

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Automated production monitoring and control system engineering by combining a standardized data format (CAEX) with standardized communication (OPC UA)

Miriam Schleipen
Fraunhofer IITB Germany

1. Introduction

Production monitoring and control systems are deployed to monitor and control the complex processes executed during the assembly of a car. In this context, a production monitoring and control system stands for a complex central or decentral IT system for collecting, aggregating and processing process signals and values in real time. It has a controlling effect on manufacturing and assembling processes, either in an automated way or by means of user interventions. In line with the definition by (Polke, 1994), a control system is meant to support shop-floor staff in managing their equipment and in controlling and monitoring the production processes. ProVis.Agent® (Sauer & Sutschet, 2006) is one example for such a system. In the following publication, engineering is viewed as a process to which multiple parties contribute using a wide variety of tools (see (Drath, 2008)). Today's world is marked by continuous change and enhancements. To remain competitive, plant operators have to respond quickly to the situation and requirements of the market. Technology has to support this. Any change in the production process has to be considered and the equipment has to be re-configured and re-customized. Today, efficient modifications to manufacturing systems are really difficult and result in an increased demand on adaptivity. For an efficient information exchange, all systems involved have to interact as seamlessly as possible in the heterogeneous environment. This is called interoperability. The cost pressure in modifications to or re-engineering of existing plants and the development of new plants is increasing constantly. This process is both cost-intensive and error-prone and includes a multitude of technical as well as human interfaces. To enhance the mutability of plants, it is necessary to find generic, reusable solutions to meet these very challenges. (Fay et al., 2008) therefore describes an approach to systematically assess engineering organisations and, on this basis, to identify means which are particularly suited to improve engineering within this organisation.

Production monitoring and control systems monitor and control manufacturing plants and systems. Before they can be taken into operation, they have to be configured. To this end, the topology and the structure of the relevant production plant as well as information about the incoming and outgoing interfaces of the components involved in the production process

have to be customized. In addition, process control images are required to visualize the current production activities. At present, the customization of marketable production monitoring and control systems is associated with considerable manual effort since a staff member has to enter plant planning information in the customization tool of the control system. Today, production it takes place at the end of the plant planning process. Potential planning errors are not recognized until the functionality of the production monitoring and control system is tested in the real-world plant. This is very time and cost-intensive. The vision is a plant which can simply be linked with the process by 'plug-and-produce'. The pre-requisite is a self-description containing all information required by the production monitoring and control system. There has to be a consistent, standardized data exchange format that can be used throughout all phases of the life cycle. The potential data sources that have been identified for control technology are the digital factory and the preceding planning phases (see (Bär et al., 2008)). Production monitoring and control system engineering of the future will therefore require a consistent neutral data format. In addition, standardized communication and processing mechanisms are necessary. Various sectors of industry have already specified their formats in detail, e. g. the Weihenstephan standards. Since the approach presented here is to be applied universally, the standards that have been developed for a specific sector and that can only be applied there are unacceptable. The existing XML-based standards in automation technology are largely limited with respect to the potential quantity of information that can be processed. NE100, FDCML and EDDL, for instance, are restricted to information about field devices. However, to obtain all the information required by control technology, data from multiple sources is required. This includes information about the process signals from the relevant field devices as well as plant visualization data and topology information from plant planning or from layout planning. For this reason, a universal format allowing all information from various sources to be collected and retrieved, as necessary, is needed. To avoid the development of just another new, hardly accepted data format, an existing industry standard should be used rather than establishing a completely new one. The independent XML-based CAEX (Computer Aided Engineering Exchange, (IEC 62424)) data exchange format was originally developed in process engineering. Fraunhofer IITB, however, has proved that the format is also suited for the efficient exchange of data in production engineering. In CAEX, the plant data can be managed and transmitted using library structures. This allows norms and standards to be modelled as CAEX libraries as well, satisfying those users that have already made their data comply with these standards. CAEX data can be processed further on the basis of a knowledge-based system, universal rules or analytical, computer-based tasks. The format takes account of the problem that the standardization of the tools available on the market only makes sense to a certain degree, if it makes sense at all. Therefore, a general approach has to be chosen. The exchange of CAE planning data between various systems is structured and organized, and in this application it is used as a standardized data format for the automated engineering of production monitoring and control systems.

In this application, CAEX was complemented by OPC UA (OPC Unified Architecture, (IEC 62541)), the service-oriented successor to the industrial OPC communication standard. OPC UA allows for the communication, synchronization and processing in the prototypical engineering framework. The architectural approach is based on a central OPC UA server and its address space, on which the individual components of the production monitoring and control system, of the associated customization and the process visualization act as OPC

UA clients. The underlying idea is a consistent standard interface, standardized communication with all systems involved, service-oriented processing, investment protection in view of supplier-specific formats and ultimately the enhancement of the quality of data through automated processes.

Combining both standards to form one framework boosts the strengths of each and opens up new potentials for the red-hot topic of "automation of automation". This chapter will present a solution which combines the two standards to form one framework.

2. The CAEX data format (Computer Aided Engineering Exchange)

Now that proprietary, stand-alone software solutions have been superseded, industry calls for a consistent information flow between the individual systems (see Fig. 1). As (RWTH, 2008) has shown, the definition and implementation of universal standard is only possible and useful to a certain degree. The main challenge lies in the vast complexity of application models and aspects.

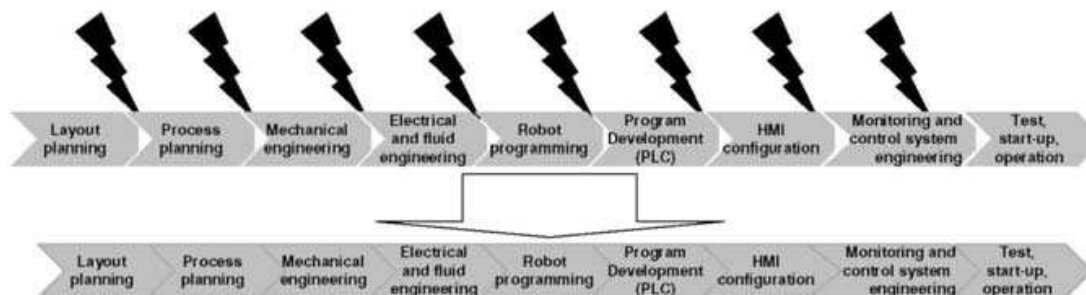


Fig. 1. Consistency of data, a prerequisite for boosting efficiency

CAEX was developed in cooperation between the Department of Process Control Engineering of the University of Technology Aachen, Germany (RWTH Aachen) and the ABB Research Center in Ladenburg. The definition of CAEX has been taken down in the (IEC 62424) standard. CAEX is a semi-formal description language which is based on XML (Extensible Markup Language). The requirements for a data exchange format have been specified in (Draht & Fedai, 2004a). First and foremost, the format should support library concepts and object-oriented approaches. It should be possible to integrate libraries from users and suppliers as well as project libraries. In addition, both a top-down and a bottom-up system design has to be supported.

(Epple, 2003) describes the motivation behind the development of CAEX. It is supposed to be independent of specified standards and protect investment in view of supplier-specific formats. Furthermore, CAEX is to boost the efficiency in data exchange. CAEX originated in process engineering. Nevertheless, it is equally suitable for production or manufacturing technology, as (Schleipen et al., 2008) has shown. In line with the aforementioned requirements, CAEX has been designed as a tool-neutral, object-oriented data format for storing hierarchical plant information. The exact structure and elements of CAEX are described in more detail in (Draht & Fedai, 2004b). CAEX supports concepts such as encapsulation, classes, instances, inheritance, hierarchies and attributes. The CAEX format has been defined as an XML schema.

The plant data can be managed and transmitted in the form of library structures. This allows norms and standards to be modeled as CAEX libraries as well, thus meeting the need of

those customers whose data comply with these norms already. CAEX data can be processed further by means of a knowledge-based system using universal rules or by means of analytical, computer-based tasks. In this context, the abstraction from a specific equipment entity or a specific data structure is essential.

CAEX allows for the standardized storage of planning data (in syntactical terms), supports object-oriented concepts and even enables the storage of unfinished planning stages. The meta modeling techniques simplify the creation of a standard data exchange. Using standardized interfaces, every system involved has to know all structures or possible data which can be exchanged. Instead of using generalized structures, CAEX defines its own model structures, which are managed in libraries. During data exchange, the CAEX libraries can be transmitted to other systems together with the data. All systems supporting the CAEX data model are capable of exporting the structural and semantic information from the libraries and to interpret the received data. This reduces complexity. Furthermore, the format is extremely scalable. This is useful while modeling whether complex production lines or one single machine. There are multiple advantages of a standardized, tool-independent format like this. Engineers, for example, can re-use valuable engineering results from existing systems, either in full or in part. Thus, they benefit from the experience of the past when planning a new plant or carrying out a new project. In view of the increasing time and cost pressure, this is a significant advantage over starting from scratch. A standardized data format renders production systems more transparent, which makes them easier to compare. In addition, a consistent format facilitates the computer-assisted evaluation and processing of these CAEX data.

At the top level, the CAEX data model (Fig. 2) consists of four areas: three libraries - InterfaceLibrary, RoleLibrary and SystemUnitLibrary - and the specific plant structure - SystemHierarchy.

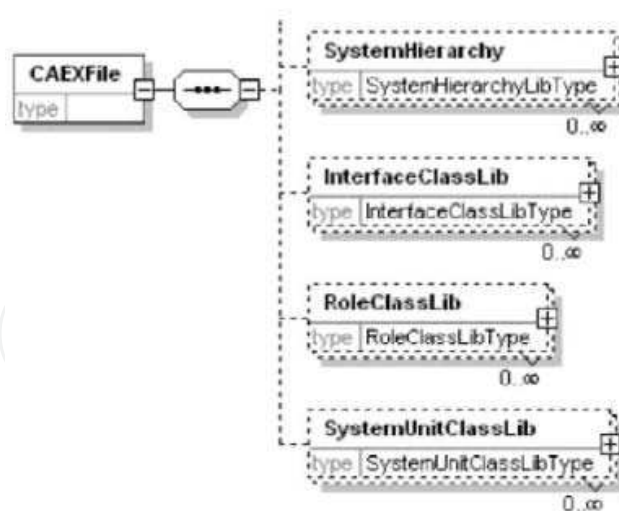


Fig. 2. CAEX data model (top level)

The libraries are specification collections. The 'SystemHierarchy' models the actual topology of the plant. To represent the plant, the instances of the models predefined in the libraries are used, which are then added by up-to-date parameter values. The basic components can be interlinked by means of various pre-defined XML elements within CAEX, allowing various semantic links to be modeled. Interfaces can be allocated to elements from the

RoleLibrary, for example, enabling the creation of links. Elements from the SystemHierarchy get an internal structure through allocated SystemUnits and a functional meaning through allocated roles. As far as the entire CAEX file is concerned, its structure is shown in Fig. 3. It will visualize the connection between the individual basic components and serve as an illustration for the explanations below.

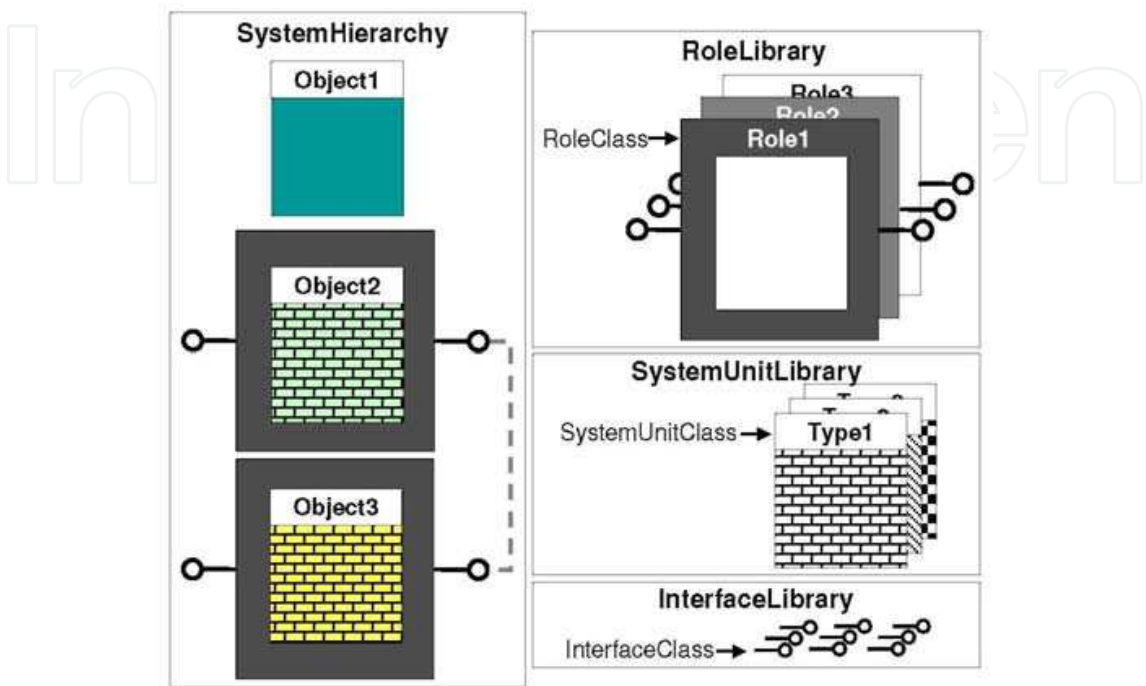


Fig. 3. Basic CAEX components and their connections

The InterfaceLibrary is the library of the interface classes. A class defines a type of interface, the link between two elements. The implemented interfaces allow two objects to be interconnected. The interface determines the type of relation and the semantic meaning of this connection. Interfaces may contain attributes. A plant topology interface, for example, may possess the attribute 'direction'.

In the RoleLibrary, the functional roles of the objects are described, e. g. a conveyor belt, a welding cell or a turntable. The role of an equipment component does not possess any information about the internal structure of the system. Instead, it describes the semantics as well as the available interfaces and general attributes. The abstraction through the role allows the symbolic parameter of a conveyor belt, for example, and thus the graphical object for a conveyor belt, to be assigned to a piece of equipment. The technical realization, the installation and implementation can be carried out at a later point in time, independently from previous steps. Nevertheless, this feature enables all conveyor belts, for example, to be visualized by a consistent graphical object. Both a unidirectional and a bidirectional conveyor belt, for instance, may be allocated the role of 'conveyor belt'. The roles thus implement an 'I am a ...' relationship.

The SystemUnitLibrary describes complex physical or logical equipment components ('SystemUnits'). They are similar to classes in object-oriented programming. Unlike roles, these elements possess information regarding the functionality and the internal structure of the elements. These elements can be compared with types or, in other words, they can act as

a combination of assembly units. The 'SystemUnit' node introduces a new structural description. Each structure can be made up hierarchically and use other, predefined structures as sub-elements. Therefore, it is not necessary to define the contents of the individual sub-elements at the type level. For this purpose, there are system classes that go beyond the usual CAEX structures in the application presented here. The system classes describe the elements that exist within the system in detail, including the contents and the meaning they have. The type only refers to this. In addition, the equipment element can implement a role defined in the role class library. The role allocation may result in other role-specific attributes and interfaces being added to the element, determining the way in which it interacts with other elements. The SystemUnitLibrary only defines the structure of the elements, while it does not allocate any attribute values. The only possible value specifications in the structural definition are those of the default values for the attributes. However, they only serve as initial values and thus form part of the definition; they may be overwritten in the description of the specific plant.

In addition to the aforementioned libraries, the CAEX meta model has a SystemHierarchy, where specific objects (InternalElements) and their planning data are managed and stored hierarchically. If a SystemUnit has been defined in the SystemUnitLibrary, a specific object of this system unit can then be included in the SystemHierarchy. This allows the real-world plant to be viewed as a structure made up of instances of SystemUnits. However, it is also possible to define instances without underlying SystemUnits. Subsequently, specific values are allocated to the attributes of the elements. In addition, the connections (called InternalLinks) between the elements can be modeled. They allow the plant topology, specific process sequences or assembly sequences, for instance, to be mapped. A good example of this kind of connection would be the link between an equipment component and its successor or specific products manufactured there.

The CAEX data model applied by Fraunhofer IITB has one more special feature. To allow for the electronic description of all 'elements' involved in the production process, these are broken down into three categories in practice: resources, products and processes. Control technology usually focuses on resources. The engineering framework classifies all existing system elements into three categories - products, processes and resources (PPR approach). Products stand for all products and product components, processes include all kinds of activities, and resources comprise equipment, staff, software, etc. This classification brings about additional semantic meaning for the system elements, such as 'I am a product', 'I am a resource' or 'I am a process'. The individual types of elements can be linked with each other, with resources being the central component in this model as resources execute processes and resources process products.

In addition to describing the production systems more accurately, the further classification is also aimed at a more detailed description of the products and processes involved. According to (Schleipen et al., 2008), this makes sense because this information is then available for the 'control technology of the future' or for 'components of manufacturing execution systems (MES)'. For the classification, the RoleLibrary, the SystemUnitLibrary and the SystemHierarchy of the CAEX model are relevant. In these areas, the structures of all three types - products, processes and resources - can be defined and/or applied. The InterfaceLibrary allows interfaces available to all types of elements to be defined. For this reason, it is not broken down into different categories. The classification takes place at the highest level in each case, before the individual elements are defined or deployed. This

ensures that all elements defined at a lower hierarchy inherit the semantic meaning of the common parent node. Fig. 5 shows how the data is embedded in the CAEX PPR model. In spite of the enhancements introduced by the description semantics developed by Fraunhofer IITB, the resulting CAEX file complies with the CAEX schema, of course.

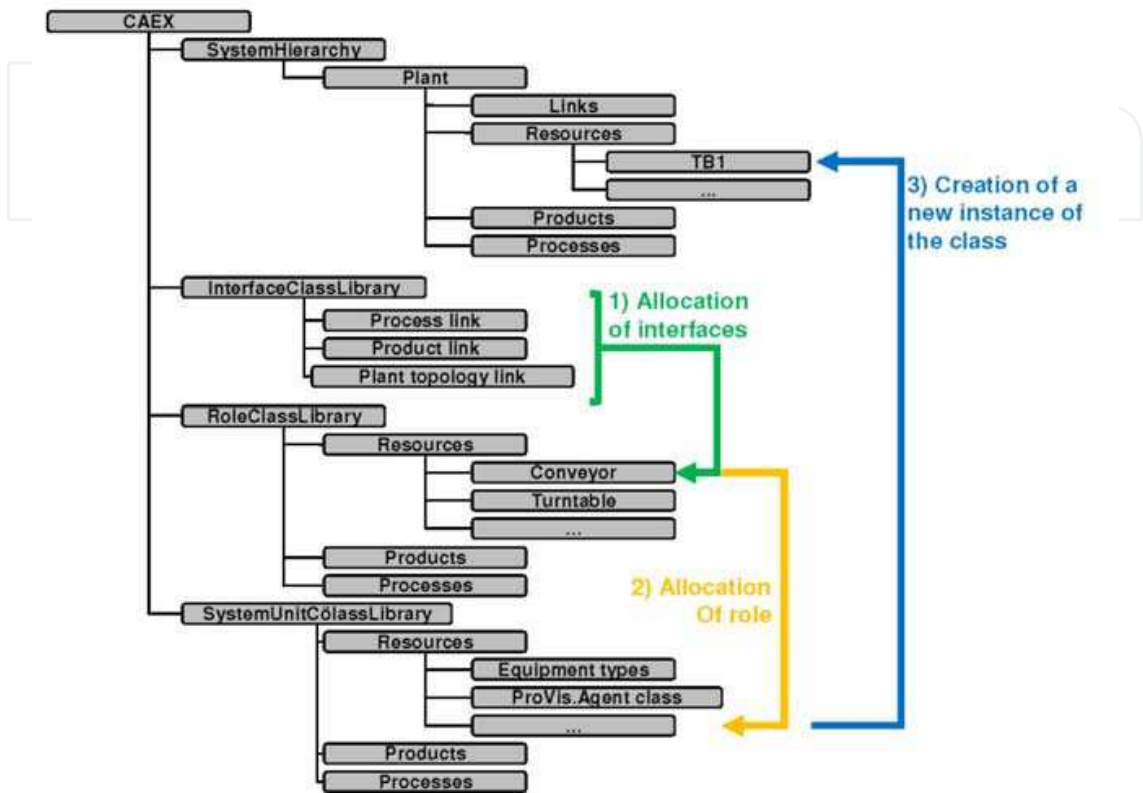


Fig. 4. CAEX PPR data model

The InterfaceLibrary is broken down into the elements 'plant topology link', 'product link' and 'process link'. The interfaces defined in this area allow two elements of the other structures to be interconnected. To be able to link two elements with each other using InternalLinks, both of them have to implement the same type of interface, i. e. support the same kind of connection. The 'plant topology link' enables two resources to be linked. The 'product link' connects a resource with a product, while the 'process link' links a resource with a process. Fig. 5 shows a sample situation in which the TBI conveyor belt transports a car shell and is topologically linked with the DT1 turntable. Interfaces are not assigned directly to the elements; instead, they are allocated by means of the functional roles. Roles are defined in the RoleLibrary. On the one hand, roles specify the class to which an element belongs, such as conveyor belt or turntable. On the other hand, they determine the way in which it can communicate with other elements by implementing the interfaces. At the topmost level, all roles are classified into product, process and resource roles. Each lower level implies a more detailed specification of the role features. This behaviour is in line with the concept of inheritance in object-oriented programming. By being allocated a role, the 'TBI' element (Fig. 5), for example, gets the following information. 'I am a resource, namely a conveyor belt. I can execute processes, process products and form part of a plant topology.'

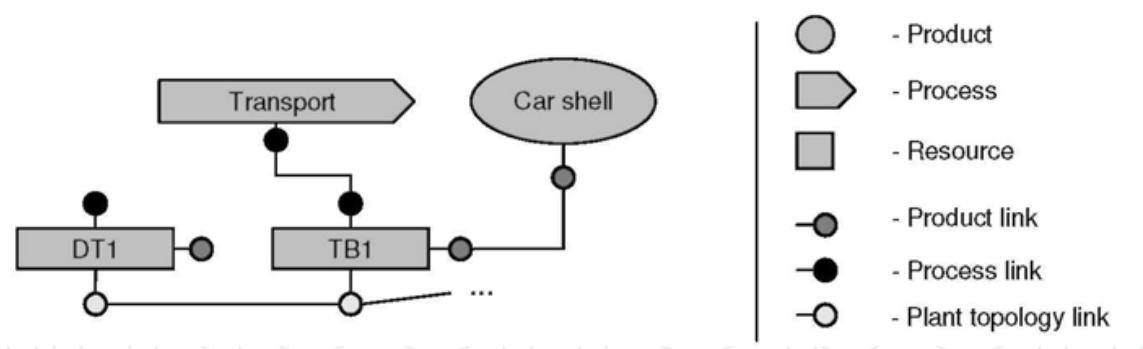


Fig. 5. Linking the elements by means of the implemented interfaces

The SystemUnitLibrary is also broken down into product, process and resource. In addition, the three top categories each have a class for their type basis. The resource class thus has an 'equipment type basis', where the available types of equipment are defined. This is where structural templates are specified as well, which reflect the basic structure of the tool. In the case of 'ProVis', this would be a 'ProVis' class, for example, which describes all potential types of process variables.

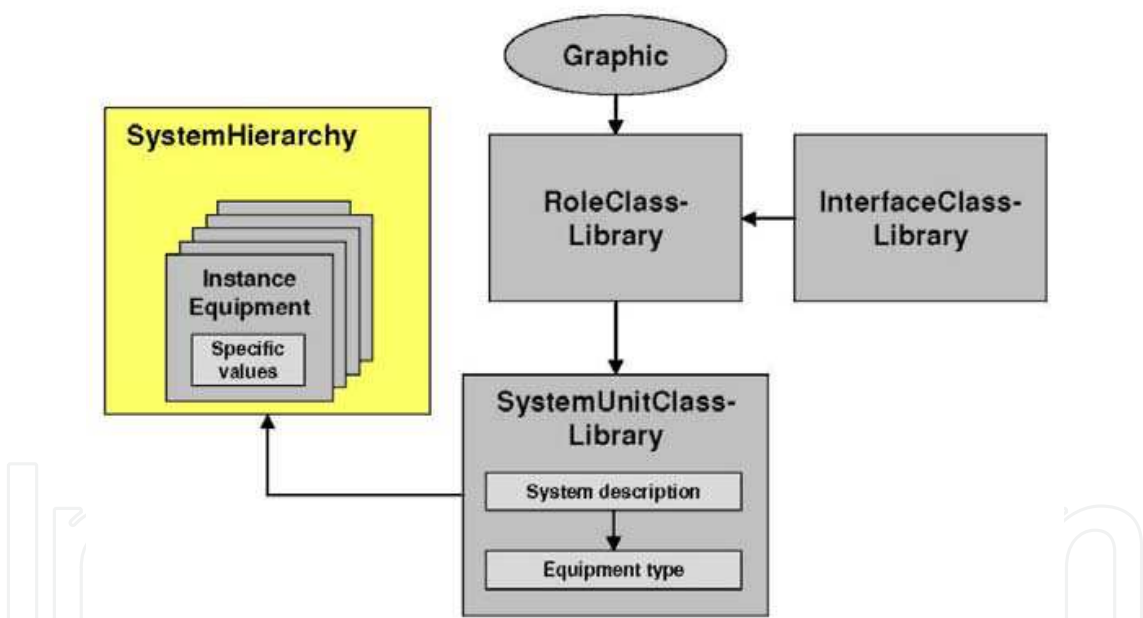


Fig. 6. Creating a specific equipment object

To create a specific equipment object, e. g. a 'conveyor belt', the following steps should be taken. First of all, a type of equipment defined in the SystemUnitClassLibrary is instantiated. A role is, in turn, allocated to the type of equipment. Furthermore, it is possible to assign a graphical object to the role, which allows the conveyor belt to be visualized graphically (see Fig. 6). Within the framework of the SystemHierarchy, specific values and attributes are assigned to this element.

3. The OPC UA communication standard (OPC Unified Architecture)

In 1996, the specification of standardized communication originated. It was meant as an alternative to proprietary automation buses and allow for communication between applications stemming from different suppliers. This is how OLE for Process Control (or OPC) was created. OPC, however, does not stand for just one partial functionality. Instead, it provides multiple features. These include OPC Data Access (DA), which enables PLCs to transmit real-time data to visualization devices. Other examples are Alarm&Events, Batch, Historical Data Access or XML Data Access. The OPC specification is implemented by means of software. To enable communication between applications, OPC relies on DCOM (Microsoft Distributed Component Object Model), a state-of-the-art technology of that period. It has a client-server-based architecture in which the server provides data, which is, in turn, accessed by the client.

In view of the fact that the original OPC specifications had evolved historically and the technological capabilities had been enhanced in the course of the years, the OPC Foundation presented the successor of OPC (OPC Foundation, 2009) in 2006. The OPC Unified Architecture (or OPC UA) standardizes the existing specifications. Instead of continuing to use DCOM, OPC UA is based on a service-oriented architecture (in short SOA), which is independent of platforms and operating systems. OPC UA allows for both the horizontal and the vertical exchange of data in a multi-functional server. This ensures that it can be used more universally not only at a field level but also at the level of MES or ERP systems. In addition, OPC UA supports asynchronous and distributed communication while ensuring security, reliability and redundancy by providing appropriate features. At the heart of OPC UA (Fig. 7), there are the abstract method descriptions, also called base services, which are translated into a protocol by the transport layer.



Fig. 7. OPC UA model

The information model does not only consist of a hierarchy of 'folders', 'items' and 'properties' as was the case with OPC. Instead, it is a full-mesh network made up of nodes allowing all kinds of meta and diagnosis information to be transmitted. In this context, a node is an object containing attributes for read access (similar to Data Access (DA) and Historical DA). It owns methods that can be called (similar to Commands) and events that can be sent (similar to DA DataChange). The address space specified by the information

model includes a type model allowing all types of data to be described. On this basis, various other organizations such as EDDL are specifying their own information models. The integration of the CAEX meta model is another example of this trend. As a consequence, any world model can be modelled in the address space of the OPC UA server. Thus, there is no limit to the ways in which it can be applied in multiple sectors of industry.

Applications for OPC UA are developed with the help of APIs, which the OPC Foundation provides to software developers. Owing to the multitude of options OPC UA offers, it was chosen as the standard on which the communication within the framework should be based. The architectural concept of the engineering framework is based on a central OPC UA server and its address space, on which the individual components of the framework operate as OPC UA clients. The information model of the address space was modeled in line with recommendations for the specification (OPC Foundation, 2006). The clients (i. e. the individual components) are capable of integrating additional information in the address space, such as creating new nodes. In this context, the components only observe those parts of the address space that are relevant to them (see Fig. 8).

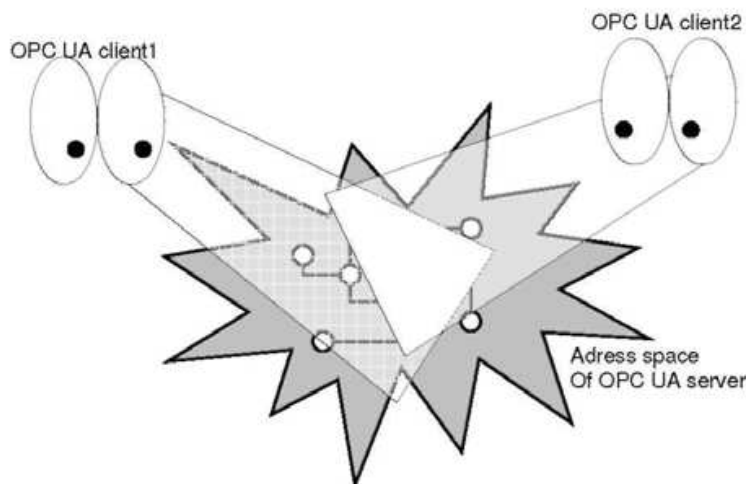


Fig. 8. Detail of the architectural concept

They are notified by the server as soon as some information that is important to them has been received. Hence, the OPC UA server supports the synchronization of processes in the monitoring and control system. If new, CAEX-compliant data is integrated in the engineering framework, the appropriate OPC UA client will instruct the server to create a new node. The client in charge of transforming the CAEX data is notified when the new node has been generated and receives the information it requires. This mechanism is identical for all other clients. The implementation is based on the associated base services (OPC Foundation, 2007). For further details, please see (Schleipen, 2008).

4. Combination of both standards: the engineering framework for the production monitoring and control system

If two systems are to communicate with each other, the two decisive issues are

- "What is communicated?" - The required contents have to be structured. In addition, the meaning of the contents have to be clear.

- "How does communication work?" - The communication mechanisms have to be specified. This includes both the process and the methods, etc.

CAEX is the answer to the first question. It structures data and fills them with semantic meaning. OPC UA, by contrast, is the answer to the second question. It assists communication and information processing in the engineering framework. In this context, CAEX is responsible for the static part, whereas OPC UA handles the dynamics of the procedures and data. The application of this combination of standards is not restricted to production monitoring and control systems and their process visualization. Instead, it can be applied in multiple ways (see Fig. 9).

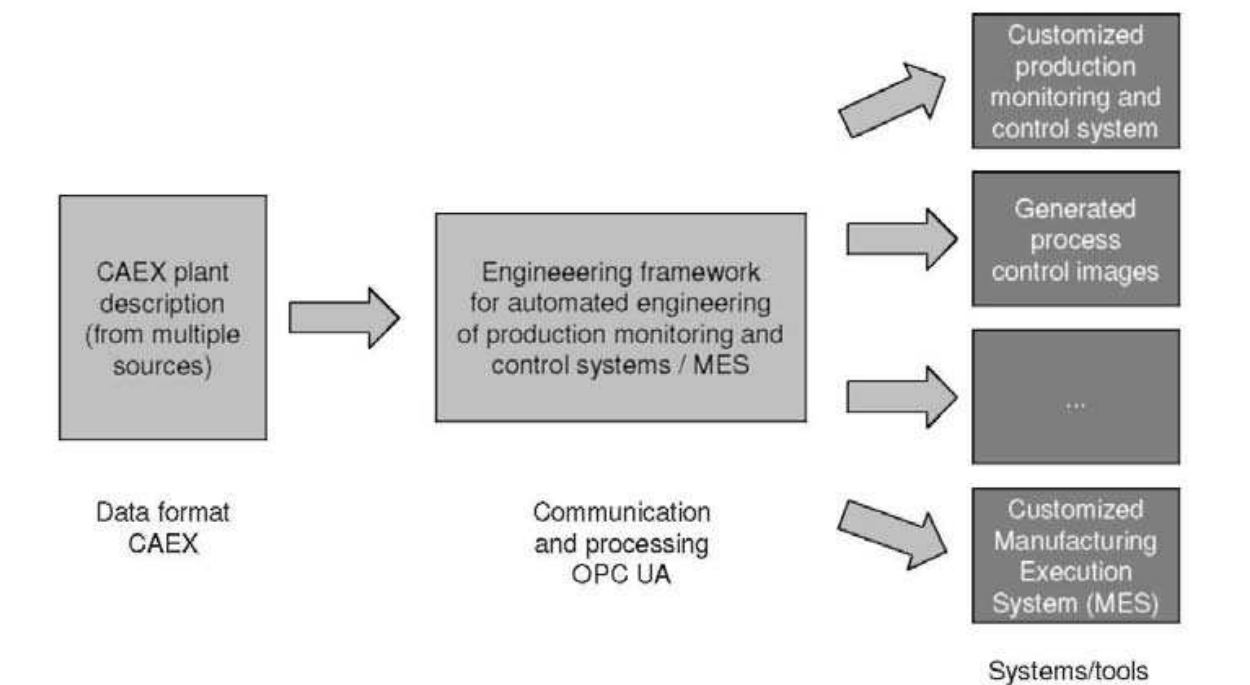


Fig. 9. Functionality of the engineering framework

To evaluate the engineering framework, various sample applications have been created. These include the customization of the production monitoring and control system and the generation of process images, which are parameterized by means of a layout manager and do not only represent resources but also additional products and processes. These sample cases have been tested using various sample systems. For one thing, there are various conveyor systems at hand to validate customization and simple visualization. They are shown in Fig. 10 and include conveyor belts and turntables as well as a test station or welding cell.

For another, a hierarchical example (Fig. 11) was developed to test the layout manager. The topmost hierarchical level consists of two equipment aggregates called TA1 and TA2. They are aggregations of several pieces of equipment. At the second hierarchical level, the TA1 equipment aggregate consists of two conveyor belts (TB1 and TB2), a DT1 turntable and another equipment aggregate called TA3. The TA2 equipment aggregate includes two conveyor belts (TB3 and TB4) and a DT2 turntable. The third and thus lowest hierarchical level is represented by the TA3 equipment aggregate. It is made up of a DT3 turntable, a TS1 test station and a TB5 conveyor belt. Thus, the sample application consists of twelve pieces

of equipment in total, three of which are aggregates (the equipment aggregates TA1 to TA3) and nine of which are 'genuine' pieces of equipment.

To allow for the overall visualization of resources, processes and products, one of the conveyor belt examples was complemented by processes and products (Fig. 12).

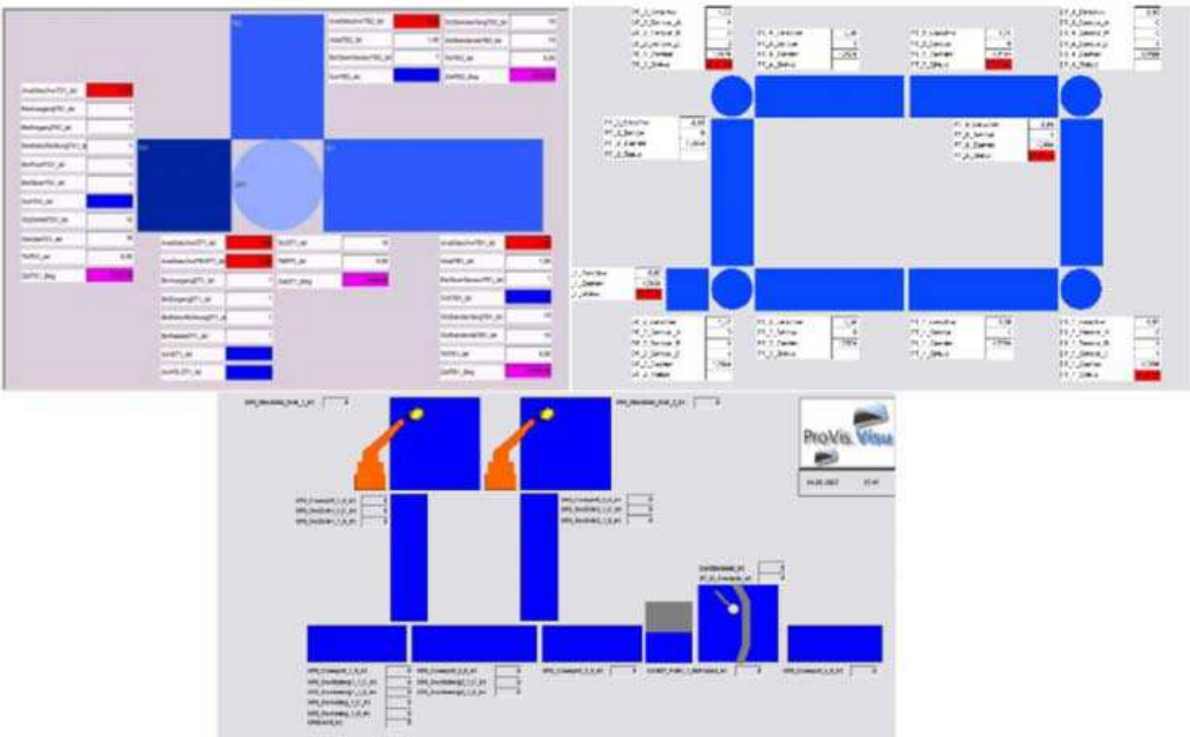


Fig. 10. Sample applications

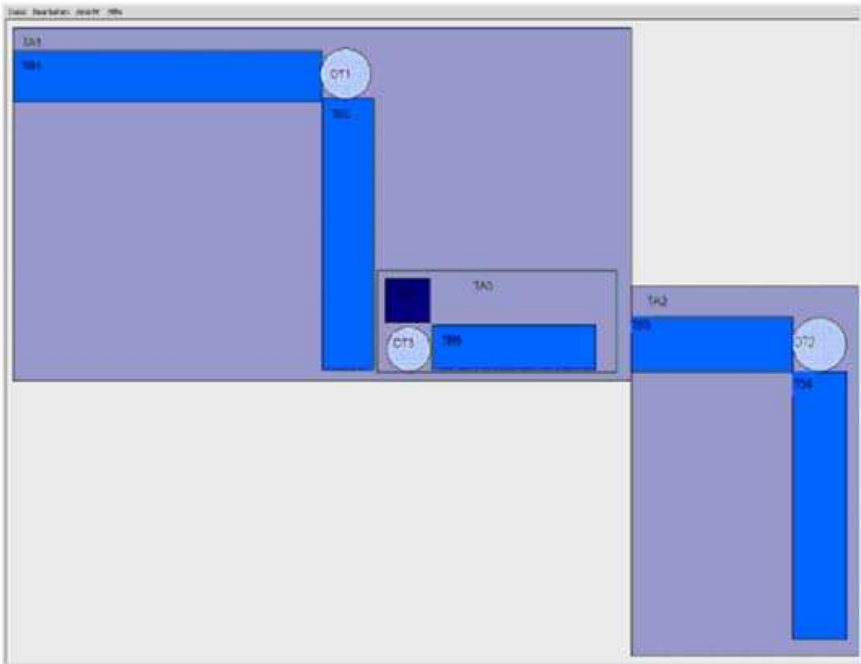


Fig. 11. Hierarchy-based sample application

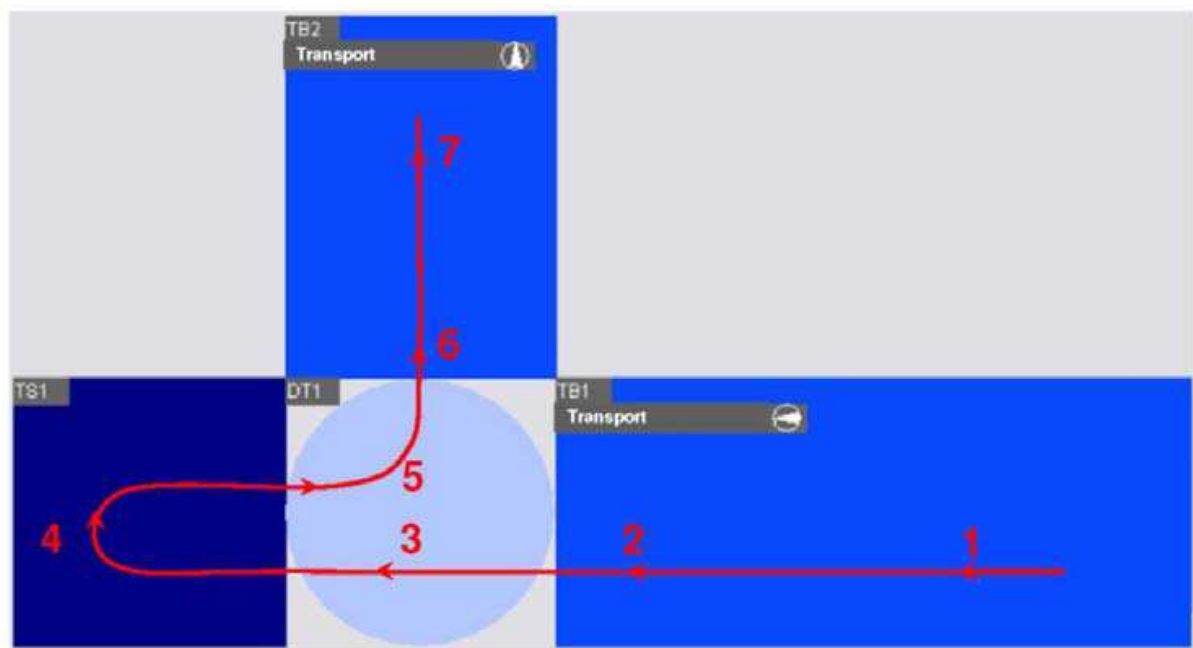


Fig. 12. Basic image of the sample application for resources, products and processes

The following paragraphs will outline the workflow applied for engineering within the framework. If a CAEX description is available for the equipment (see Fig. 13), it is transmitted to the change management of the production monitoring and control system on the basis of OPC UA.

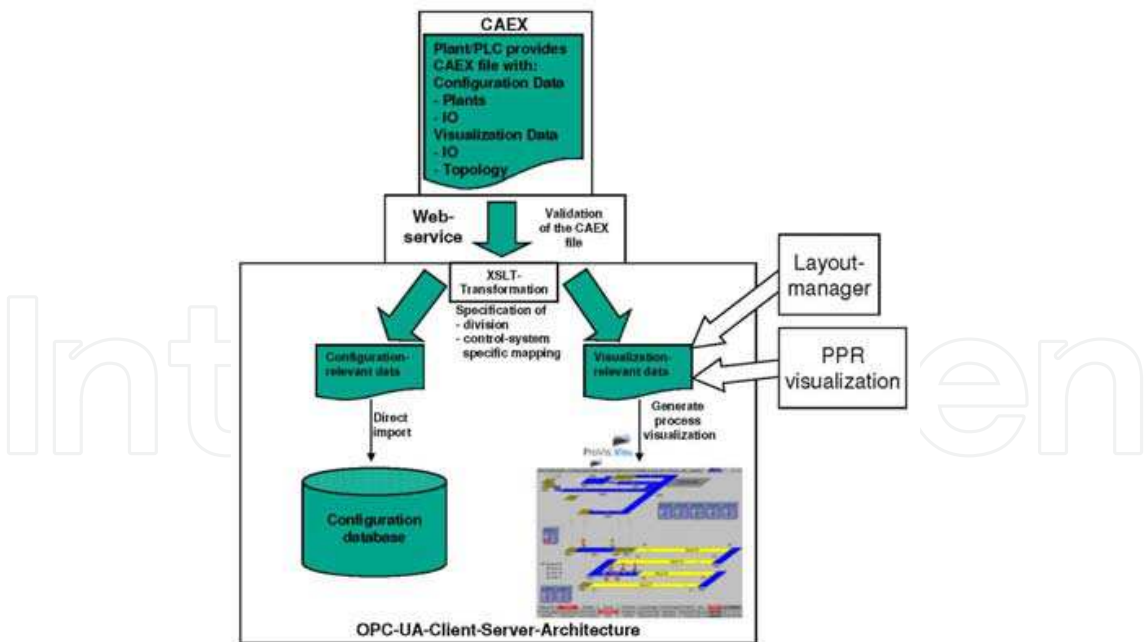


Fig. 13. The engineering framework

The provided CAEX data is validated against the CAEX XML schema with respect to structural correctness. Subsequently, it can be processed further on the basis of the structure and semantics. For this purpose, the mapping based on system descriptions or templates is

used to convert the data into a production monitoring and control system specific CAEX format. To generate this kind of mapping between two CAEX files from different suppliers and/or levels, the structural templates of the SystemUnitLibrary are considered. In the case of a mapping between a 'ProVis' CAEX file and the CAEX file provided by the equipment or PLC, the 'ProVis' class and the PLC class in which the available structures including all attributes are specified are considered. To make things easier, the data types and the units of the relevant attributes are specified there, too. At first sight, this resembles a proprietary interface. However, the mappings are independent of the actual data structure in the tool and can be generated and converted more easily thanks to XML and the predefined structures, since the converter using the mappings has to be created just once. In addition, graphical support is available, making things even easier for users.

Assisted by the central OPC UA server and several OPC UA clients, the pre-processed data is used both to customize the production monitoring and control system and to generate the process control images for ProVis.Visu® in an automated way using a layout manager. The CAEX document is split into data relevant for configuration and visualization (see left and right path of Fig. 13). The resulting sets of data are transmitted to the configuration and visualization components of the ProVis production suite respectively. The visualization data is used to generate the process images. The configuration-relevant part is imported into a database. The system can use this data to perform the I/O and plant customization for the process image of the runtime system (see (Schleipen et al., 2008)). This considerably reduces the manual and thus error-prone part of customization, as the production monitoring and control system (CS) plant configuration, the CS I/O customization and the CS image customization are, in part, performed automatically (also see (Sauer & Ebel, 2007)). In this process, the generation of the process control images as the human-machine interface has to be considered above all. In manufacturing, an automated system for image customization will only be accepted if the user interface is user-friendly and intuitive. The special field of human engineering (Syrbe, 1970) aims to adapt machinery and other technical equipment to humans to optimize their cooperation. The characteristics, potentials and requirements of human beings are taken into account, and the visualization of machinery and/or equipment is based on these conditions. For this reason, human engineering deals with both the physical/ physiological and the mental characteristics of human beings (Charwat, 1994).

In (Syrbe, 1970), the following seven rules are presented which form the basis of the high-quality design of the human-machine interface:

- "Mind the properties of the sense organs"
- "Depict process states in a task-dependant way"
- "Choose an attractive design which directly corresponds to the task"
- "Avoid information unnecessary for fulfilling the task (noise information)"
- "Mind the unconscious attention control of human beings"
- "Mind population-stereotypical expectations"
- "Design correlating display and operation elements in a strikingly similar way and those that do not correlate in a particularly divergent way"

In this process, visualization is to be based on ergonomic guidelines. In addition, appropriate algorithms have to be developed to position the existing equipment components as well as I/O signals on the process control image in line with the actual layout. Finally, users should be able to adapt the process control images to their personal requirements.

If users create process control images manually, the very same process may be depicted differently depending on the preferences of the person who has drafted them. In contrast to process engineering, there are no standards such as P&I diagrams in DIN EN ISO 10628 in production and manufacturing technology specifying the layout of certain components. As a consequence, the same processes do not necessarily look identical in visualization. In addition, the manual generation of images has the drawback that it is time-intensive and error-prone. Thus, the process control images should be as standardized as possible, while, at the same time, being as individual as necessary.

The layout of the visualization was defined in various views (Schleipen & Schick, 2008), allowing for a topological and a structural overview of the entire plant and making it possible to zoom into the equipment. Moreover, the equipment can be operated in line with the potentials of the system. The topological view visualizes the topology of the equipment to be monitored. In this context, it should be possible to zoom into the equipment. To this end, a hierarchical level model was created allowing several equipment aggregates to be combined to form a larger system. Fig. 14 illustrates this approach. It enables entire production halls to be visualized clearly in just one image while ensuring that the most important information such as faulty states in the aggregations, also called 'collective alarms', are displayed.

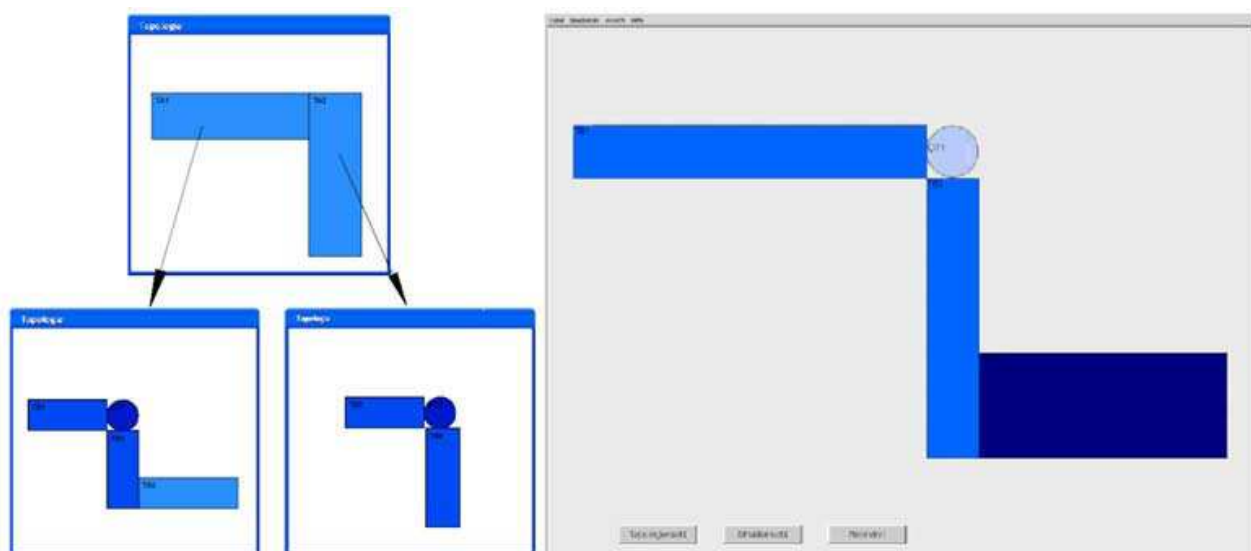


Fig. 14. Concept and implementation of the topological view (Schleipen & Schick, 2008)

The structural view (see Fig. 15) provides an overview of the signals of the existing pieces of equipment to users. Every line stands for a piece of equipment contained in the overall plant. The other elements represent the linked process variables, their slots and their current values.



Fig. 15. Structural view (Schleipen & Schick, 2008)

The operational view shown in Fig. 16 allows the users to operate the plant they monitor. It only displays the process variables of one piece of equipment rather than those of the entire equipment, as is the case in the structural view.



Fig. 16. Operational view (Schleipen & Schick, 2008)

The design of all views is based on ergonomic requirements (also see DIN EN ISO 9241-12). To enter the user-specific settings, a graphical user interface was created. Nevertheless, the basic structure is maintained when the process control images are generated. Otherwise, there would be the problem that the images vary considerably depending on who has created them, as was the case with the manual generation of the process images. For the topological view, users can determine the piece of equipment that forms the highest hierarchical level/the most interesting part of the plant. In a second step, it is possible to

define the process variables and slots that are to be represented in the structural and/or operational view. The last part, the 'representation', defines the color specifications or the path to the bitmap graphic that is to be used to visualize the equipment. This information can, in part, be extracted from the CAEX descriptions. In addition, users may store all the settings they have chosen.

In addition to visualizing equipment, ongoing processes and processed products have to be represented. An approach to the practice-oriented representation of products and processes has been developed and implemented. Combined with a product identification system, the products can thus be mapped at their current position. This allows the products and the progress of the process to be traced by linking ident systems, for example, with control technology. In addition to visualizing plants, in this component, participating processes and products can be visualized as well. This allows the products to be traced and the progress of the process to be monitored based on dynamically changed CAEX data. If the CAEX model is contained in the address space of the OPC UA server, it is at all times possible to identify the processes currently executed and the product processed by the system. The structure of elements, equipment and products within the images is made up dynamically, as is the allocation of products and processes to a resource (piece of equipment). To visualize movements of products and changes in current processes, it is required to map the current production situation at regular intervals. Changes in processes and product positions do not have to be visualized in real-time for production monitoring and control technology. To update product and process representations in process visualization, intervals of five seconds (or a maximum of ten seconds) are sufficient in this case. The process signals, by contrast, continue to be visualized in real-time. The presented information has to be as clear as possible, allowing even inexperienced users to interpret them intuitively. Image generation is to comply with the engineering general approach. In addition, the universality of the CAEX model should not be restricted by image generation. Since control technology only visualizes abstracted production process information, there is no need to visualize complete products or processes there. Rather, it is sufficient to visualize products as bitmaps at discrete positions of the resources. Text-based process information providing an option to access additional data will do. In addition, the visualization of the direction of the process (flow) is important because it can provide valuable hints for detecting potential faults in the production process.

The resource and process names are shown in a text field. To ensure that the provided information does not conceal the image elements located next to the resource, they are positioned in the top left corner of the resource. For visibility and readability reasons, the texts are presented against a neutral background in the form of information bars. Depending on the information provided by the resource, the relevant bar is either shown or hidden completely. The layout of the information bars consists of a dark gray background and white fonts. This colour combination ensures that the text can be read clearly (see Fig. 17, top).

5. Conclusion

The engineering framework presented in this contribution allows data to be processed and communicated electronically for production monitoring and control technology. This ensures that a fault-resistant, semi-automated production monitoring and control system engineering can be realized. Hence, control technology as a representative of IT systems in plant operation can be linked with planning at an early stage. Fig. 19 shows the benefit of the earlier coupling of planning and the customization of production monitoring and control systems. It increases efficiency in time and thus costs in the delivery and customization of production monitoring and control technology

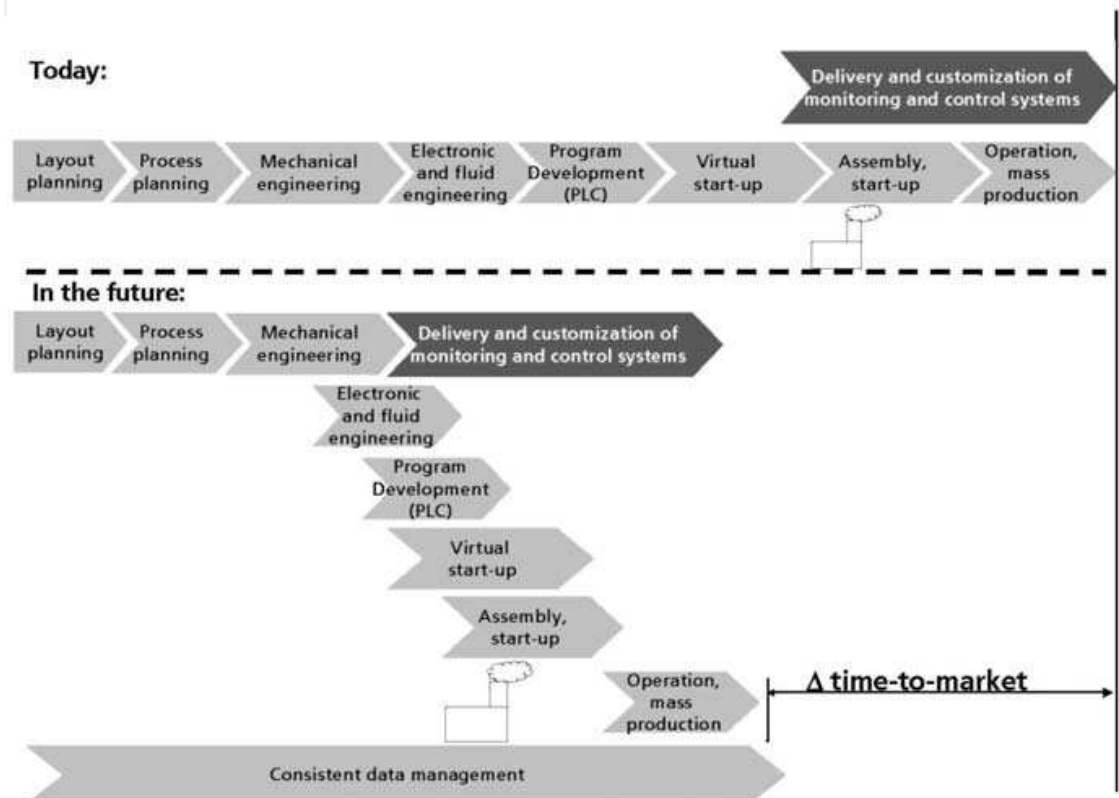


Fig. 19. Early coupling of planning and the customization of production monitoring and control systems according to (VDI 4499-2)

In parallel to the engineering framework, there are new potential fields of deployment. When starting up a system, complex production monitoring and control system can be parameterized and tested using simulations even before the software is actually taken into operation. To this end, the system simulation has to be controlled by a PLC that does not form part of the simulation program so that control technology can access the simulation signals. Control technology can then be based on these real signals using the well-known communication mechanisms of automation technology, e. g. OPC. Fig. 20 provides a schematic overview of this kind of coupling. For the production monitoring and control system, it is insignificant whether the OPC signals stem from a real-world production process or from a PLC linked with simulation. If this kind of link is in place, the data processed and/or generated in the production monitoring and control technology can be used to improve the input data of simulation. This enables control technology to be tested at

an early stage. Furthermore, it serves as an additional data source for simulation. This type of data includes evaluations from the production monitoring and control system, for instance. An evaluation of the process data in the production monitoring and control system allows for the provision of quality features for executable configurations in simulation.

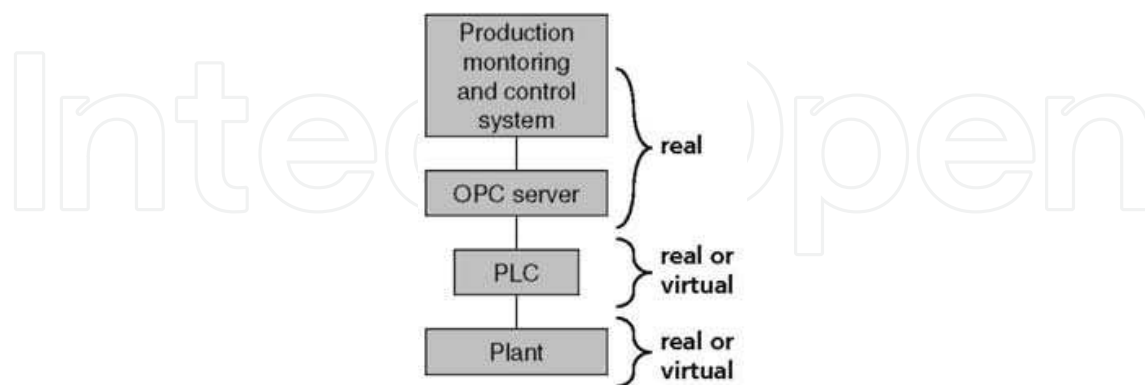


Fig. 20. Link between simulation and production monitoring and control system using OPC

The development of the engineering framework includes considerations regarding the legal consequences that result from the findings for the individual users. These consequences can be classified as follows: contractual problems, product liability, safety of equipment and industrial law. As a general rule, the results generated automatically should, at any rate, be approved by technical specialists, in accordance with legal experts. Staff members should receive appropriate training and be familiar with topics such as responsibility for defaults, product liability and CE marking to be able to perform a plausibility check for the created software and configuration and to judge it. In this process, they can be supported by a checklist that has to be observed in this case. This ensures that important and necessary topics and aspects are taken into account. As far as liability is concerned, there are various parties dealing with these topics or problems. These include the software suppliers who are liable for the software they provide. In addition, they include the manufacturers or suppliers of machinery who are responsible for the machine. Finally, it is the operator of the plant who is liable once the system has been taken into operation. Vis-à-vis the final customer, these parties may have joint liability and take responsibility for flaws in the products produced. As a consequence, there are aspects other than technological potentials that play an important part and that have to be considered and observed.

6. References

- Bär, T.; Mandel, S.; Sauer, O.; Ebel, M. (2008) Durchgängiges Datenmanagement durch plug-and-work zur virtuellen Linieninbetriebnahme, *Proceedings of 2. Karlsruher Leittechnischen Kolloquium*, pp. 105-122, 978-3-8167-7626-0, Karlsruhe, Mai 2008, Fraunhofer IRB Verlag, Stuttgart
- Charwat, H. J. (1994) *Lexikon der Mensch-Maschine-Kommunikation*, Oldenbourg, 3486226185, München
- Drath, R.; Fedai, M. (2004) CAEX - ein neutrales Datenaustauschformat für Anlagendaten - Teil 1, *atp - Automatisierungstechnische Praxis*, Vol. 46 (2004), No.2, (February 2009) 52-56, 0178-2320

- Drath, R.; Fedai, M. (2004) CAEX - ein neutrales Datenaustauschformat für Anlagendaten - Teil 2, *atp - Automatisierungstechnische Praxis*, Vol. 46 (2004), No.3, (March 2009) 2027, 0178-2320
- DIN EN ISO 9241-12:2000-08 Ergonomische Anforderungen für Bürotätigkeit mit Bildschirmgeräten - Teil 12: Informationsdarstellung (ISO 9241-12:1998), Deutsche Fassung EN ISO 9241-12:1998, Beuth, Berlin
- DIN EN ISO 10628:2001-03 Fließschemata für verfahrenstechnische Anlagen - Allgemeine Regeln (ISO 10628:1997, Beuth, Berlin)
- Drath, R. (2008) Die Zukunft des Engineering - Herausforderungen an das Engineering von fertigungs- und verfahrenstechnischen Anlagen, *Proceedings of 2. Karlsruher Leittechnischen Kolloquium*, pp. 33-40, 978-3-8167-7626-0, Karlsruhe, Mai 2008, Fraunhofer IRB Verlag, Stuttgart
- Epple, U. (2003) Austausch von Anlagenplanungsdaten auf der Grundlage von Metamodellen, *atp - Automatisierungstechnische Praxis*, Vol.45 (2003), No.7, (July 2009) 61-70, 0178-2320
- Fay, A.; Schleipen, S.; Mühlhause, M. (2009) Wie kann man den Engineering-Prozess systematisch verbessern?, *atp - Automatisierungstechnische Praxis*, Vol.52 (2009), No.1-2, (January 2009) 80-85, 0178-2320
- IEC 62424: Specification for Representation of process control engineering requests in P&I Diagrams and for data exchange between P&ID tools and PCE-CAE tools, text English.
- IEC 62541: OPC Unified Architecture
- OPC Foundation (2006) OPC UA Part 3 - Address Space Model 1.00 Specification, July 2006
- OPC Foundation (2007) OPC UA Part 4 - Services DRAFT 1.01.05 Specification, February 2007
- OPC Foundation (2009) OPC Unified Architecture, <http://www.opcfoundation.org>, April 2009
- Polke, M. (edit.) (1994) *Prozeßleittechnik*, Oldenbourg Verlag, 3486225499, München
- RWTH Aachen, Lehrstuhl für Prozessleittechnik (2008) ACPLT: CAEX-IEC62424, <http://www.plt.rwth-aachen.de/index.php?id=228&L=1ks>
- Sauer, O.; Sutschet. G. (2006) ProVis.Agent: ein agentenorientiertes Leitsystem - erste Erfahrungen im industriellen Einsatz, *Proceedings of VDE-Kongress 2006*, pp. 297302, 978-3-8007-2979-1, Aachen, October 2006, VDE Verlag, Berlin-Offenbach
- Sauer, O.; Ebel, M. (2007) Engineering of production monitoring & control systems, *Proceedings of 52nd IVK - Computer science meets automation*, pp.237-244, 3939473170, Ilmenau, September 2007, TU Ilmenau Universitätsbibliothek, Ilmenau
- Schleipen, M. (2008) OPC UA supporting the automated engineering of production monitoring and control systems, *Proceedings of 13th IEEE International Conference on Emerging Technologies and Factory Automation ETFA*, pp. 640-647, 1-4244-1506-3, Hamburg, September 2008, IEEEPress
- Schleipen, M. ; Drath, R.; Sauer, O. (2008) The system-independent data exchange format CAEX for supporting an automatic configuration of a production monitoring and control system, *Proceedings of IEEE International Symposium on Industrial Electronics - ISIE 2008*, pp.1786-1791, 978-1-4244-1666-0, Cambridge, June 2008, IEEEPress

- Schleipen, M.; Schick, K. (2008) Self-configuring visualization of a production monitoring and control system, *Proceedings of CIRP International Conference on Intelligent Computation in Manufacturing Engineering - CIRP ICME 08*, 978-88-900948-7-3, Naples, July 2008
- Syrbe, M. (1970) Anthropotechnik, eine Disziplin der Anlagenplanung, *Elektrotechnische Zeitschrift*, Vol. 91 (1970) 12, No. A, (1970) 692-697, 00137359
- VDI 4499-2: Digitaler Fabrikbetrieb, Gründruck, 2009



Factory Automation

Edited by Javier Silvestre-Blanes

ISBN 978-953-307-024-7

Hard cover, 602 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

Factory automation has evolved significantly in the last few decades, and is today a complex, interdisciplinary, scientific area. In this book a selection of papers on topics related to factory automation is presented, covering a broad spectrum, so that the reader may become familiar with the various fields, and also study them in more depth where required. Within various chapters in this book, special attention is given to distributed applications and their use of networks, since it is one of the most relevant subjects in the evolution of factory automation. Different Medium Access Control and networks are analyzed, while Ethernet and Wireless networks are looked at in more detail, since they are among the hottest topics in recent research. Another important subject is everything concerning the increase in the complexity of factory automation, and the need for flexibility and interoperability. Finally the use of multi-agent systems, advanced control, formal methods, or the application in this field of RFID, are additional examples of the ideas and disciplines that experts around the world have analyzed in their work.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Miriam Schleipen (2010). Automated Production Monitoring and Control System Engineering by Combining a Standardized Data Format (CAEX) with Standardized Communication (OPC UA), Factory Automation, Javier Silvestre-Blanes (Ed.), ISBN: 978-953-307-024-7, InTech, Available from:

<http://www.intechopen.com/books/factory-automation/automated-production-monitoring-and-control-system-engineering-by-combining-a-standardized-data-form>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen