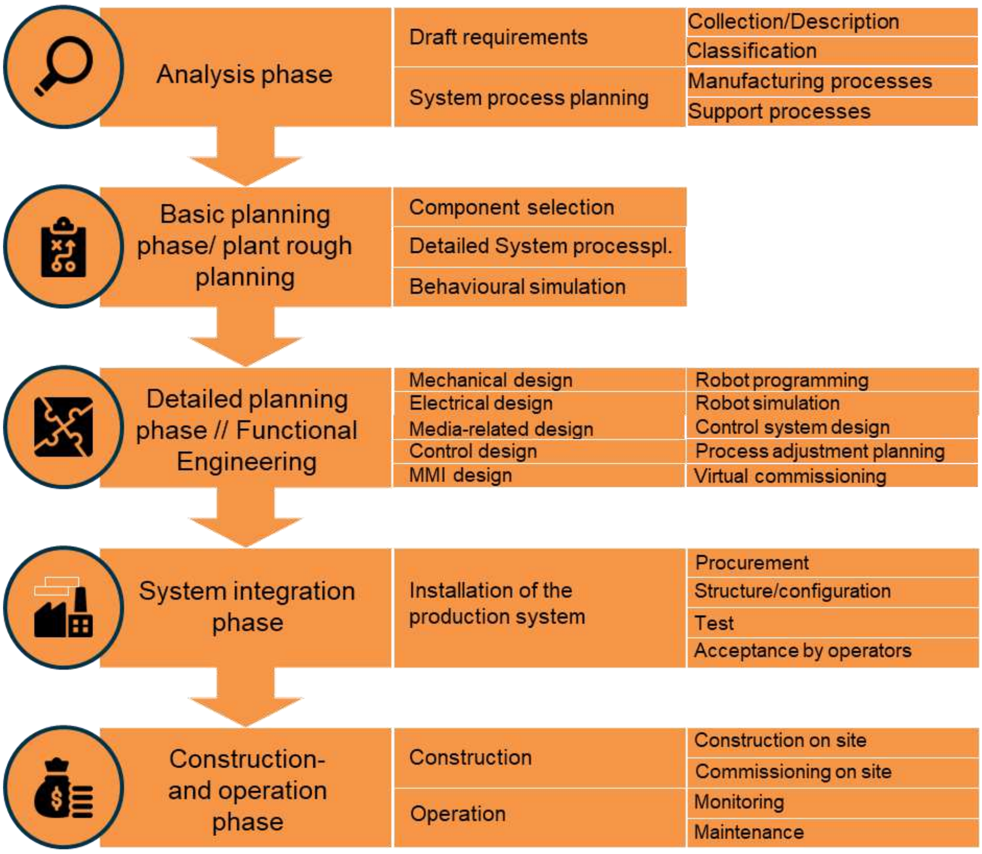


Figure 1 Planning phases after the AutomationML reference process (Lüder 2015, S.6-9)

1.2.2 Information Metamodel as Input/ Output Concept

As already mentioned, the above-described representation of information flows via concepts is a model. But if you want to see the connections of elements of a model, depending on the case, this can only be done by means of a metamodel. This is a model whose subject of investigation is a different model (here the description of the factory design process on planning concepts) and whose level of investigation is therefore called the meta-level. Its representation is called a metaschema.

Building on the above considerations, in a meta-model of information logistics, the understanding of this comes from the contexts of artifacts, engineering activities and planning concepts (see Figure 2).

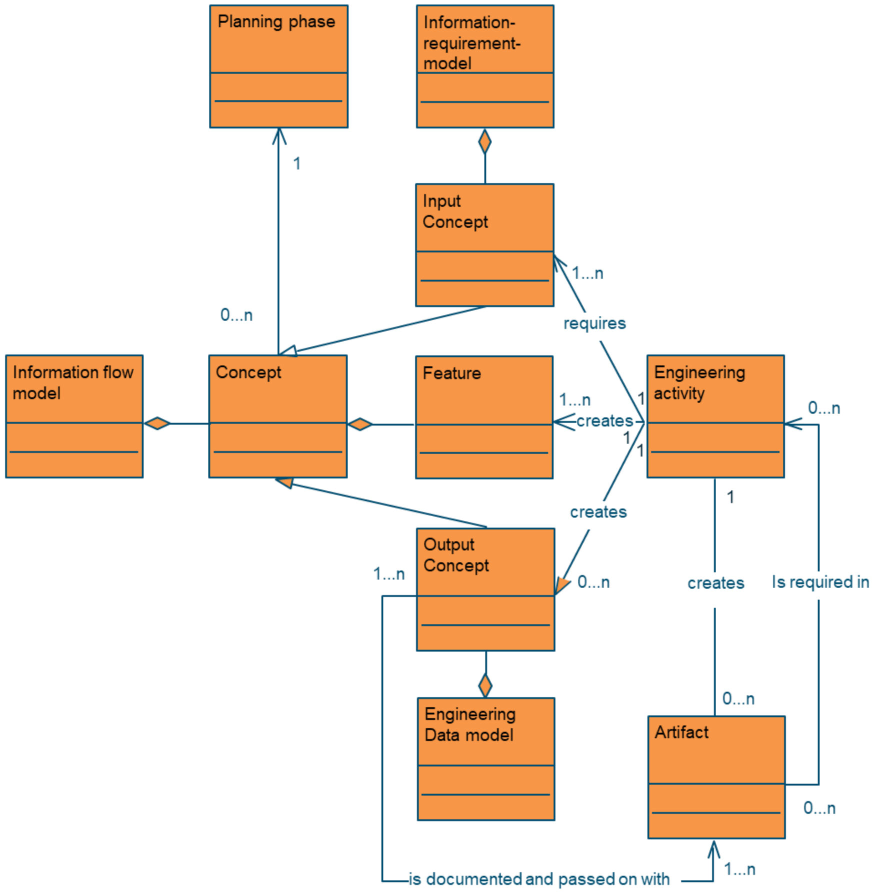


Figure 2 Metaschema of information logistics (Baumann 2020, S.20-22)

1.3 Scope

Resource, no product, no process!

“The three-view-concept is an example for interlinked engineering data. To understand this concept, different point of views must be applied to complex plant engineering data: the focus on resources, processes, and products.

- A resource is an entity involved in production; they execute processes and handle products. Examples for resources are robots, conveyors or machines.
- A product depicts a produced good. Products are processed by resources – this includes material handling, creation of intermediate products etc. This is valid for manufacturing as well as process engineering. Products can be built up hierarchically and described, e.g. by a (part) assembly tree.
- A process represents a production process including sub-processes, process parameters and the process chain. Examples are a welding process, a transport process or a filling process. In technical terms, processes modify products.” (Schleipen and Drath, 2009)

For different stakeholders, the PPR type of assets may be different. During the lifecycle of assets, the asset can also change its PPR type. During the production of the asset (e.g. an IIoT device or a laser scanner produced by the device vendor) it is of PPR product. After sale during the usage phase of the asset, it is a resource (e.g. when the laser scanner or the IIoT device is part of the production infrastructure for the plant operator).

1.4 References working groups and documents

WG Component, *Whitepaper: AutomationML Component, October 2020*

WG AR APC, *Application Recommendations: Automation Project Configuration, November 2021*

WG AML2AAS, *Application Recommendations: AAS Representation, November 2021*

2 Common Data Model

2.1 Definition Common Data Model

The plant development process consists of a variety of individual engineering disciplines. Each of these disciplines is already working with digital models. A system layout is created in the appropriate tools, as is a mechanical design, an electrical circuit diagram or various simulation models. The individual steps build on each other and each discipline consumes information from the previous process. For example, virtual commissioning uses layout data, behavioral information, mechanical drawings including kinematics and hardware configuration data from electrical design. After successful engineering, some information is required for the further maintenance process. Here too, specific tools and data models are used. A centralized Common Data Model is necessary for connecting and referencing all engineering processes. The Common Data Model only includes the specific data, that needs to be shared with other engineering processes.

The digital "Common Data Model" is intended, among other things, to accelerate shorter project runtimes in the engineering of production plants for the faster integration of new products and technologies.

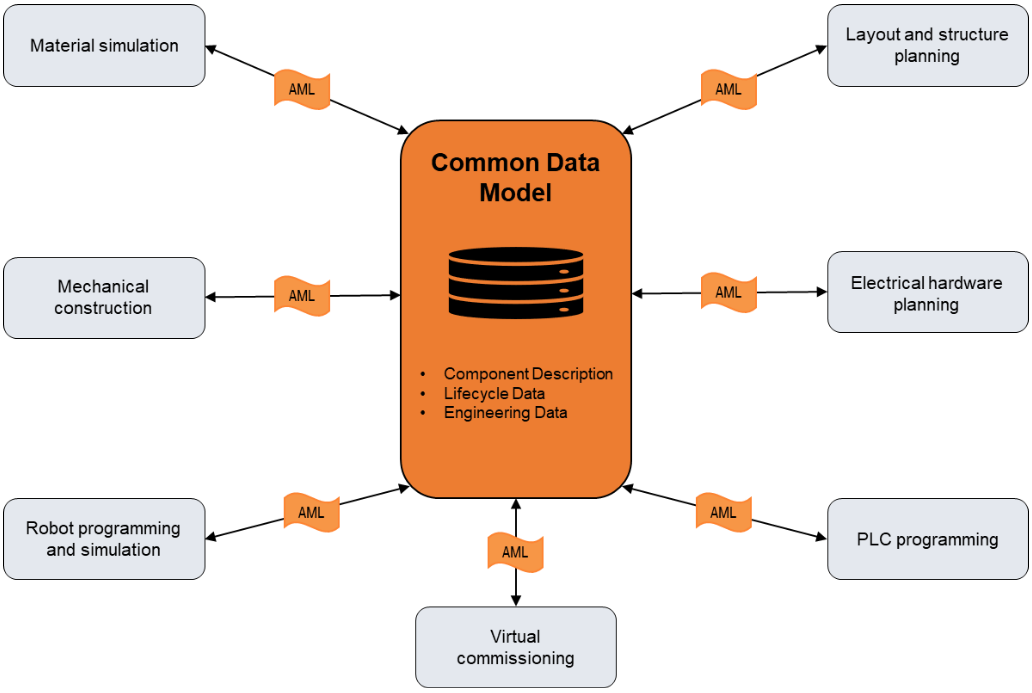


Figure 3 Definition "Common Data Model" (Draht2 2021, S.379-385)

2.2 Target picture of data exchange within the toolchain

If one forms the union of the intersections of all data exchanged between two tools, the result is an overall model that can be used for the data exchange of all the tools involved.

This overall model is the core of a data management platform (see Figure 4). In this case, it is necessary for each tool to decide how its internal data model and the overall data model relate.

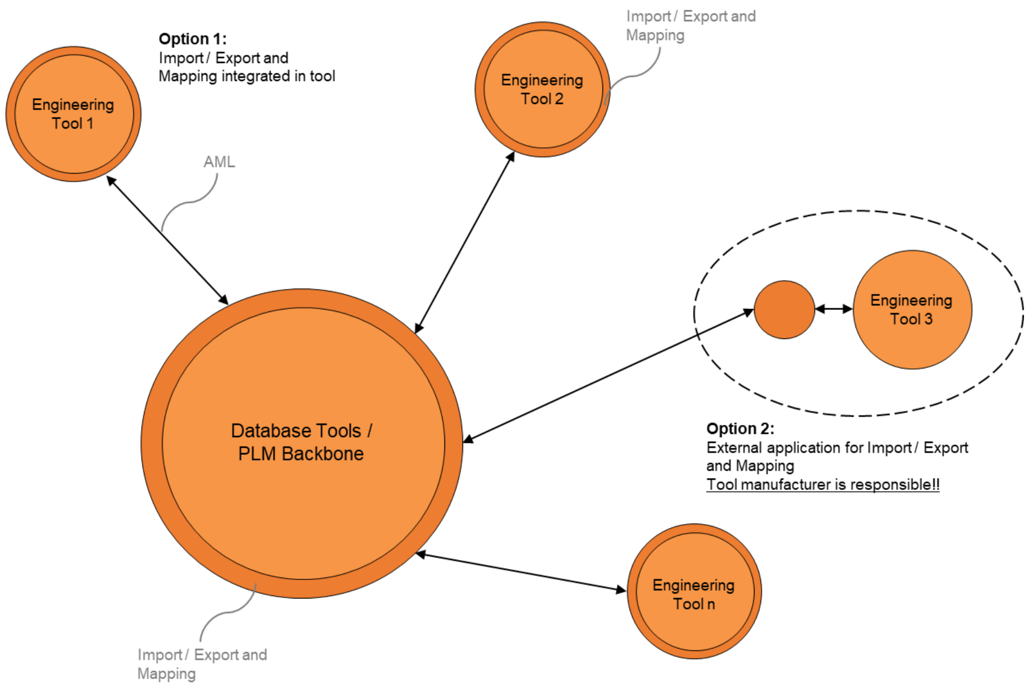


Figure 4 Possibilities for Toolchain

2.3 Tool categories for this Application Recommendation

The different tools are directly matching to different user groups or disciplines in the engineering process. This document will focus on the following categories.

2.3.1 Factory and Layout planning

The factory and layout planning is the actual planning process for plant engineering. It enables the creation of a quantity structure and layout, to provide a holistic resource structure of the production line for all planning processes in the early phase.
Layout and process planning accompanies large parts of the planning process. Layout planning includes the use of the surfaces, i.e. the positioning of the individual devices such as robots, tools and protective equipment. Process planning maps the sequence of the individual production tasks and accompanying activities combined with the time required on the individual devices. From the initial estimation of the areas and time requirements as well as a rough process, the plans are gradually refined with the progress of the engineering process and stored with concrete planned data. As part of the other planning steps, the timing, spatial positioning and associated use of resources are compared with the planning specifications. The details result in changes in order to achieve the goals set. As a rule, in layout and process planning, the changes from the detailed planning swell together and are distributed back to the planning steps from there. The process of bringing in, consulting and deciding on the changes and then distributing them is usually done manually as well as the maintenance of the changes in the detailed planning. Especially up to the securing of time requirements and layout in the robot simulation, this data is still subject to changes, since from the process step of the robot simulation statements to the point of accessibility and positioning of the tools as well as the required times.

The production planers prepare a technical enquiry, which will be transmit to the linebuilders. Afterwards the different linebuilders give the offers back to the purchaser.
For the decision which linebuilders get the order a technical evaluation of the hole layout and resources is necessary. For this the linebuilders have to give a functional layout back in the workflow.

Collaboration between:

- production planning department
- cost planning department
- facility planning department
- building media planning department
- linebuilders

Input:

- specification of the products to manufacture on the lines
- area and buildings, in which the productionlines must fit in
- cyclediagramm
- BIM model
- architecture data

Output:

- rough layout of the production lines
- quantity structure
- forecast and cost of complete productionline and engineering
- structure of the whole equipment with all elements inside
- positions
- coordinates
- zero-points

Next process step:

- offers to the linebuilders
- offers from the linebuilders
- mechanical construction (detail engineering) e.g. of robotgripper and fixture

2.3.2 Mechanical construction

The mechanical construction creates assembly instructions, single-part descriptions and drawings, structural calculations and parts of the plant documentation, such as maintenance instructions and intervals, piece lists, spare parts and consumable lists. For this purpose, the construction of several CAD systems, but mainly the software CATIA, and online planning tools for pneumatics, as well as the Office package for bills of materials and documentation. Accordingly, the main file formats of the design are STEP, CatPart/CatPro-duct, formats of component manufacturer tools, PDF, office formats and image formats such as JPG. In the event of process deviations, recursions or loops are carried out with additional personnel, additional project meetings oriented towards the specifications are carried out and protocols are kept in place for communication assurance. The construction department essentially finishes the construction, so the design department itself is not involved in the acceptance process. But results of the acceptance are returned to the design in the event of a concrete need for change.

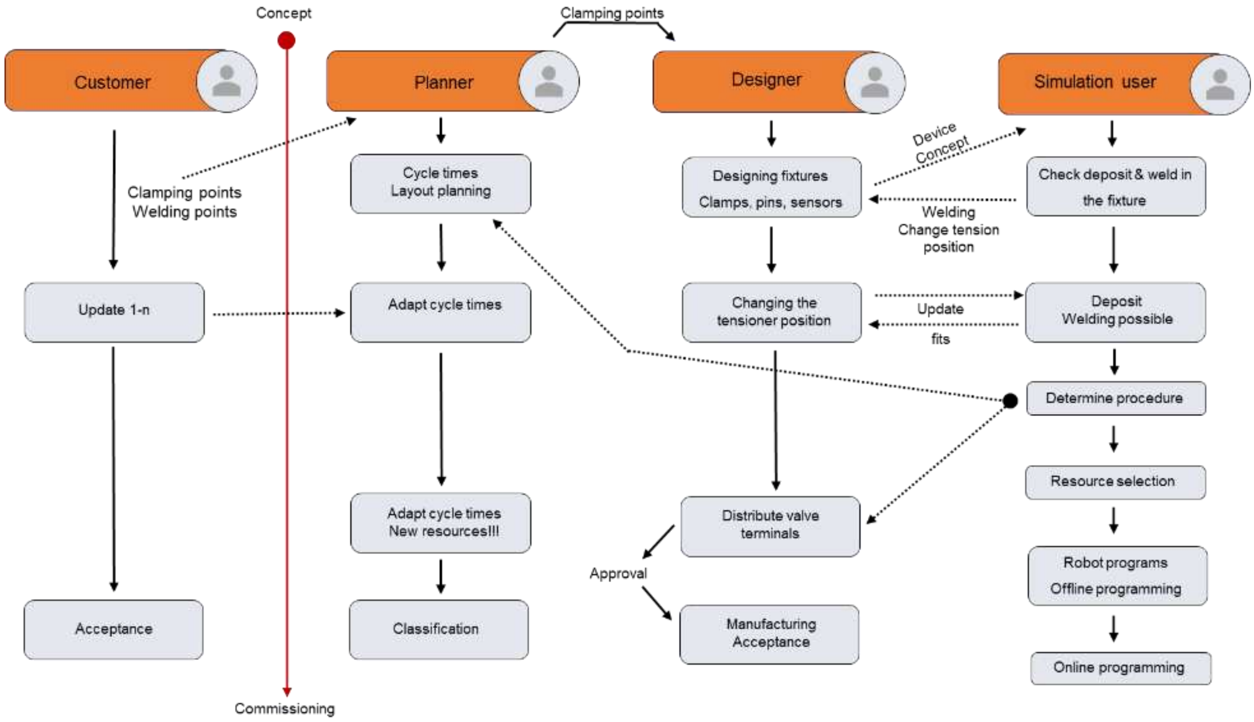


Figure 5 Mechanical Construction

Collaboration between:

- process planning
- factory planning
- product designer
- joining designer

Input:

- product data
- process data
- factory layout
- standards and specifications
- sequence diagram
- safety concepts

Output:

- 2D and 3D-construction drawings
- mechanical bill of material
- safety layout
- process description
- optional: pneumatical and hydraulic construction drawings

Next process steps:

- geometrical simulation
- pneumatical and hydraulical simulation
- electrical construction

2.3.3 Electrical construction

Electrical engineering deals with all technical equipment that is operated with electrical energy. This concerns in particular drive, control, measurement and regulation technology, automation and energy distribution.

The electrical engineer takes care of:

- conception and planning
- development based on available components
- construction of the control cabinet layout
- wiring
- mounting
- technical service

for electrical systems, installations, machines, devices and processes.

Input (supplier):

- design layout
- material release list
- customer specifications
- operating concept

Output:

- schematic diagram
- cabinet layout
- documents for the production
- bill of material's
- labeling /marking list
- special manufacturing instructions

Next process step:

- ordering components
- programming of the PLC
- simulation

2.3.4 Media related design

Media related engineering deals with all technical equipment that is operated with other than electrical energy e.g. hydraulic, pneumatic or fluid.
This concerns in particular drive, control, measurement and regulation technology, automation and energy distribution.

The media related engineer takes care of:

- conception and planning
- development on the basis of available components
- construction machine layout
- wiring and piping
- mounting
- technical service

for media related systems, installations, machines, devices and processes.

Input (supplier):

- design layout
- material release list
- customer specifications
- operating concept

Output:
- schematic diagram
- machine layout
- documents for the production
- bill of material's
- labeling /marking list
- special manufacturing instructions

Next process step:
- ordering components
- programming of the PLC
- simulation

Media related engineering can be done in parallel with electrical engineering.

2.3.5 PLC software development

A very frequently occurring task within the planning process of production and automation systems is the exchange of automation project configuration information of automation system devices between ECAD and PLC systems. To avoid multiple engineering in the participating systems ECAD and PLC systems need an interface for sharing this information.

In case of beginning engineering in the ECAD tool certain rules must be observed to get the hardware information in the correct location in the PLC tool. In case of beginning engineering in the PLC tool non placed functions must be placed and operated in the ECAD tool. The Application Recommendation “Automation Project Configuration” (AR APC) describes these workflows and the method of hardware configuration modelling using AutomationML.

The difficulty in the data exchange of automation project configurations is caused by the fact that ECAD tools and PLC tools have different views of automation system information. Whereas ECAD tools depict all electrical detail information of devices applied within automation systems in PLC tools only a logical compilation of the automation devices is used. So, in ECAD tools there are defined e.g. devices which are involved in an automation system, voltage connectors which are used for power supply of the devices, and wire types which are used to connect devices. But these are not used in PLC tools. On the other side in PLC tools there are device and control application specific conditions defined e.g. baud rates which are used within the communication connections, control code variables which are associated to control device inputs and outputs, and control application codes. But these are not needed in ECAD tools. Nevertheless, both types of tools have some information in common. For example, the wiring of a certain automation device to a PLC defines the address the device can be accessed within the PLC. This must be considered by development of import and export tools.

Usually, in a production system engineering process the construction phase in the PLC project will begin later than in the ECAD system because the completion of the ECAD documents is the base for the production of the control cabinet. The combination with the software within the plant and the following commissioning will not take place before all control cabinets are completed.

So the PLC engineer will usually attend later to the project than the ECAD engineer. Nevertheless, at an early point of time (during ECAD engineering) the automation project configuration of the plant must be defined because the ECAD documents must be generated and the parts must be ordered.

ECAD systems normally can handle the components of different PLC manufacturers which have certain analogies from a point of view of electrical hardware. But additionally, there are system specific / manufacturer specific parameters. Therefore, only the engineering system of the PLC manufacturer can guarantee a complete and comfortable handling of all parameters of a hardware component. So, the configuration of the PLC system should be done as far as possible within the engineering system of the PLC manufacturer.

Accordingly to the described criteria in most cases the following workflow is established.

1. Engineering the basic device configuration within the PLC project of the PLC programming tool and exporting it to ECAD tool.
2. Importing PLC configuration to ECAD tool, engineering of the ECAD project, and exporting the ECAD project to PLC programming tool.
3. Importing ECAD project into PLC programming tool and engineering of the PLC project.

If no ECAD project exists so far, the ECAD engineer first of all defines a raw project within the engineering system of the PLC manufacturer, the PLC programming tool. The ECAD engineer selects all needed components and defines the bus topology in close cooperation with the PLC engineer who has to implement the requested functions later on. This close cooperation ensures a high consistency regarding the selected hardware components. The automation project configuration will be exported from the engineering system of the PLC manufacturer and imported into the ECAD tool.

Based on the existing ECAD project the ECAD engineer executes the complete hardware construction, sometimes with slight adaptions. During this process the symbolic names for variables, tags or signals can be defined too. So, the PLC configuration is done under the following conditions:

1. PLC configuration can be imported from PLC programming system.
2. Configuration via graphical placement on overview page or navigator.
3. PLC-Device selection carried out from ECAD database.
4. Drag&Drop on pages from navigator.

Collaboration between:

- electrical hardware planning/electrical engineer
- PLC programming/PLC programmer
- simulation engineer

Input:

- electrical design ECAD (AML-export based on AR APC)
- electrical configuration
- list of sensors and actuators (I/O list)

Output:

- completed/modified automation project configuration for exchange to ECAD tools & virtual commissioning.

Next process steps:

- virtual commissioning
- commissioning
- operation & maintenance

2.3.6 Virtual commissioning

The production system development can be divided into the phases system design and system realization (Figure 6).

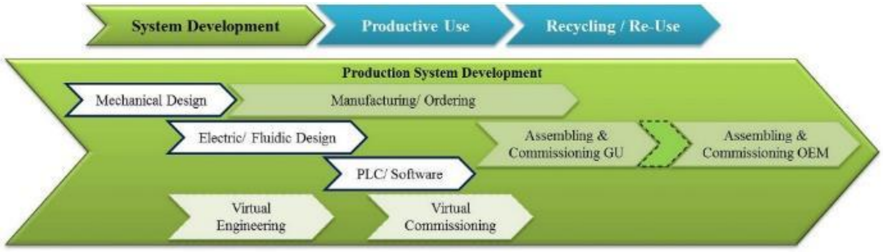


Figure 6 Production system development process

During the system design phase, mechanical, electrical, and software design are carried out. In principle, those phases are executed in parallel and depend on each other. Additionally, system validation activities are carried out in parallel to the three phases. These activities can be divided into the phases Virtual Engineering (VE) and Virtual Commissioning (VC).

The virtual commissioning can be observed as a separate phase of the development process of production systems. In this phase, all functionalities of the PLC's and robot's software are tested based on a simulation model of the production system that doesn't exist in real hardware at this time (Süß et. al 2015 & Damrath et. al 2016). For VC, first draft of the PLC and robot software as well as a working and tested simulation model of the production system must be available. This required simulation model, as well as VE's simulation model, contain several sub-models, e. g. 3D simulation, material flow and robot simulation (Strahilov et. al 2018 – section 9.3.4). Essential differences between VE's simulation and VC's simulation are

- on the one hand the additional behaviour model respectively behaviour simulation, that are needed to simulate logical behaviour of each component connected to the real PLC, and
- on the other hand the missing process simulation, which is replaced by PLC's software.

Furthermore, the 3D simulation model of the production system serves to visualize the movement of the system during the production process by the PLC by means of PLC software but not to check the mechanic of the system such as VE.

Based on practical experience, the 3D simulation model created and used by VE is taken as base to use it for the purpose of VC. In the market, several tools provide required functionalities to prepare the 3D simulation model and to conduct VC.

Additionally, the behaviour model of the production system is required by VC. This model represents the logical behaviour of each component connected to the PLC and communicates with the PLC via signals. For this purpose, the behaviour model of each component installed in the production system must be created.

Along with the behaviour simulation, robot program simulation is required as well. In this case, robot programs that are detailed during the PLC/software design are used and not OLPs (Strahilov et. al 2018 – section 9.3.1 & 9.3.3). To run the simulation of those robot programs, specific tools are required that must support specific robot programming languages. For this purpose, most robot manufacturers provide a tool to create and simulate robot programs.

To increase the benefit of VC, modelling effort must be kept to a minimum. For this reason, continuous usage of VE's simulation model must be done to prevent unnecessary repeated modelling time.

As soon as all functionalities of the PLC software are successfully tested, VC is completed. The result of this phase is the validated and optimized real PLC software that runs on the target hardware PLC. As following step, commissioning of the real production systems can be initialized (Strahilov et. al 2018 – section 9.2).

Based on practical experience, the main difficulty within the engineering workflow of production systems are the specific data formats of the used tools. Even worse is the use of standard data formats which do not support the exchange of data without losses, e. g. PDF. One exception is the use of the standard data format AutomationML to exchange simulation models between simulation tools and virtual commissioning tools. In this regard, an extended usage of a standard data format within the whole workflow is an important step which must be done during the design of the engineering workflow of production systems.

Collaboration between:

- customer / plant manufacturer
- plant operator
- plant planning engineer
- electrical engineer
- simulation engineer
- PLC programmer

Input:

- hall layout
- assembly sequence
- documentation (pneumatic diagram)
- electric diagram
- mechanical design
- 2D layout
- 3D simulation incl. layout
- robot offline programs (OLP)
- PLC programs (Backup)
- customer-specific libraries and standards

Output:

- documentation (incl. video, signal recording, …)
- behavior simulation
- 3D geometry simulation
- validated robot and PLC programs

Next process steps:

- commissioning
- equipment operation

2.3.7 Equipment operation and maintenance

The main focus of the common data model and the enrichment process is usually on equipment engineering. However, there is also great potential in plant operation. Two main cases have to be distinguished. Operational maintenance and applications related to big data and analytics.

Operational maintenance

Operational maintenance is directly connected to plant engineering and commissioning. The main tasks of operational maintenance are as follows:

- Installation of the plant hierarchy in the systems of maintenance based on the plant structure defined in the electrical design
- Location of spare parts in maintenance systems based on the components installed in the plant
- detailed description of spare parts (metadata, drawings, documentation, ...) based on component manufacturers' data sheets

Almost all of this information is generated in previous steps of plant engineering. It is therefore crucial that this content is transferred to operational maintenance systems. In addition, the operational department works on the production system. This means that information in the data model is changed. This modified model must be returned to the next planning process.

Collaboration between:

- plant manufacturer
- plant operator
- plant planning engineer
- electrical engineer
- mechanical engineer

Input:

- bill of materials
- factsheets
- plant structure
- maintenance guide

Output:

- materials
- function locations
- equipment
- maintenance structure
- maintenance plan
- working plan

Next process steps:

- operational maintenance
- procurement of spare parts
- spare parts strategy
- repairing

Applications Big Data and analytics

In the later operation of the production facilities, various information from different sources is used for analytics. In the case of predictive maintenance, corresponding measures are derived on the basis of these analytics.

In order to implement an analytics strategy and the technical solutions, the following information is essential:

- Knowledge of the asset hierarchy
- Knowledge of the installed components incl. all manufacturer's specifications
- Knowledge of the provided diagnostic values (e.g. via OPCUA) and their exact relationship to production equipment
- Knowledge of all connections (interfaces) and their parameters

In order for appropriate analyses to be established efficiently, this information must already be able to be derived from the engineering data and the corresponding data model.

Beside both maintenance use cases, also the production management will use the different parts of production facility to structure and organize the production strategy and production people. The main effect on a common data model will be the ability to restructure existing production assets or object in a different view. This view will not be a technical view like in the phase of electrical planning, it will be an organizational view on technical assets.

Collaboration between:

- plant manufacturer
- plant operator
- plant planning engineer
- electrical engineer
- PLC engineer

Input:

- bill of materials
- description of OPC UA interfaces
- IP-adresses of components
- plant structure

Output:

- data values in the analytics platform including reference to object

Next process steps:

- data analysis e.g. predictive maintenance
- data evaluation
- dashboard structure
- derivation of maintenance strategies
- energy management

2.4 Description of use cases

The detailed technical solutions for every use case are described in separate documents.

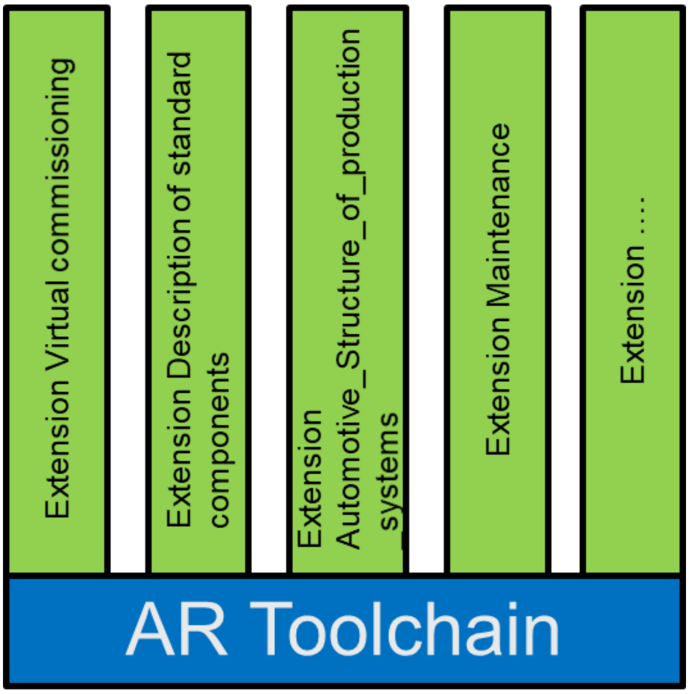
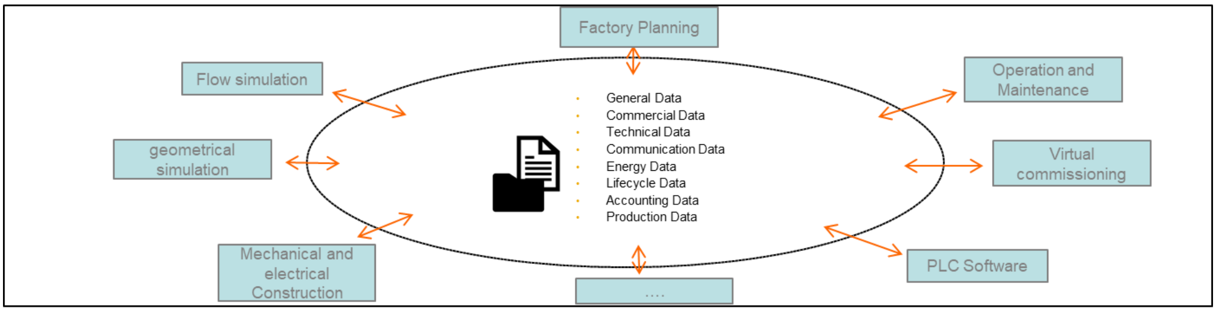


Figure 7 AR Toolchain structure

2.4.1 Description of standard components

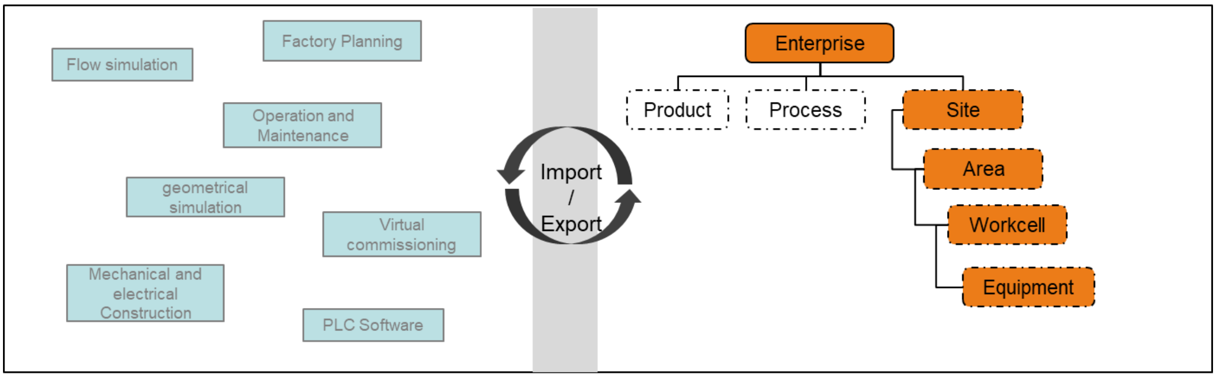


Throughout the plant creation process, complex component descriptions in various types are used in different ways. tools are needed. So that's not a waste of the way. Descriptions of 1 component will arise, will require a standard description of a component.
In this context, components are mainly active and passive series components that are used in automation systems (e.g. sensors, actuators, cables, infrastructure devices, robots).
Their type models should be able to be provided by the component manufacturers in the machine-readable replacement format AML. To do this, the target structure and standards are classified, the definition of the formats of the sub-models is provided, and the automatisms for quality assurance and filling of the target standards provided in the component models are defined.

Figure 8 Use case "Description of standard components"

Details are available in the "ARE_ Description_of_standard_components"

2.4.2 Base structure of automotive production systems

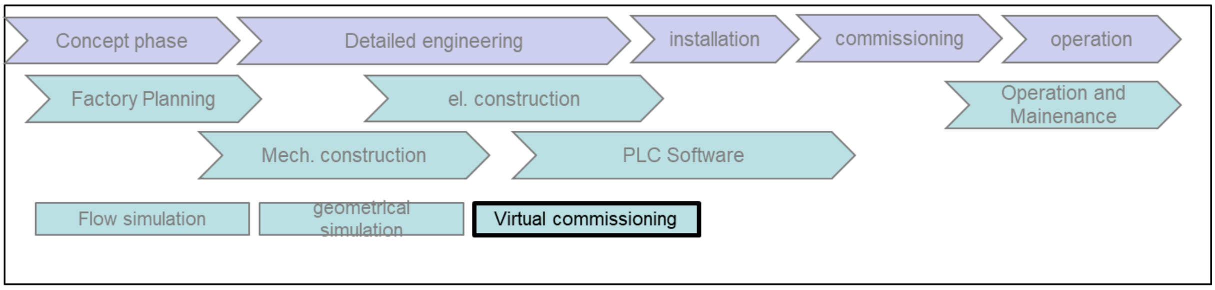


It consists of three views: product, process, and resource (following the so called PPR concept).
On the right handed side, resources are used to model the hierarchical structure of the production system. The structure objects are always the same and always existent. It follows the hierarchy definition given in IEC 62714-2 (see IEC 62714-2 ED1 [IEC 62714 2015]). Each resource of the production system is represented as an object in the resource tree.

Figure 9 Use case "Structure of automotive production systems"

Details are available in the" ARE_ Automotive_Structure_of_production_systems"

2.4.3 Virtual commissioning



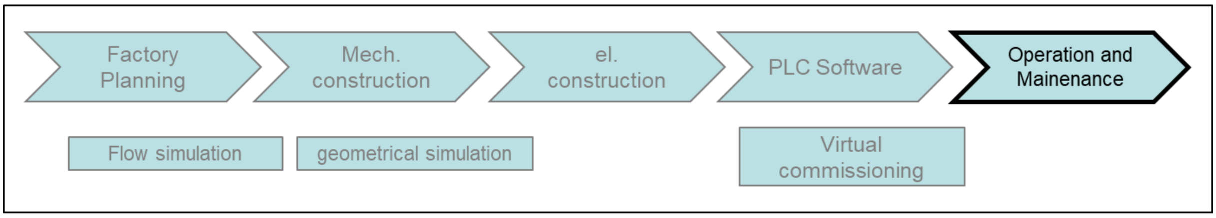
The development of production systems encompasses various phases which could be simplified as sequential process (Drescher at al 2013). In this system development phase of the production system, mechanics, electronics and software of the system are designed (Groover 2007).

The virtual commissioning can be observed as a separate phase of the development process of production. In this phase, all functionalities of the PLC's software are tested based on a simulation model of the production system that doesn't exist in real hardware at this time (Süß et. al 2015 & Damrath et. al 2016). For the purpose of VC, first draft of the PLC's software as well as working and tested simulation model of the production system must be available. The simulation model has to be created out of many information provided by prior development phases. The success and the efficiency of a virtual commissioning are basing the way how this information is provided and consumed by the involved tools

Figure 10 Use case "Virtual Commissioning"

Details are available in the "ARE_ Virtual_Commissiong"

2.4.4 Equipment operation and maintenance



The main focus of the common data model and the enrichment process is usually on equipment engineering. However, there is also great potential in plant operation. Two main cases have to be distinguished. Operational maintenance and applications related to big data and analytics.

Almost all of this information is generated in previous steps of plant engineering. It is therefore crucial that this content is transferred to operational maintenance systems. In addition, the operational department works on the production system. This means that information in the data model is changed. This modified model must be returned to the next planning process. In the later operation of the production facilities, various information from different sources is used for analytics. In the case of predictive maintenance, corresponding measures are derived on the basis of these analytics.

Figure 11 Use case "Equipment operation and maintenance"

Details are available in the "ARE_ Maintenance"

3 Different disciplines and content of model parts

3.1 Different naming concepts and structures

The different engineering disciplines and tool categories are working in optimized sub processes and structures. This leads to different naming concepts for the equipment. The different naming concepts also lead to different hierarchies and different amounts of hierarchy levels.

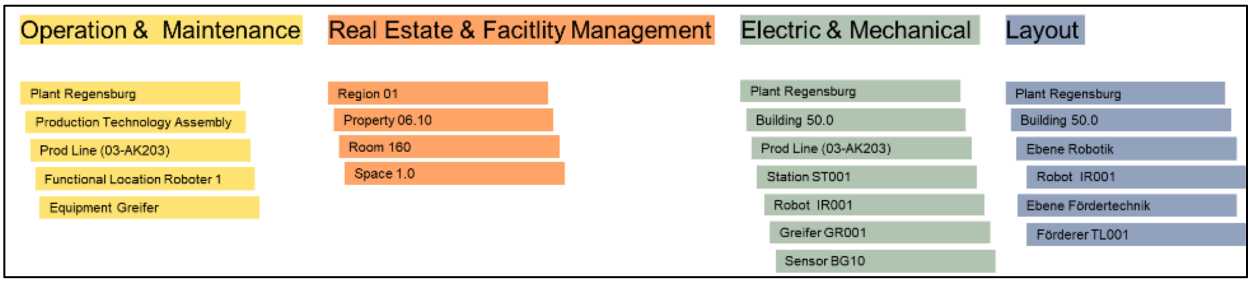


Figure 12 Example for different naming concepts

A common naming concept and equipment structure over all these disciplines would destroy much efficiency and shall not be the preferred way.

That leads to a concept with the following properties:

- every object (part, module, complete equipment, …) must contain the complete information of every discipline
- every object must have one unique identifier
- every object contains attributes with the different names
- the objects can be linked in an individual way, to model the different structures

The common data model must provide this different views and names.

3.2 Modelling hierarchies and views in AutomationML

A lot of possibilities are available in AutomationML which enable to model a structure:

- one structure directly modeled in the hierarchy
- use of mirror elements, to model a different view on an object
- internal links to show different kinds of relations
- …

To enable the different strategies with a minimum of effort, only two possibilities allowed. One common structure for simple or small use cases and one solution for different views.

3.2.1 One common structure

In case of a completely standardized naming and structure concept overall disciplines or in case of a tool or discipline specific AML export, it is possible to model the structure directly in the instance hierarchy with an object structure.

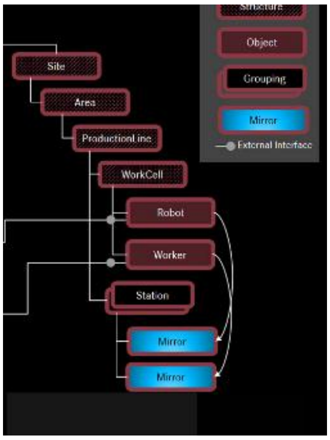


Figure 13 Mercedes-Benz AG - PPR Structure

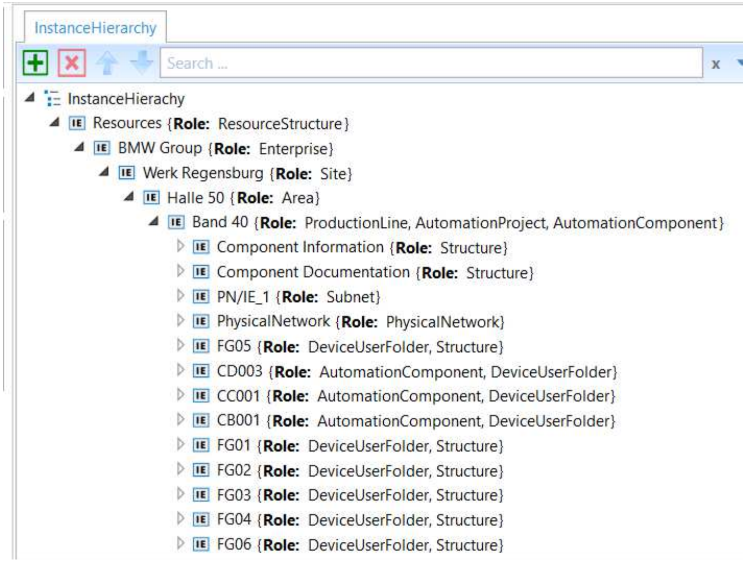


Figure 14 One Common Structure of BMW AG

3.2.2 Different views

In case of different views, it is relevant to ensure the following functionality to satisfy industrial needs:

- the different views (mentioned before)
- different amount of hierarchy levels
- parent child relations on every hierarchy level
- every discipline / tool category must be able to create the common objects
- new views can be added in a later step

Every object must be modeled in a flat list of elements. The hierarchy will not be displayed within an object structure. An object can be a single device or a complete equipment. The object contains elements with all the necessary data which describes the object itself (details in the following chapters) and elements that are describing the different structures and names.

Object (name or ID)

- **Object description (attributes or internal elements)**
- **Structure description (attributes or internal links)**

The structure description contains attributes and internal links:

- Attributes: discipline specific names of the object
- Internal links: discipline specific child links

Object (name or ID)

- **Object description**
- **Structure description**
    - **Attributes**
        - **LayoutName**
        - **mechName**
        - **contrName**
        - **…**
    - **Internal links**
        - **ChildLayout**
        - **ChildMech**
        - **ChildContr**
        - **…**

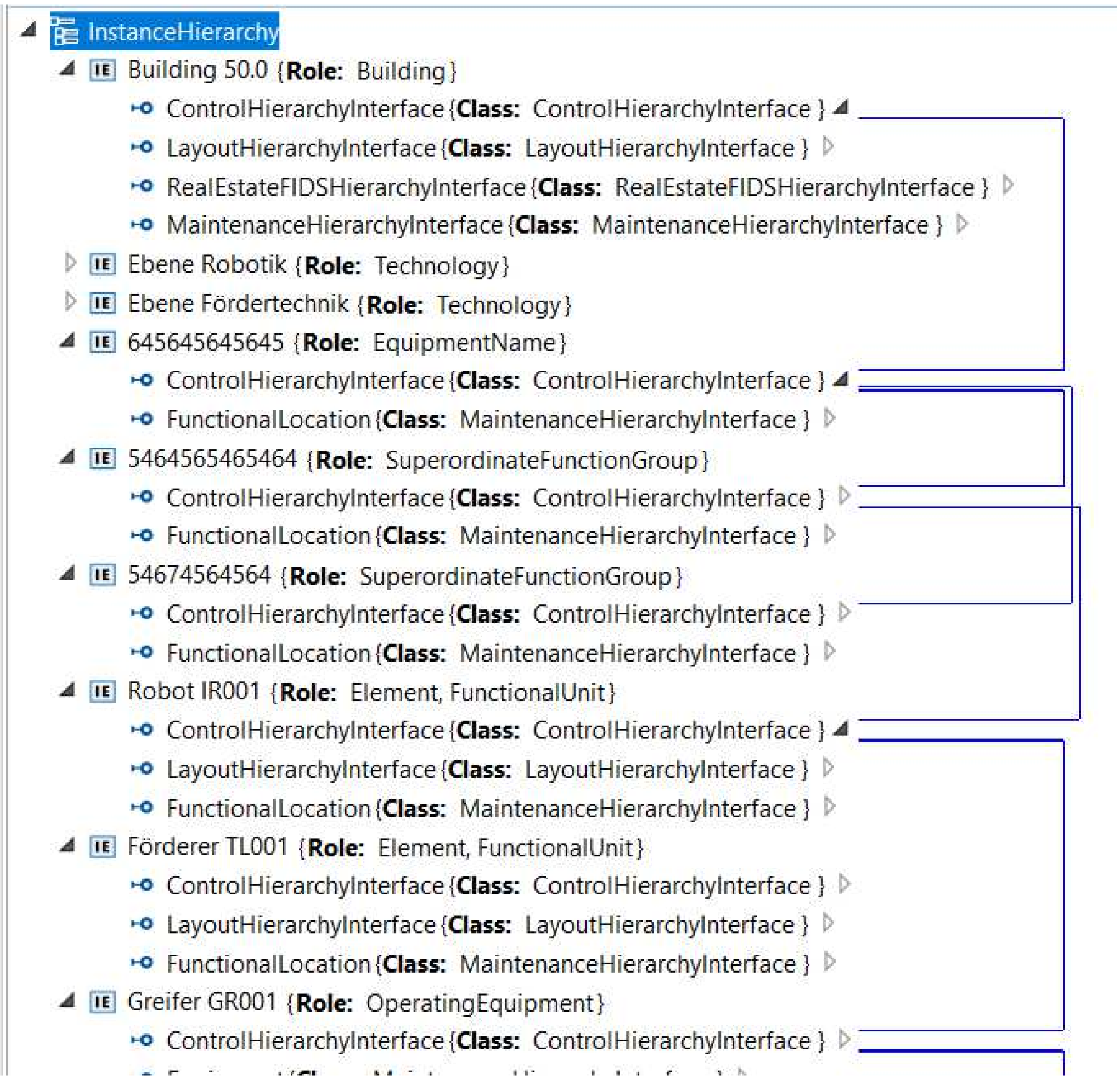


Figure 15 Child links

3.3 Modelling of Restrictions and IP-Clearing

The basic assumption is that a specification of the data supply chain exists, i.e. it is which data user is allowed to receive which data and which data he is allowed to send to other data users. The aim of the overall system is to protect assets and production data worthy of protection (in further IP) when cooperating with companies. The own and foreign administrators are trusted.

Basically, user rights can be decided not only on the basis of the users and files, but also on what action is performed with the files. A common distinction is the separation between read and write rights.The aim of the project is to modify AML files before sending/sharing in such a way that the recipient only receives the data for which it has permission. In Chapter 3.5, the prerequisite for this was defines that AML files are annotated in such a way that data worthy of protection is recognized as such. Identified protected data is then removed in the second step. The internal structure of the AML files and the interdependencies, however, execute the removal of the data. Figure 3 shows an example of why this is the case. Objects in AML can be used within the InstanceHierarchy, SystemUnitClassLib, RoleClassLib, and InterfaceClassLib appear and reference each other. In the example, the TL001 part is highlighted in the InstanceHierarchy.

It references the SystemUnitClass ConveyorXY from the SystemUnitClassLib and RoleClass AutomationComponent from the RoleClassLib. In addition, it is possible that there are references to InterfaceClassLib. All four categories may contain information worthy of protection, if any. For example, if a recipient does not want access to the specific instance TL001, must also be checked whether it has access to the corresponding SystemUnitClass ConveyorXY or the RoleClass AutomationComponent.

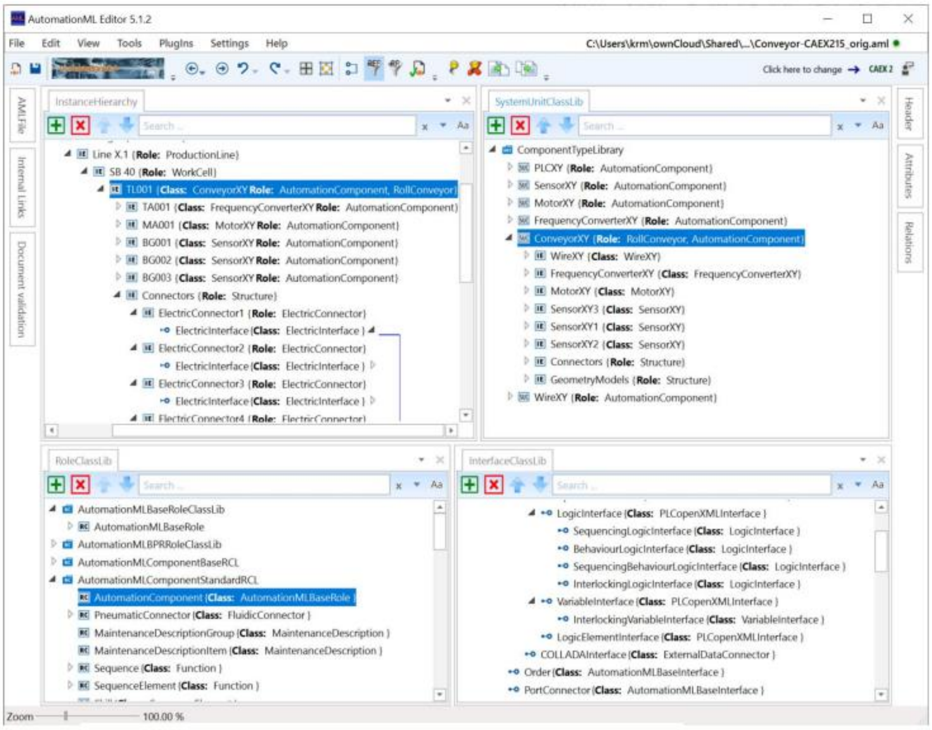


Figure 16 Example AML File

The document is then revised in three consecutive steps:

1. First revision based on users and their linked roles
    - users are assigned to projects. In principle, a user can only use data from projects where he is an employee.
    - users are assigned roles, for each role it is known what tools they use for the need to edit the AML document. Example: The user uses e-planning the E-Plan tool. In addition, the RoleClasses and InterfaceClasses must be known, who use the tools when importing an AutomationML document
2. Project-related revision based on a whitelist
    - this already defines in the call for tenders which parts InstanceHierarchy is used for the project.
3. Final post-control based on a Group-wide blacklist
    - this defines attributes and areas of InstanceHierachy that may not be exported to external partners

4 Outlook

4.1 Industry 4.0 Asset administration shell (AAS)

The asset administration shell makes assets and their related information accessible within one company or even for cross-company usage by means of internet technologies. It is a digital representation of an asset which can be used over the whole lifecycle of **the asset starting from an idea until the recycling phase. Interoperability is gained by means of unified information** and unified transport of this information. The information within an asset administration shell is organized in sub models which describe a specific aspect of the asset, e.g. identification, documentation or simulation specific information. Thus, the asset administration shell can be compared to a container which references different sub models. Most of the information concerning the asset is included in sub models and not directly within the container.

4.2 Interrelation of AAS and AML

For engineering information, AML may be used as sub model within the AAS. In this case, the present AML file is referenced within the AAS as “blob”.

Additionally, an AAS can be modelled in AutomationML. Therefore, mapping rules are necessary describing how to map the meta model of the AAS into AML. This is described in a common publication: Asset Administration Shell (AAS) Representation, Version 1.0.0, available via following link (Link 2019), State November 2019.

The associations AutomationML e.V., Industrial Digital Twin Association (IDTA), OPC Foundation and VDMA announce the publication of a jointly prepared discussion paper (German language only) “Interoperabilität mit AutomationML, der Verwaltungsschale, OPC UA inklusive Companion Specifications“. The paper is available for download from the respective organizations. Link (Link 2023), State April 2023.

4.3 AML Data governance

The goal of AutomationML governance is developing and ensuring a consistent, company-wide semantical model. A prerequisite for this is the consolidation and harmonization of the different discipline models existing in a company. An AutomationML governance provides the right set-up to achieve this milestone

As the very specific term “AutomationML Governance” does not exist in literature, yet this clause tries to define it by adapting the term “IT governance” as AutomationML is itself an aspect of IT.

According to Alan Calder [CAL05], “IT governance is a framework for the leadership, organizational structures and business processes, standards and compliance to these standards, which ensure that the organization’s IT supports and enables the achievement of its strategies and objectives.”

Adapted to AutomationML it could say: The AutomationML governance provides structures (the WHO) and processes (the HOW), which ensure that AutomationML supports and enables the achievement of IT strategies and objectives of a company (the WHAT). It is also responsible for developing standards as well as monitoring their implementation and compliance. The next points explains WHAT should be addressed by the board.

This is:

G1. Strategic alignment: Business and IT strategies and objectives shall be aligned.

G2. Value delivery: It shall be ensured that IT can deliver its promised benefits.

G3. Risk management: IT related risks shall be identified and addressed as information is one of the most valuable assets of a company.

G4. Resource management: Best possible investment in and adequate management of IT shall be ensured.

G5. Performance measures: Tracking and monitoring the realization of IT’s strategies and objectives is a must.

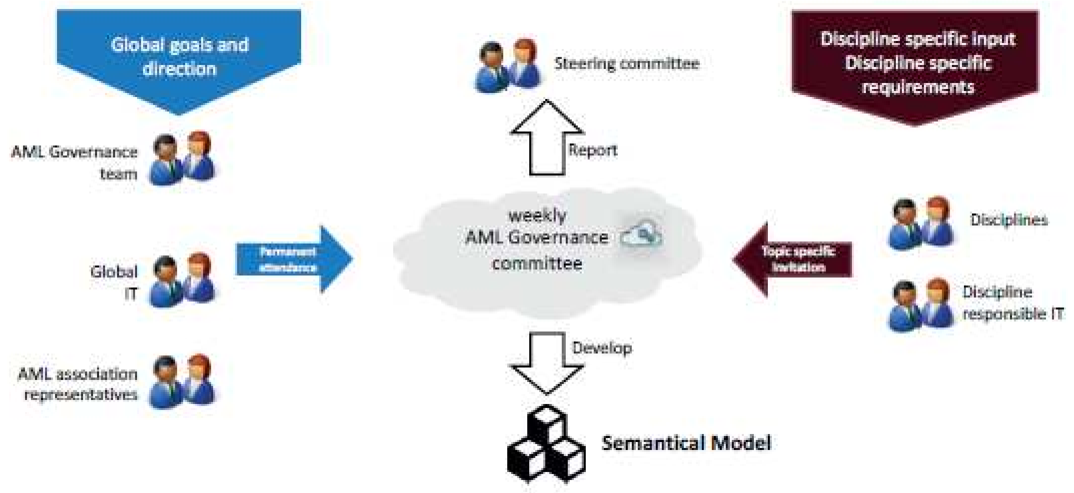


Figure 17 Roles, responsibilities and interactions in the AutomationML Data Governance (Draht2 2021)

More information to the AML Data Governance is described in the "The Industrial Cookbook" (Draht2 2021).

5 References

(Groover 2007) Groover P (2007) Automation, Production Systems, and Computer-Integrated Manufacturing. In: Prentice Hall Press Upper Saddle River NJ

(Dilts et. al 1991) Dilts D M, Boyd, Whorms H H (1991) The evolution of control architectures for automated manufacturing systems. In: Journal of Manufacturing Systems

(Li et. al 2001) Li j, Meerkov M (2001) Production System Engineering. Book: Springer 2001

(Koren et. al 1999) Koren Y, Heisel Y, Jovane F, Moriwaki T, Pritschow G, Ulsoy G, Van Brussel H (1999) Reconfigurable Manufacturing Systems. In: CIRP Annals - Manufacturing Technology

(Damrath et. al 2016) Damrath F, Strahilov A, Bär T, Vielhaber M (2015) Experimental Validation of a Physics-based Simulation Approach for Pneumatic Components for Production Systems in the Automotive Industry. In: 15th CIRP Conference on Modelling of Machining Operations

(Strahilov et. al 2012) Strahilov A, Mrkonjić M, Kiefer J (2012) Development of 3D CAD simulation models for virtual commissioning. In: Proceedings of TMCE 2012

(Süß et. al 2015) Süß S, Strahilov A, Diedrich C (2015) Behaviour Simulation for Virtual Commissioning using Co-Simulation. In: 20th IEEE International Conference on Emerging Technologies and Factory Automation

(Drescher at al 2013) Drescher B, Stich P, Kiefer J, Bär T, Strahilov A, Reinhart G (2013) Physikbasierte Simulation im Anlagenentstehungsprozess – Einsatzpotenziale bei der Entwicklung automatisierter Montageanlagen im Automobilbau. In: HNI-Verlagsschriftenreihe 2013

(Hämmerle et. al 2014) Hämmerle H, Drath R (2014) Erfahrungen bei der virtuellen Inbetriebnahme. In: Tagungsband zur Automation 2014

(Ovtcharova 2013) Ovtcharova J (2013) Virtual Engineering: Principles, Methods and Applications. In: International Design Conference – Designe 2010

(VDE 2013) VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik (2013) Cyber-Physical Systems: Chancen und Nutzen aus Sicht der Automation Thesen und Handlungsfelde. Report

(Strahilov et. al 2018) Multi-Disciplinary Engineering of Cyber-Physical Production Systems (2018), Chapter 9: Engineering Workflow and Software Tool Chains of Automated Production Systems In: Book, Springer Verlag

(Schleipen, Drath 2009) Miriam Schleipen, Rainer Drath: Three-View-Concept for modeling process or manufacturing plants with AutomationML. 13th IEEE International Conference on Emerging Technologies and Factory Automation. 22.-25.9.2009, Palma de Mallorca.

(Draht 2021) Rainer Drath: AutomationML, A Practical Guide

(Lüder 2015) Lüder, A.; Schmidt, N.: AutomationML in a Nutshell. In: Vogel-Heuser, B., Bauernhansl, T. und Hompel, M. ten (Hrsg.) In: Handbuch Industrie 4.0. Produktion, Automatisierung und Logistik, Bd. 9. Wiesbaden: Springer Fachmedien Wiesbaden, 2015

(Baumann 2020) Laura Baumann (2020): Masterarbeit: Entwicklung und Validierung einer Methodik zur artefaktinhaltszentrierten Analyse von Entwurfsprozessen von Produktionssystemen

(Draht2 2021) Rainer Drath: AutomationML, The Industrial Cookbook

(Link 2019) AR AAS Representation, November 2019, https://www.automationml.org/wp-content/uploads/2022/04/Asset-Administration-Shell-Representation-V1_0_0.zip

(Link 2023) Diskussionspapier – Interoperabilität mit der Verwaltungsschale, OPC UA und AutomationML, April 2023, https://www.automationml.org/wp-content/uploads/2023/04/Diskussionspapier-Zielbild-und-Handlungsempfehlungen-fuer-industrielle-Interoperabilitaet-5.3.pdf