Chapter 2
Multi-Disciplinary Engineering for Industrie 4.0: Semantic Challenges and Needs

Stefan Biffl, Arndt Lüder and Dietmar Winkler

Abstract This chapter introduces key concepts of the *Industrie 4.0* vision, focusing on variability issues in traditional and cyber-physical production systems (CPPS) and their engineering processes. Four usage scenarios illustrate key challenges of system engineers and managers in the transition from traditional to CPPS engineering environments. We derive needs for semantic support from the usage scenarios as a foundation for evaluating solution approaches and discuss Semantic Web capabilities to address the identified multidisciplinary engineering needs. We compare the strengths and limitations of Semantic Web capabilities to alternative solution approaches in practice. Semantic Web technologies seem to be a very good match for addressing the aspects of heterogeneity in engineering due to their capability to integrate data intelligently and flexibly on a large scale. Engineers and managers from engineering domains can use the scenarios to select and adopt appropriate Semantic Web solutions in their own settings.

Keywords Engineering process · Systems engineering · Cyber-Physical production systems · Semantic Web · Data integration

---

S. Biffl · D. Winkler (✉)
Institute of Software Technology and Interactive Systems,
CDL-Flex, Vienna University of Technology, Vienna, Austria
e-mail: dwinkler@sba-research.org; dietmar.winkler@tuwien.ac.at

S. Biffl
e-mail: stefan.biffl@tuwien.ac.at

A. Lüder
Otto-von-Guericke University/IAF, Magdeburg, Germany
e-mail: arndt.lueder@ovgu.de

D. Winkler
SBA-Research gGmbH, Vienna, Austria

S. Biffl and M. Sabou (eds.), *Semantic Web Technologies for Intelligent Engineering Applications*, DOI 10.1007/978-3-319-41490-4_2

2.1 Introduction

Production systems of any kind have to face two main drivers for evolution, (1) technical developments related to useable technologies and (2) customer requirement developments. The latter driver typically leads to product diversification. Related to technical development, especially development in the IT industry, increased communication capabilities have an important impact (Jacob 2012) and lead to the need of faster evolution of production systems (Kagermann et al. 2013).

Production systems are means for the creation of products. Following Grote and Feldhusen (Grote and Feldhusen 2014), the aim of a production system is the value-generation-based creation of goods, i.e., the creation of products. Therefore, the starting point of understanding of the engineering needs of production systems is the product. A *product* is defined as an item that ideally satisfies customer needs related to its application (Grote and Feldhusen 2014). These needs are considered within the product design as input providing main requirements to product design. In addition, product design has to consider technical capabilities of production systems as bordering conditions. Examples of products are items, which are of interest for an end user, such as cars, washing machines, mobile phones, clothes, or food, with product-related customer expectations, such as passenger safety, low energy and water consumption, internet access, style, or taste. In addition, electrical drives, cement, or oxygen need to be seen as products with more technical customer requirements, such as integrability into a production system, speed of setting, or usability in medical systems.

Within the *product design*, which has to be seen as a creative process of engineering the product: on the one hand the structure, visual nature, behavior, or functionality of the product are defined; on the other hand the way to create the product, i.e., the process, needs to be developed. This includes the definition of materials to be used and the definition of production steps to be executed. Materials cover all physical and nonphysical elements to be purchased for production, such as steel plates for cars or agriculture objects for food production. Production steps cover all required value-adding actions needed within the production process. Examples are welding of steel plates, mounting of components in car manufacturing, or cleaning and cooking in food processing.

Figure 2.1 models these dependencies between the product and the production system. Based on the defined way of product creation, the *production system* can be engineered. Therefore, under consideration of technological and economical possibilities, the set of required production steps is translated by a team of engineers coming from different engineering disciplines into a set of *production system resources* that are able to execute the production steps on the defined materials. Examples of such resources are welding robots required for steel plate welding, human workers required for the mounting of components in car manufacturing, a washing belt for agriculture product cleaning, and a steamer for cooking in food processing. These resources will be engineered in detail, implemented and finally used to create the products.

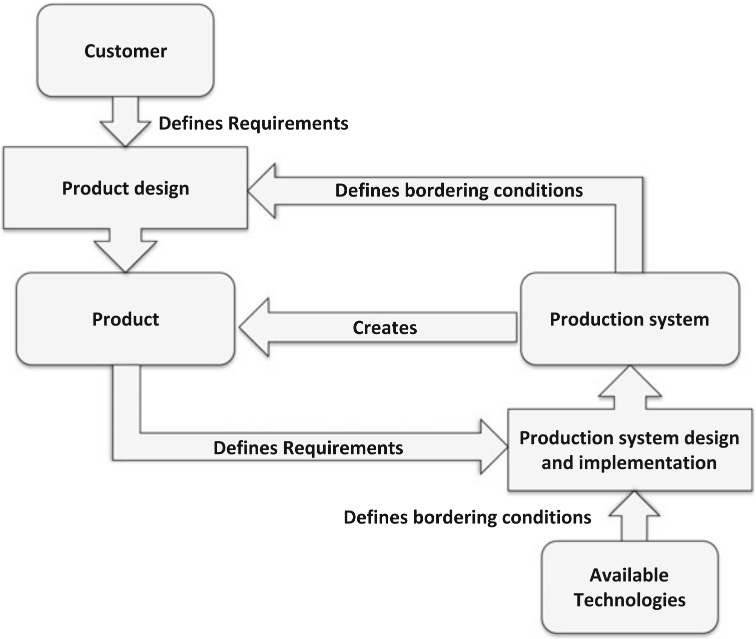

Fig. 2.1 Relations between product and the production system

The design and implementation of industrial production systems require the collaboration of several engineering disciplines. Most relevant among them are industrial plant planning, production process planning, and mechanical, electrical, and software engineering. Chapter 1 introduced a general notion of the basic concepts of *industrial production systems* and the need for *intelligent engineering applications* when moving from traditional to more flexible industrial production systems. This chapter builds on these general concepts to discuss semantic challenges and needs coming with the heterogeneous data models used in the multi-disciplinary engineering of industrial production systems.

The different engineering disciplines involved in the engineering of production systems apply engineering methodologies, tools, and data sets tailored to the needs of these engineering disciplines. These experts from different domains in production systems engineering are aware of the challenges coming from heterogeneous data models in the tool landscapes they use and want to better understand their options for a roadmap for technology adoption to effectively and efficiently move toward sufficient data integration. However, most production systems engineers are not experts in Semantic Web technologies and only few Semantic Web technology experts have a deep understanding of industrial production systems, their design, and evaluation. Therefore, this chapter intends to provide a foundation for the discussion between production systems engineering experts and Semantic Web technology experts on semantic challenges, needs, and options.

The remainder of this chapter is structured as follows. Section 2.2 introduces the production system life cycle and motivates the research question. Section 2.3 describes the engineering process of industrial production systems in more detail. This section describes the process structure, depicts the information usually involved in this process, and names the engineering disciplines involved. By that, this section highlights the multidisciplinary and multi-model character of production system engineering. In addition, this section names relevant key concepts of the multidisciplinary engineering for industrial production systems for nonexperts as a foundation for relating the usage scenarios and needs in the following sections. Section 2.4 introduces four usage scenarios that illustrate needs for semantic support both for nonexperts in engineering and nonexperts in Semantic Web technologies to provide common ground for discussing challenges and solution approaches. This section highlights especially the needs that Semantic Web technologies could address. Section 2.5 derives from the scenarios general and domain-specific needs for semantic support in more detail to enable the definition of requirements and needs for and the selection of suitable solution approaches. Finally, Sect. 2.6 summarizes the major results and provides an outlook on future research work.

2.2 Production Systems Life Cycle

Input to the production system engineering are the production steps and involved material required to produce the products intended and the technological and economical possibilities for production. Based on this input, in a first step the set of *production system resources* is selected. Here for example the type of welding robot required for the welding process of a special car is identified and named. To each production process step at least one (production system) resource is assigned. All selected resources will be put into a sequence to roughly define the production system. This assignment and sequencing is validated against economic conditions. If the resource assignment is successfully verified, each of the production resources is designed in detail in the next step of production system engineering. For the welding robot for example the welding gun is selected by the process engineer, and the mounting position is defined by the mechanical engineer, the energy supply is engineered by the electrical engineer, and the motion path is programmed by the robot programmer. The consistency of the overall engineering can be validated in the virtual commissioning using simulations. Once the detailed engineering is completed, the production system can be installed (i.e., set up physically) and commissioned (i.e., filled with control programs and started sequentially for the first time). If the commissioning was successful, the production system can be used for processing products. Over time, each running production system needs to be maintained to ensure a safe and long living operation. If the production system is not required anymore (for example the products cannot be sold anymore), it can be redesigned or decommissioned. Figure 2.2 shows this general process.

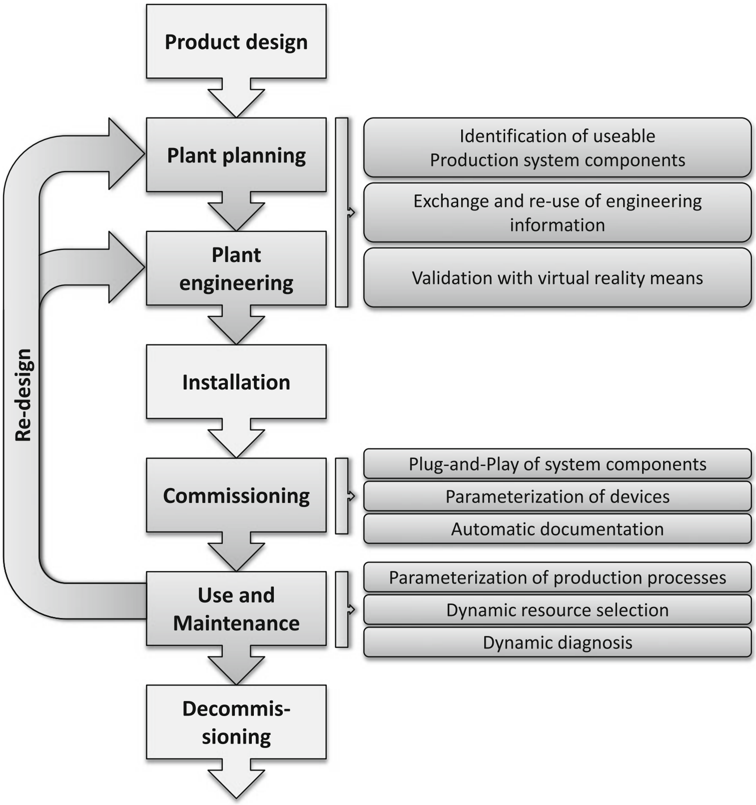

Fig. 2.2 Production system life cycle processes based on (Lüder 2014)

All technologies applicable in production systems can evolve. These developments enable new technical possibilities for the design and application of production systems. Envisioned possibilities shall include

- *plug-and-participate capabilities* of production resources (i.e., the integration and use of new or changed production resources during production system use without any changes within the rest of the production system),
- *self-* capabilities* of production resources (e.g., self-programming of production process control, self-maintenance in case of technical failures, or self-monitoring for quality monitoring), and
- *late freeze of product-related production system behavior* (i.e., fixing the characteristics of an ordered product at the latest possible point before production step execution, e.g., enabling to change the ordered color of a car until the start of painting) to name a few examples (Kagermann et al. 2013).

One step ahead of multidisciplinary engineering, information science and information technology have reached a point enabling a wide-ranging impact on production systems, especially on the behavior of value creation chains, their design, and changeability (Jacob 2012). In parallel, device-related capabilities have increased. Especially control devices as essential part of the production resources increasingly contain intelligence and are able to take over responsibilities within the control of production systems (Binder and Post 2012). Thus, unsurprisingly, advanced capabilities for information processing will find their way into production systems and realize the historic vision of Computer Integrated Manufacturing (CIM) in a new fashion (Kagermann et al. 2013).

Lüder identified a set of challenging situations in the life cycle of a production system (Lüder 2014), which can be tackled within the *Industrie 4.0* (BMBF 2007) approach (see Fig. 2.2 for details). Within the life cycle of production systems, several information processing systems are involved/applied. On the one hand, during the plant planning and plant engineering, engineering tools are applied to create models and other descriptions that represent the production system and its resources on different levels of detail. These models and descriptions as a whole will represent the specification on how to build and apply the production system. Usually, different models and descriptions are considered as engineering artifacts. On the other hand, during the installation, commissioning, and use of the production system, physical artifacts (i.e., physical system components of different complexity) are used and controlled. Thereby, control and behavior information emerge and are used which are named run-time artifacts. Usually, different engineering artifacts, physical artifacts, and run-time artifacts depend on each other or need to be at least consistent to each other. To ensure this consistency, integration capabilities on data processing level are required.

- *Horizontal Integration*. During the phase of using the production system (i.e., runtime or operation), the interaction of the different production system resources (possibly located at different geographical locations) and its control systems as well as its interaction with the production system environment (e.g., delivery of materials, energy consumption, waste disposal, or product delivery) need to be coordinated. This is considered as *horizontal integration* within a value chain network. Horizontal integration shall enable the automatic integration of new production resources within a production system, in the same way as today USB devices are integrated into a PC system by using plug-and-play mechanisms. It also shall enable the automation of routine tasks, such as process documentation or diagnosis of system components.
- *Vertical Integration*. During the development phase of the production system, starting with the plant planning until the use of the production system and its maintenance, it is of interest to coordinate the different steps of artifact creation and to ensure the availability of all necessary information/artifacts developed in previous engineering process steps. Therefore, an integration of engineering activities, engineering tools, and engineering disciplines is required, enabling the exchange and possibly reuse of developed information. This is named *vertical integration*. Vertical integration shall enable the automatic application

of the correct parameters out of the engineering information to configure a production resource or to dynamically select the right production resource for a special product order.

Figure 2.2 depicts an assignment of the challenges to life cycle phases. These challenging situations can only be tackled by enabling the adequate application of information and knowledge about the product, the product structure, and required production processes, the production system architecture and resources, their properties, and their possible production processes within the engineering and control of the production system. However, this adequate application is not possible yet. Most of the knowledge is only implicitly given within the engineering and run-time artifacts of a production system, and have to be modeled and made explicit for further use. Improved support is required for the modeling of the semantics of engineering artifacts as well as the subsequent usage of these semantics.

To deal with these challenges and to start a discussion between the two communities of production system engineering and information sciences related to these topics topic, this chapter intends to address research question (RQ) on the "**need for semantic support in multidisciplinary production system engineering**." Domain experts in production systems engineering typically are not also experts in Semantic Web technologies. However, these experts can provide usage scenarios in multidisciplinary engineering, which illustrate needs for semantic support, which pose challenges to the domain experts in their daily work. We aim at identifying recurring needs for semantic support in these scenarios to provide a foundation for communicating these challenges to Semantic Web technology experts, who typically are not also experts in multidisciplinary engineering.

To address this research question, we applied the following *methods for research*. We synthesized representative usage scenarios from literature study and own experience. Following the process of conceiving, instantiating, and changing a production system during engineering and operation phases, we identified recurring engineering activities that involve significant challenges regarding heterogeneous and/or implicit expert knowledge, which is necessary to conduct and possibly automate the engineering activity, as needed for CPPS engineering. We analyzed the needs for semantic support in these scenarios and identified common needs. The set of needs was validated and extended in discussions with domain experts, who adapted the generic scenarios to their own engineering contexts.

This chapter provides the following contributions for scientific communities and target audiences of the book. Engineers and managers from engineering domains will be able to get a better understanding of the benefits and limitations coming from using Semantic Web technologies. Engineers and managers from engineering domains can use the scenarios to select and adopt appropriate Semantic Web solutions in their own settings. Semantic Web researchers can use the scenarios and needs to get a better understanding of the challenges and requirements of the multidisciplinary engineering domain, which will support and guide the researchers in developing new technologies and solutions for this important application area more effectively and efficiently. Within this chapter, key challenges for semantic modeling exploited in production system engineering are identified. Therefore, selected important *Industrie 4.0* challenges are aggregated in four main scenarios of the application of ideas of *Industrie 4.0*: two scenarios related to engineering and two scenarios related to production system usage phases in the overall life cycle of a production system. These scenarios are recurring scenarios not only discussed within the collected requirements for *Industrie 4.0* (see Kagermann et al. 2013) but also addressed in several research activities.[1]

(1) The first scenario "*Discipline-crossing Engineering Tool Networks*" considers the capability to interact appropriately within an engineering network covering different engineering disciplines, engineers, and engineering tools. This scenario highlights the need for a common vocabulary and technical interfaces between engineering tools over all engineering disciplines involved in an engineering organization creating a production system to enable a fault-free information propagation and use.
(2) The second scenario "*Use of existing Artifacts for Plant Engineering*" has a focus on knowledge reuse and protection within engineering organizations. This scenario considers the problem of identification and preparation of reusable production system components within or at the end of an engineering project, and the selection of such components within engineering activities. Here, the focus is on the required evaluation of component models to decide about the potential usability of the component within a production system.
(3) The third scenario, "*Flexible Production System Organization*," discusses the problem of run-time flexibility of production systems. This scenario sketches requirements following the intention of integration of advanced knowledge about the production system and the product within the production system control at production system runtime.
(4) Finally, scenario four, "*Maintenance and Replacement Engineering*," combines engineering and run-time information of a production system toward improved maintenance capabilities of production system components.

These scenarios allow researchers and practitioners from the Semantic Web area to better understand the challenges of production system engineers and managers, to define goals and assess solution options using Semantic Web technologies. Major semantic challenges come from the need to provide tool support for processes that build on heterogeneous terms, concepts, and models used by the stakeholders in production system engineering and operation. From the challenges illustrated in the four scenarios, we derive needs for semantic support, including the support for multidisciplinary engineering process knowledge from the usage scenarios, as a foundation for evaluating solution approaches.

---

[1] AutomationML projects: https://www.automationml.org/o.red.c/projects.html.

2.3 Engineering of Industrial Production Systems

In Sect. 2.2, the life cycle of production systems and, especially, its engineering phase has been described on a high level. To enable the detailed evaluation of the multidisciplinary character of production system engineering, we will describe in this section the engineering process of production systems in more detail. Following the Engineers Council for Professional Development (Science 1941), an engineering process is defined as a sequence of activities that creatively apply scientific principles to design or develop structures, machines, apparats, or manufacturing processes; all with respect to reach an intended function for economic and safe operation. Thus, an engineering process is based on a sequence of design decisions to be taken. Engineering processes are executed by engineering organizations (EOs). Following the VDI regulations (VDI 2010), an EO can be an engineering company, an engineering subunit of a company, or a combination of different companies and/or subunits executing one or more engineering processes for a limited time and providing/containing the necessary resources for process execution. Thus, each EO will execute at least one engineering process described by a procedure model. The procedure model determines the sequence of engineering activities, covering design decisions with required input and output information, tools to be used, and responsible roles. The procedure model formalizes the engineering process and reflects the technical content of an engineering process in an organization.

Figure 2.3 shows the structure of an engineering activity, which can be also seen as a design decision or a set of design decisions. Within an engineering process for each engineering activity, the following requirements should be met: to enable the execution of the engineering activity, (1) a set of predecessor engineering activities has to be finished, (2) specific information has to be available, and (3) appropriate engineering tools should be usable. Humans with the required skills and knowledge should execute the engineering activities. The engineering activity will create information reflecting the taken design decisions and provide them for successive design decisions.

In addition, engineering activities establish a network of activities. As described by Lindemann (Lindemann 2007), the set of engineering activities containing the design decision execution has a hierarchical structure (see Fig. 2.4). On the highest level of the engineering process, design decisions can be considered as engineering process phases. These phases contain engineering activities, like control programming, which can be separated into sub-activities, like hardware configuration and code design. Again, these sub-activities can be divided into sub-sub-activities, like I/O configuration. Finally, the hierarchical decomposition of engineering activities ends with elementary actions, e.g., naming input and output variables of controls, defining the wiring path between a control and a sensor, or specifying the rotation speed of a drive for a special situation.

This hierarchical structure of engineering processes is reflected in nearly all engineering process models in literature (Lüder et al. 2011). Most engineering

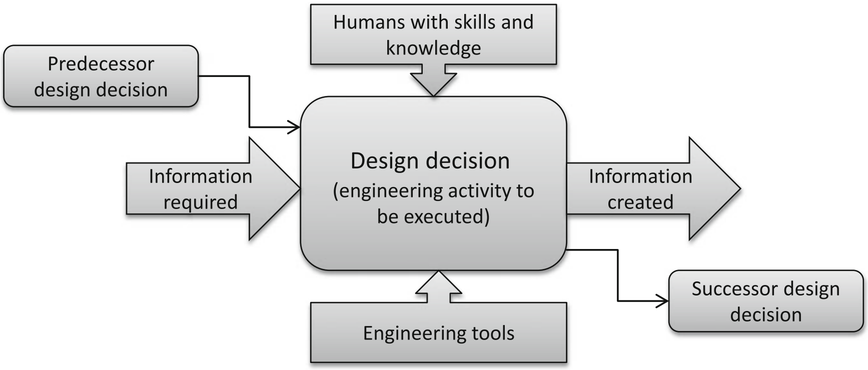

Fig. 2.3 Impact on/of design decisions within an engineering activity based on (Schäffler et al. 2013)

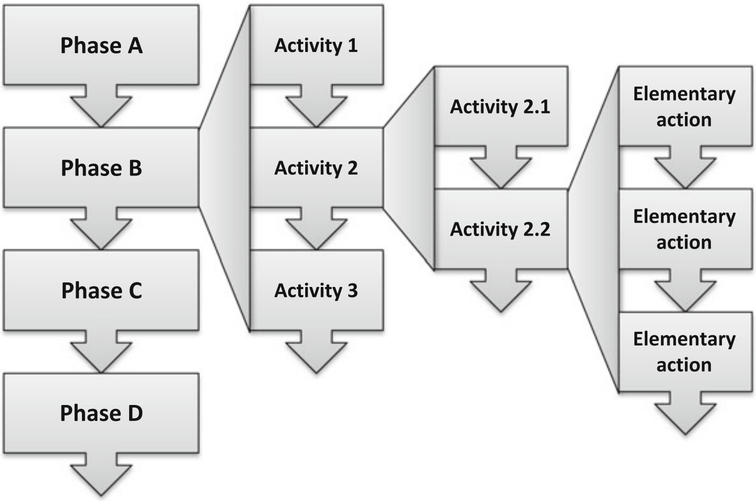

Fig. 2.4 Engineering activity hierarchy based on (Lindemann 2007)

processes have been developed for different purposes in the area of production systems engineering, resulting in considerable process variety. However, the described engineering processes can be distinguished with respect to the type of business model used, the industrial domain addressed, the completeness with respect to the life cycle of products and production systems, and the involved engineering disciplines and stakeholders. In the following, engineering processes will be distinguished with respect to the business model they follow. Of course, the set of processes considered in this chapter is not complete. From a business point of view, there are main business models as follows: *solution providers*, *component providers*, and *product developers*.

- The *solution provider business* addresses the engineering of a production system, e.g., a car manufacturing plant or a steel mill, as a unique system developed to produce specific products in a special way. Usually, solution providers cover plant planning, plant engineering, implementation, commissioning (i.e., acceptance test), and use of production systems. Thereby, these processes are project oriented and in some industrial domains—they are called "*project business*." Examples of such engineering processes are the engineering processes specified in the VDI 5200 Guideline "Factory planning" (VDI 2009a), the AutomationML reference engineering process (Drath et al. 2008), the engineering process proposed by Schnieder (1999), and the engineering process presented by Kiefer (2007).
- The *component provider business* addresses the development of reusable production system components, e.g., a welding robot or a type of steel press, which are typically intended for one industrial domain and a special set of production process steps they are applied within. These components are mainly developed based on best practices in solution business projects, market analysis, and technological possibilities. Therefore, the engineering processes typically contain the component planning, component engineering, implementation, validation, and the clear documentation of the developed engineering artifacts. Examples of such engineering processes are the AQUIMO engineering process (Aquimo 2010) and the Domain Engineering process (Maga et al. 2010).
- The *product development business* addresses the development of products usable within several industrial domains, e.g., a robot, an electrical drive or a sensor. Here it needs to be kept in mind that products in this business model are not consumer products but objects applicable within different technical settings by adapting them to the application case. The results of the engineering processes of this business type are intended to be multipurpose devices and construction elements. Facing the very different types of product-related engineering processes like (Ulrich and Eppinger 2007; Schäppi et al. 2005; VDI 2004), the product business mostly follows an engineering process covering the phases requirement analysis, product planning, product engineering, implementation, and validation.

There are some newer approaches considering different businesses in parallel or its combination. Here, two representative engineering processes are the engineering process of VDI Guideline 3695—"Plant engineering" (VDI 2010)—and the engineering process of the VDI Guideline 4499—"Digital factory" (VDI 2008, 2009b). The different engineering processes described above provide the possibility to be combined into one generalized engineering process. Figure 2.5 shows the generalized engineering process, which combines the product, component, and solution provider businesses using the knowledge about production system components, their functionalities, and use. The generalized engineering process consists of three subprocesses: (a) *product development business* for the design of products usable as mechatronic units within production systems and its components; (b) *component provider business* for the design of components usable as mechatronic units within

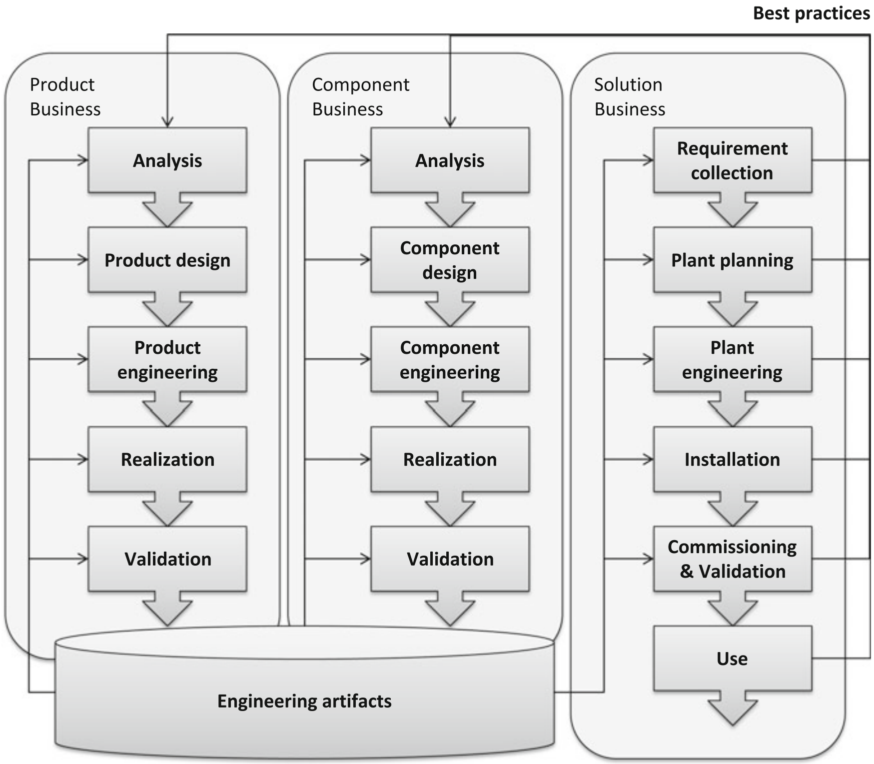

Fig. 2.5 Generalized engineering process based on (Lüder et al. 2011)

production systems; and (c) *solution provider business* for the production system design and implementation exploiting predefined mechatronic units. These businesses define subprocesses with their phases that cover all businesses needed to engineer a production system. Depending on the developed system, each process/phase can proceed with more or less effort.

Within the five phases of the subprocess for *product development business* the best practice of the production system design, implementation and use, the knowledge from components design, and the market and technology conditions are exploited as starting points and basic requirements. The results of the subprocess are mechatronic units and their information sets containing all engineering artifacts developed within the complete process and describing all relevant engineering knowledge (Lüder et al. 2010, Hundt et al. 2010) which can be used in various application cases not limited by industrial domains.

Similar to the subprocess for product development business, the subprocess for *component provider business* with its five phases starts also with an analysis of the best practice and requirements of the production system design, implementation, and use as well as the available mechatronic units, their provided functionalities,

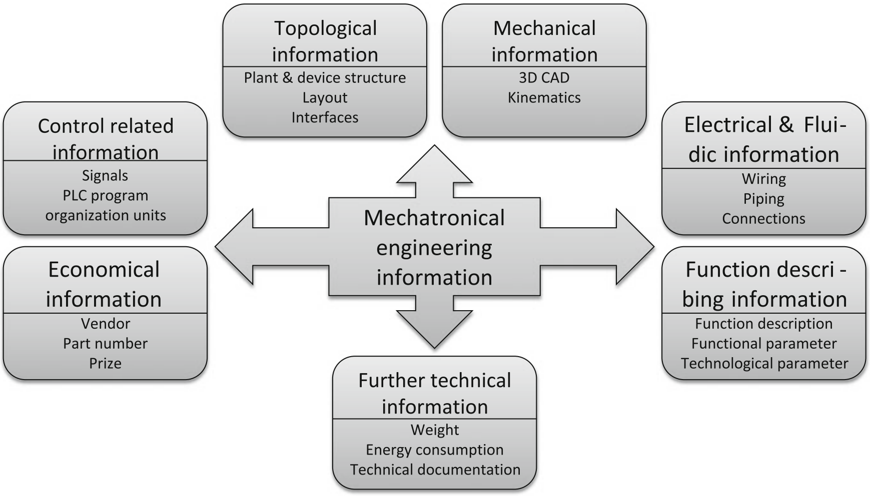

Fig. 2.6 Mechatronic engineering information sets based on (Lüder et al. 2010)

fulfilled requirements, and available engineering knowledge. This subprocess results in the provisioning of mechatronic units and their describing information sets with a higher complexity compared to the product development business. The mechatronic units are developed to be used in application cases limited to one industrial domain. Thus, the component- and product-related subprocesses will both feed a library of mechatronic units usable within components and production system-related subprocesses (see Fig. 2.6).

The subprocess for production system design and implementation (i.e., *solution provider business*) with its six phases exploits the provided set of mechatronic units as reusable components set to set up the production system resources. This subprocess starts with the requirement collection phase to collect all product-related and further requirements of the intended production system, e.g., for manufacturing a specific type of car. Based on these requirements, the production system is planned, engineered, implemented, commissioned, and used as described above. Beneath the running production system, the production system-related subprocess will result in best practices for the different industrial domains usable within the product and component design processes.

Note that the described process structure is idealized. Depending on the EO executing the engineering process and the industrial domain addressed, there is a wide variety of realizations and particularizations of this process. Typical production systems developed by the named processes are production systems like car body welding shops, combustion-turbine-based power plants or process plants for polyvinyl chloride generation. Within these production systems, components and devices such as welding robots, conveyor systems, gas turbines, pipes, vessels, PLCs, inductive sensors, and level indication switches, and drives are combined. Looking at these engineering processes, there are different engineering information sets to be created and used. Typically, the following are the kinds of engineering information sets (see also Fig. 2.6):

- The *plant topology* as a hierarchy of production resources and devices involved within these production resources;
- *Mechanical construction* representing geometry and kinematics of production resources and devices including their properties;
- *Electrical and communication wiring* of electrical and communication construction including all wire- and device-related properties;
- *Behavior information* related to expected processes within production resources as controlled and uncontrolled behavior;
- *PLC code* of any kind to control devices and production processes; and
- *Generic data* summarizing further organizational, technical, economic, and other kinds of data, such as order numbers.

The information represents also the involved engineering disciplines, e.g., plant planning, process planning, mechanical engineering, fluidic engineering, electrical engineering, communication system engineering, control engineering, simulation, and procurement. Depending on the application case, also other special engineering disciplines or natural sciences can be involved, such as chemistry in process plants or nuclear physics in nuclear power plants.

Figure 2.7 shows typical stakeholders involved within the engineering of a production system:

At first, there is the *plant owner* (1). He is responsible for the economic success of the production system and, therefore, defines the products to be produced, the capabilities of the production system to produce the products, and bordering conditions to the production system. Within this task, he exploits information gained by the product *customers* related to the intended product portfolio and certain

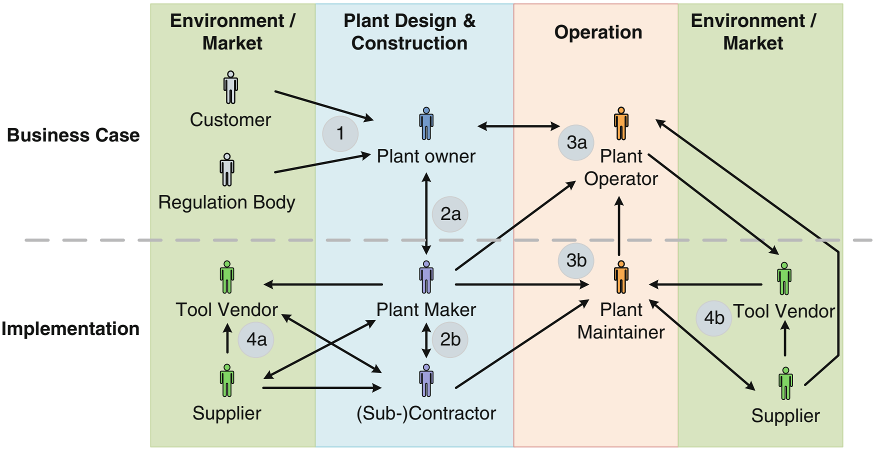

Fig. 2.7 Stakeholders in added-value chains related to industrial plant engineering

regulation bodies defining bordering conditions to the production system related to system safety or environmental protection.

The *plant owner* will contract a *plant maker* (or *system integrator*) to create the production system following the defined intentions and bordering conditions given in a technical and functional requirement specification (2a). The *plant maker* will take up these basic documents and will ensure the engineering, installation, and commissioning of the production system. Within the necessary activities, he is assisted by *subcontractors* (2b) taking over some of the engineering, installation, and commissioning activities and by *component and device suppliers* (following the device and component provider business) providing information for the engineering and physical objects for installation.

After the production system has been commissioned and handed over to the *plant owner*, the *plant owner* will authorize a *plant operator* to run the production system and to produce the intended products (3a). As production systems, like all technical systems, will degrade over use time, the *plant operator* gets technical support from *plant maintainers* (3b) who are responsible to ensure the availability of the production system resources and devices.

The *tool vendor* has essential impact on engineering (4a), use, and maintenance (3b) activities related to a production system. He provides engineers with software tools that are needed to create engineering artifacts, access plant components at commissioning and runtime, and enable device diagnosis. The capabilities and quality of the software tools have essential impact on the efficiency and quality of the activities they are applied within. Therefore, the software tool users (*plant maker*, *plant operator*, *plant maintainer*, *supplier*, and *subcontractor*) will provide the *tool vendor* with requirements toward tool capabilities resulting from their special application conditions. The described network of interactions between the different stakeholders is depicted in Fig. 2.7, which represents also the flow of information between the different stakeholders.

The discussion of the complete flow of information goes far beyond the scope of this chapter. Therefore, we will focus on discussing selected illustrative examples: *Customer* and *regulation bodies* will provide requirements for the product and the production system to the *plant owner*. The *customers* will define the product characteristics (e.g., product type and quality, amount) resulting in the definition of the production process to be executed on the production system. The regulation bodies will define safety-related product requirements (like CE conformity) and production system-related safety and workforce requirements (like protection guidelines from the employers' mutual insurance associations). All these kinds of information are related to function-related information or further technical information.

- *Plant owner* and *plant maker* will exchange both requirements to the production system and production system realization information. Usually, the *plant owner* will provide a requirements specification including functional and nonfunctional requirements within a tender document (German: *Lastenheft*) and the *plant maker* will reply with a technical specification (German: *Pflichtenheft*). Finally, the *plant maker* will provide a complete plant engineering documentation. Thus, the communication between *plant owner* and *plant maker* will cover all types of information shown in Fig. 2.7.

- *Plant maker* and *subcontractors* will exchange the same type of information like *plant owner* and *plant maker* but on a more detailed level covering only the parts of the technical system the *subcontractor* should contribute to.
- *Suppliers* will provide the *plant maker* and *(sub-)contractors* with information about their components or devices. Usually, *suppliers* will provide technical documentation covering user manuals, installation guidelines, and basic engineering information covering all information sets of Fig. 2.7.
- *Tool vendors* will get from all involved stakeholders (except for *customer* and *regulation bodies*), the information/requirements on engineering and data exchange capabilities of the tools to be used in engineering. Here, the most interesting information are the types of artifacts to be created in the engineering process (types of diagrams, such as P&ID or wiring diagrams, types of lists, such as device lists or terminal stripe assignments) and the capabilities for efficient engineering.
- *Plant operator* and *plant maintainer* will get the "plant as is" documentation from the *plant maker*. Also, here the exchanged information will cover all information sets of Fig. 2.7. This information is required to appropriately understand, run, and maintain the plant.

Note that the share of effort contributed by the different stakeholders may vary based on company size, industrial setting, and application case. For example, the engineering effort share of the *plant owner* and *plant maker* differs according to the company size. Larger *plant owners* within the automotive industry, e.g., Volkswagen or Daimler, will usually execute the basic engineering of the production system until the definition of the production resources to be applied and afterwards hand over to the *plant maker*. In contrast, small *plant owners* in consumer goods industries, such as a furniture manufacturer, only define the product and the *plant maker* will make the complete engineering. Similar sharing patterns can be found in maintenance activities, where, e.g., in the process industry special device *suppliers* and/or *subcontractors* will take over responsibilities for maintenance activities, while in automotive industry the *plant owner* will take the complete burden of maintenance.

2.4 Usage Scenarios that Illustrate Needs for Semantic Support

After having detailed the engineering process of production systems and illustrated its multidisciplinary and multi-model character, this section will introduce relevant scenarios of information application and information creation within production system engineering. Thereby, this section will provide a deeper view on needs for

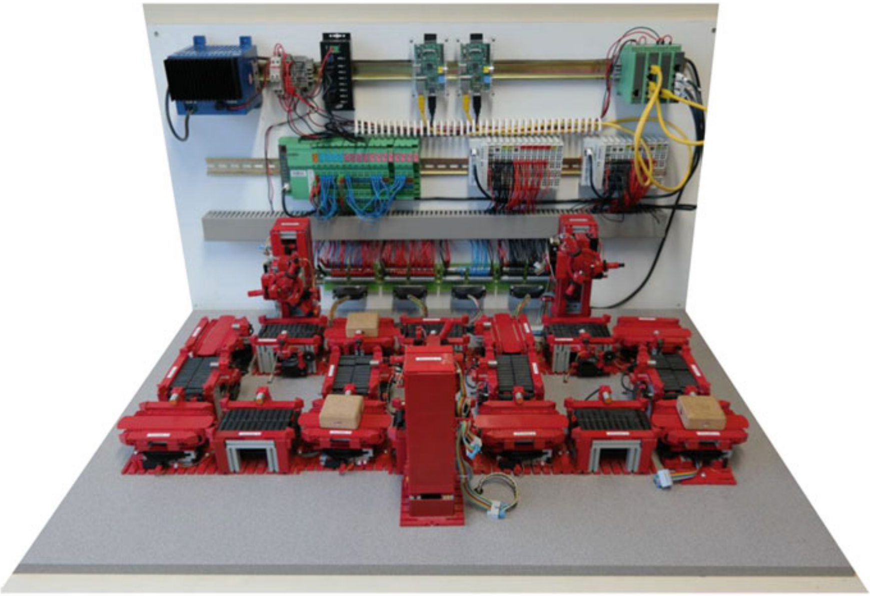

Fig. 2.8 Lab size production system as running example

semantic integration during typical steps for the creation and application of data and information within the life cycle of production systems. These scenarios will cover different life cycle phases of the production system to represent the typical engineering knowledge and integration needs introduced in Sect. 2.2. For illustration, the scenarios will be discussed in the context of a lab-size production system that is hosted at Otto-v.-Guericke University Magdeburg.[2] The production system consists of a set of three multipurpose machines, eight turntables, and ten conveyors, and is wired with Field-IOs to Raspberry-Pi-based controllers as depicted in Fig. 2.8.

Each of the machines, turntables, and conveyors is equipped with the same types of sensors and actuators. There are inductive proximity switches to detect materials, position indicator switches for position sensing of the turntables and machine heads, and drives to move conveyor belts, rotate turntables, rotate tools, and move the machine heads. All control devices are connected to three modular Field-IOs by wire. Two Ethernet switches, a controller, and the Field-IOs establish a Modbus TCP network to enable access to the physical inputs and outputs. Finally, the Raspberry-Pi-based controller runs a PLC program that controls all devices.

---

[2] Institute of Ergonomics, Manufacturing Systems and Automation at Otto-v.-Guericke, University Magdeburg: Equipment Center for Distributed Systems, http://www.iaf-bg.ovgu.de/technische_ausstattung_cvs.html, last access March 2016.

2.4.1 Scenario 1—Discipline-Crossing Engineering Tool Networks

As mentioned in Sect. 2.2 the engineering of production systems is multidisciplinary engineering involving different engineering activities executed by engineers coming from different engineering disciplines with different engineering tools, where the engineering activities establish a network of dependencies. Thus, efficient engineering depends on efficient networking. For example, the *Industrie 4.0* value chain processes “Plant and process planning” and “Production system engineering” can be improved significantly with consistent engineering data exchange between the involved engineering activities (and therefore, the involved tools and engineers) This is also visualized in Fig. 2.5 as the exchange of engineering information through the mechatronic component library that contains engineering artifacts.

At the beginning of the life cycle of a production system, the plant owner will contract the plant maker to create a production system following a set of defined requirements. The plant maker will start an engineering process, as described in Sect. 2.2, involving subcontractors for special engineering purposes. The main aim of the plant maker (and also of the involved subcontractors) is the efficient creation of all required engineering artifacts (e.g., plans and programs) required to install and commission the production system with sufficient quality, following the aims of the plant owner and the requirements of subsequent phases of the production system life cycle.

As described in Sect. 2.2, the production system engineering process involves the execution of engineering activities in different disciplines, like mechanical, electrical, and control/software engineering. Each discipline has to specify discipline-specific parts of the production system. Looking at the example plant during the engineering, the different required plant parts have to be selected and instantiated resulting in a list of system components to be used. Within mechanical engineering, these system components are detailed, leading to the identification of devices (e.g., sensors, drives, and controllers) and other buyable parts and their location and application in the production system (and the production process). Within electrical engineering, the used devices are integrated into wiring networks. In control engineering, the software programs for the controllers are implemented, driving the devices to achieve the required system behavior. Therefore, each of the involved disciplines and engineers may deal with the same objects (e.g., plant sensors, drives, and controllers) from different viewpoints. To enable the mutual understanding of the different engineers, they require across all disciplines a common set of concepts (e.g., signals), which have a discipline-related incarnation (e.g., the position of a plug in mechanical engineering, a wiring point in electrical engineering, and a program variable in control engineering).

This set of *common concepts* is currently not explicitly represented in discipline-specific and isolated tools. The main reason for this problem is the existence and development of discipline-specific engineering tools by tool vendors. Usually, tool vendors focus their business on one engineering discipline or on related engineering disciplines. Thus, during the last decades engineering tools have been developed and highly optimized for the execution of discipline-specific engineering activities, like MCAD tools, ECAD tools, and control programming tools. These tools apply engineering-discipline-specific concepts and languages that evolved in parallel to the engineering tools following the understanding and acting in the engineering discipline. However, the focus on one engineering discipline led to neglecting the consideration of other engineering disciplines involved in the overall engineering process and of higher level participants, such as project managers or system integrators.

Within recent years, the need for better integration of engineering tools along the complete engineering tool chain (or tool network) has been identified and has received prominent attention as foundational requirement for the *Industrie 4.0* approach (Wegener 2015). Thus, there is a need for the identification of common concepts between and within the different engineering disciplines and to represent and integrate these common concepts to enable lossless data and information exchange along the engineering tool chains/networks. The identification of common concepts has to be supported by tool users (especially the plant maker) and tool vendors in cooperation.

Exploiting the developed common concepts within the network of engineering activities, two main benefits can be reached. On the one hand, the involved engineers can improve the quality of the developed system architecture by mutual discussions and understanding needs of the different involved engineering disciplines. Thereby, the overall quality of the developed production system can be improved. On the other hand, the information flow within the network can be improved. The common concepts will improve the capabilities of implementation and application of appropriate tool interfaces enabling lossless data exchange and a fault-free data application in the data-receiving tool.

Security is an issue in this scenario. The information exchange can be executed between different involved legal parties providing engineers to the engineering process. In Fig. 2.7, it is easy to see that on the one hand the involved stakeholders are interested in protecting the information exchange against access from outside of the engineering network. On the other hand, they might be interested in securing information against access of different partners within the engineering network to protect vital knowledge. For example, one subcontractor may require knowledge protection with respect to other subcontractors of a supplier to ensure that component-specific models can only be applied by special subcontractors (e.g., in case of virtual commissioning or simulation).

To ensure the necessary quality assurance and the high degree of parallel work in engineering, several needs for semantic capabilities arise. The knowledge on engineering artifacts needs to be explicitly represented (N1) to make them accessible and analyzable; the views of several engineering disciplines need to be integrated (N2); the quality assurance, e.g., consistency checks of engineering model views across disciplines, and automation of engineering, e.g., the generation of derived artifacts, require functions for engineering knowledge access and analytics (N3). In addition to structured data stored in databases or knowledge bases, the use of semi-structured data, e.g., technical fact sheets that include natural language text, or linked data, e.g., component information from a provider, in the organization, and on the Web can help improve the automated support for reuse processes (N4). These required capabilities provide the foundation to improve the support for multidisciplinary engineering process knowledge like supporting the exchange of advice to amendments within the different created artifacts based on identified dependencies (N6).

2.4.2 Scenario 2—Use of Existing Artifacts for Plant Engineering

As mentioned above, the frequency of activities in the life cycle of production systems is increased significantly (especially by reducing the time span of use phase) following the reduction of the life cycle of products. *Industrie 4.0* faces this trend by enforcing the integration of the different value chains of production systems, its components, and its devices (as discussed in Sect. 2.3). An example of this integration is given in the VDI/VDE guideline (VDI/VDE 2014). It considers the integration of manufacturers of PLCs, screws, metal sheets, and washing machines (see Fig. 2.9). Here, it gets visible that components used in technical systems

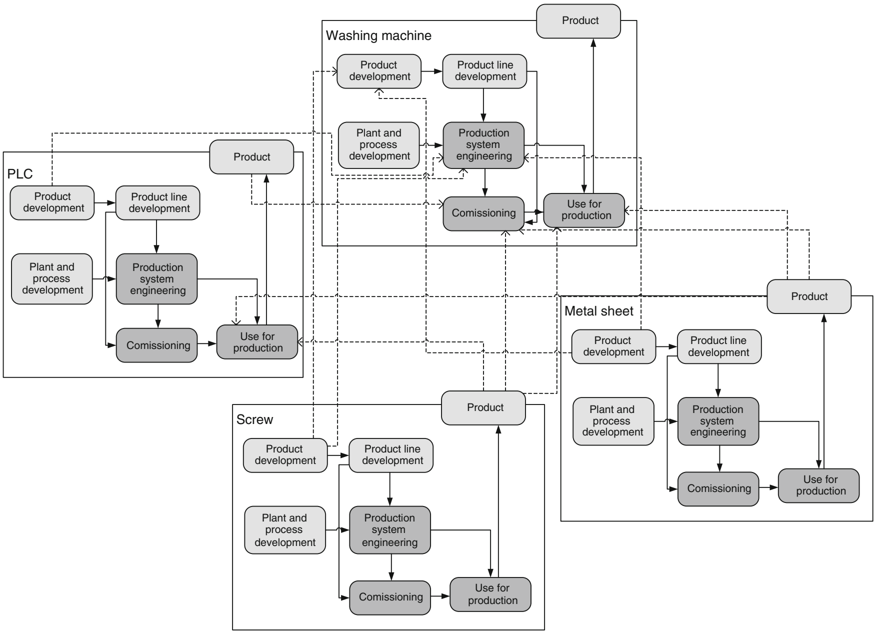

Fig. 2.9 Example for integration of value chains based on (VDI/VDE 2014)

(production system and products) can be reused. Thus, also their "digital shadows" developed/exploited in engineering, i.e., the data shown in Fig. 2.6, can be reused. As an example, the *Industrie 4.0* value chain processes "Plant and process planning" and "Production system engineering" can be improved significantly with better automated support for the reuse of existing artifacts during plant engineering. This automation of support for reuse is currently hampered by a weak integration of the diverse knowledge needed from several roles in the engineering process for matchmaking between: (1) reuse requirements for production systems and (2) the capabilities of reusable devices and components.

The plant maker is strongly interested in a most efficient engineering process as the engineering process efficiency has a major impact on the economic effect of the engineering project. Therefore, the plant maker intends to reuse existing engineering artifacts coming from various sources. These sources can be the plant maker (and his subcontractors) exploiting libraries of engineering artifacts developed in prior projects (reuse of engineering artifacts), or suppliers of devices or components providing libraries with detailed engineering information for these devices and components. To make this reuse possible, the engineering artifacts have to be developed and prepared for integration into a library, need to be identified as appropriate for an application case, and be adequately incorporated in the artifact set to be established by the plant maker. This reuse can be prepared for each discipline separately. However, reuse preparation that covers several disciplines based on a leading discipline (VDI 2010) is probably more efficient. An example for such a reusable artifact is a conveyor belt. It contains a set of automation devices like a drive chain and material sensors. For each conveyor belt, the mechanical engineering defines mechanical parts, the involved automation devices, and their assembly; the electrical engineering specifies the wiring of the devices including active and passive infrastructure; and the control engineering develops appropriate control functions for the belt motions. All these contributions can be developed once and reused in each project requiring a conveyor belt with these specified characteristics.

The preparation of artifacts for reuse in other engineering projects assumes the existence of an overall system architecture (within one discipline or across disciplines) with types of applicable system objects (the example in Fig. 2.9 includes machines, turntables, conveyors with drives and sensors, which are interconnected by wires), and system dependencies and interrelations (in the example, wiring relations, neighborhood relations, or control dependencies). The definition of system-object types requires well-defined semantics for production system components in the production system hierarchy, see, e.g., in Kiefer (2007; Hundt et al. 2010). The definition of system dependencies and interrelations requires a well-defined semantics for dependencies between components (within and across engineering disciplines), and a unique mapping of discipline-dependent artifacts to components. For example, the drive controller within the drive chain of the conveyor belt is considered in mechanical, electrical, and control engineering. Within electrical engineering, the set of control inputs and outputs is connected to wires enabling the information exchange between the drive controller and the overall controller for the conveyor belt. Within the control programming, there is a control variable assigned to each input and output of the drive controller. Consistency requires to ensure that for each connected input and output there is a variable with an appropriate type. The definition of these well-defined semantics is an activity of the plant maker in cooperation with his subcontractors and suppliers. Therein, the plant maker has to consider his specific production system architecture, while suppliers have to ensure the possibility to apply their components/devices within different plant architectures and engineering processes. For example, the plant maker will focus on the requirements of the specific product like a car body, while the provider of conveyors as components or drives as devices will focus on the applicability of these conveyors and drives within application cases going beyond welding shops including for example logistic systems for conveyors and elevators for drives.

If libraries of artifacts exist, the plant maker has to be able to identify the appropriate library entry (i.e., component or device) for his application case (i.e., the specific production system) within the library. This means, that the plant maker (especially the plant planner, the mechanical engineer, the electrical engineer, and the control engineer) will select appropriate production system components and devices (in the example, machines, turntables, and conveyors, drives, sensors, communication switches, and wire types) to execute the necessary production processes for the intended products. All engineering roles have to select the relevant components/devices based on requirements coming from the customer regarding the product, the production system owner regarding business performance capabilities, the engineers of the other engineering disciplines involved regarding production system interfaces. To enable the selection of appropriate components and devices, three elements are needed: suitable descriptions of the requirements to be fulfilled by the component, suitable descriptions of components and their capabilities, and a methodology to map the requirements to the reusable components. Usually, the requirements will address the required functionalities of the component/device (e.g., drilling function of a machine, material identification function of a sensor) in combination with the application conditions (e.g., drilling depth needs, material types that shall be identified, environmental conditions of sensor use).

Currently, descriptions of components and devices are very heterogeneous. For devices, first classification standards, like eCl@ss[3], exist to provide device type classifications following the functional requirements (like servo-stepper motor or a pointed profile mill) and device properties that enable the evaluation of the device application range (e.g., maximal torque of a motor or the maximum cutting depth and the cutting diameter of the profile mill). Even through the existence of first component models towards classification standards, there is no commonly accepted classification of components available. Examples of available models are eCl@ss or vendor-dependent classifications, such as the drive chain classification of Lenze

---

[3] eCl@ss: www.eclass.de.

(Götz 2013). Such a classification shall focus on the executed production process (or its parts) provided by the component following, for example, the DIN 8580 standard series. As the requirements to components and devices may be very different (coming from different sources and describing very different subjects), a discipline-crossing general representation of required and provided component capabilities, which can be compared, shall be developed based on a well-defined semantics of objects. In this task, the plant owner (knowing best the required production processes), the plant maker (who has to make the mapping), and suppliers (who know the component/device capabilities) have to be involved to bring in existing knowledge of their disciplines, possibly defined in standards for aspects of the application domain.

Exploiting the developed discipline-crossing general representation of required and provided component capabilities within the network of engineering activities again two main benefits can be reached. By exploiting completely engineered, tested, and validated library elements, the involved engineers can improve the quality of the developed system by preventing engineering errors and exploiting the integrated components/devices in the most appropriate way. In parallel, engineering efficiency can be increased by reusing developed structures, plans, and other engineering artifacts, so these artifacts do not need to be "redrawn" over and over for each new project.

Security issues in this scenario have a similar level compared to Scenario 1. The developed libraries contain sensitive knowledge of the device and component vendors. They contain device- and component-related information which shall only be accessible to a limited set of roles. Hence, access rights and encryption are part of the libraries and the component/device models. Beyond this knowledge-related issue, there is also a market-competition-related issue. Each vendor has a vital interest in making his products more attractive for users than the products of the competitors. Therefore, the vendor may make his product easier to find and use or make other vendors' components less likely to score well with requirements. These comparisons can be supported by a well-defined model semantics.

Against the background of the required quality of engineering artifacts and the assurance and high degree of parallel work in engineering, several needs for semantic capabilities can be identified. The knowledge on system requirements, system architecture, and reusable components/devices need to be explicitly represented (N1), the heterogeneous views of system procurers and several engineering disciplines need to be integrated, in particular, there has to be a common view on the system architecture, e.g., a best-practice architecture in an application area, such as for printing machines or welding cells (N2), the mapping of system requirements to component/device capabilities requires functions for engineering knowledge access and analytics, e.g., the capability to explore the system architecture model at design time (N3). In addition to structured data in databases or knowledge bases, the use of semi-structured data, e.g., technical fact sheets that include natural language text, or linked data, e.g., component information from a provider, in the organization and on the Web can help to improve the automated support for reuse processes (N4). Based on these required capabilities, software systems to support reuse in an engineering environment can be designed (N5) to improve the support for multidisciplinary engineering process knowledge (N6).

2.4.3 Scenario 3—Flexible Production System Organization

Within Scenario 2, the integration of value chains has been considered related to engineering activities. Looking at Fig. 2.9, integration can also be related to the *Industrie 4.0* value chain processes "Commissioning" and "Use for production." These processes require information from the engineering phases of the life cycle for the production system in scope, but also from the "Use for production" phase of the life cycles of relevant system components and devices used in the production system in scope. The value chain processes "Commissioning" and "Use for production" can be improved significantly with better automated support for flexible production system organization. This automation of support is currently hampered by a weak integration of the diverse knowledge coming from several roles in the engineering process with the flexible organization of a production system at runtime.

The plant owner and, forced by the plant owner, the plant operator are interested in a most flexible application of the production system. There are various types of relevant flexibility regarding produced products, production processes, and production resources to automate production processes (Terkaj et al. 2009). Most often, the plant owner wants more production-resource-related flexibility (i.e., the ability to integrate and change production system components, such as adding another multipurpose machine and a new cycle of turntables and conveyors in the example plant. These changes aim at increasing production resource availability. A plant owner may want to change a machine head to enable other milling or turning processes without negatively impacting the overall production system functionality). A plant owner may aim at increasing product-portfolio-related flexibility (i.e., the ability to change the product portfolio, like adding a new product type without negatively impacting the overall production system functionality).

To achieve sufficient resource-related flexibility, *Industrie 4.0* envisions the dynamic integration or change of production system components within the production system at runtime (Kagermann et al. 2013). Integration or change is envisioned for small components, such as position indicator switches or drives as given in the running example, for larger components, such as complete machines or turntables, and for parts of production systems. To make this vision possible, two information processing features are required. First, the newly integrated or changed production system component has to provide information about (a) its capabilities, (b) its access paths, and (c) its control-related features. In the context of the example production system, an example sensor component shall provide information about (a) its possible measurement capabilities, (b) the fieldbus protocol, addresses and services to access the sensor values, and (c) the control-application code fragments, e.g., function blocks, to be applied in the control system for using the sensor component. Second, the overall control system of the production system must be able to reason about the information provided by the component and to integrate the component at runtime within the processes (especially production processes) to be executed. Concretely, the component capabilities need to be compared with the required capabilities for the production processes (e.g., a product instance shall be identified with inductive sensing capabilities) and, if appropriate, shall be utilized in the overall control architecture for the production control based on the provided access paths and control code fragments.

To enable product-related flexibility, *Industrie 4.0* assumes that each product shall know in a certain way the required material and processes for its creation (Kagermann et al. 2013). In the context of the example production system, the product shall know what types of machining steps are required for its manufacturing and the required machining parameters (e.g., tool speed, chip size, cooling). This information has to be used on the one hand for the identification of the applicable manufacturing resources and on the other hand to parameterize the processes executed on the resources by applying the related values within the control system. To enable this flexible approach, production processes need to be modeled and automated reasoning about required and provided capabilities has to be supported. As production system resources shall be applicable within various industries (e.g., the example production system could be applied for metal-part processing or wooden-part processing), the process description and reasoning shall be independent of the application industry.

For both types of required flexibility, a common process description based on the product–process–resource concept (Pfrommer et al. 2013) is desirable. This description requires expressing in semantically well-defined and comparable ways, the concepts and relationships of (a) the needs for capabilities of production system components and devices; (b) the component/device use conditions (access path and control); and (c) product-related processing requirements.

Again, this is a cooperative task for the plant owner (knowing best the product and its required production processes), the plant operator (running the control system), the plant maker (knowing the overall system architecture including the control architecture), and suppliers (who know the component/device capabilities). Exploiting the common process description, the plant owner and the plant maker can reduce the effort for adapting the production system to the changing needs and can increase in parallel the reachable quality of the changes toward the changed requirements. As an example, a metal part of a car needs to be drilled in a certain way before welding. A model of this drilling process and its dependencies to the product can be exploited to identify automatically the right drilling machine for this step by comparing it with the capabilities of the set of production resources. This will lead to faster adaptations and a better ROI.

In this scenario, security plays a completely different role than in Scenarios 1 and 2. Here, the intention is to exploit the modeled information automatically within the production system. Thus, beyond the security issue of preventing unauthorized access to the information or parts of it, safety issues of the production system have to be reflected. The provided information shall not harm the production system, staff, products, or the environment by resulting in malfunctions. Thus, beyond the encryption discussion of Scenarios 1 and 2, here the common process description needs to be validated with respect to the resulting behavior of the created system.

To enable the necessary definition of system requirements for components/devices, the matching to component/device capabilities, and the proper use of components/devices at runtime require several needs for semantic capabilities. The knowledge on system requirements, system architecture, and component/device capabilities needs to be explicitly represented (N1), the heterogeneous views of system requirements and component/device capabilities need to be integrated (N2), the matching of system requirements to component/device capabilities requires functions for engineering knowledge access and analytics, e.g., the capability to explore a system architecture model at runtime (N3). These required capabilities provide the foundation to improve the support for multidisciplinary engineering process knowledge (N6). The integrated engineering knowledge needs to be provided at production system runtime (N7) to enable decision support during the production system operation phase.

2.4.4 Scenario 4—Maintenance and Replacement Engineering

Scenario 4 will extend the application of information from plant engineering, commissioning, and use phases of the system life cycle toward the "Maintenance and decomposition planning" and "Maintenance" phases of the *Industrie 4.0* value chains in the life cycle. These phases can be improved significantly with better automated support for the assessment of the impact of changes to selected plant components or devices. This automation of change impact support is currently hampered by a weak integration of the diverse kinds of knowledge coming from several roles in the engineering process with the maintenance knowledge during production system operation.

Physical production system components do not necessarily remain stable over the complete lifetime of a production system. These components are subject to certain types of wear-out processes and sporadic faults, which may result in a production system component and (very often) production system that do not work as required. In the context of the example system, the fault of a drive in a conveyor may make parts of the production system inaccessible and therefore (temporarily) lost to the production process.

In practice, the plant maintainer tries to mitigate the risk of a degraded production system with appropriate maintenance and repair strategies, including predictive maintenance and component diagnosis. In case of predictive maintenance, the engineering knowledge about typical component wear-out processes is applied to identify the probability of a fault in the near future, to define component change or repair cycles based on the fault risk, and to ensure proper replacement engineering. In case of diagnosis, run-time measurements are applied to identify dangerous states of components. In the case of the example production system, the knowledge of the usual drive lifetime can be used to change the drive before it is likely to fail. For diagnosis, sound sensors can be installed to measure vibrations in the system and to identify dangerous vibrations announcing the breaking of drive chain parts.

New maintenance capabilities can be reached, if the run-time information can be combined with engineering knowledge. If the supplier of a drive and the plant operator can aggregate their knowledge on drive chain wear-out processes in correlation with drive chain use in the production system and the sound propagation of drive chains, they may enable the development of more sophisticated drive-chain diagnosis methodologies making predictive maintenance actions dependent on sensor measurements and, thereby, automatic control, to name an example.

In addition, existing engineering knowledge has to be applied to ensure the correct integration of new devices within replacement strategies. Here, the involved engineer has to ensure that the new system components fulfill the needs of the application, even if the new component is not an exact copy of the component being replaced.

Such scenarios require a common semantics of engineering and run-time information related to system components and devices. General behavior models of components are required, which exploit engineering information and specific system knowledge, and can be combined with run-time information coming from the production system. The creation of such a common component-behavior semantics is a cooperative task of all roles involved in the design and use of production systems (except for the customer). The plant owner, plant operator, and plant maintainer have to provide knowledge about the production system's run-time behavior. Plant maker, subcontractor, and supplier have to provide knowledge about the production system components and devices and the systems engineering. With the common component-behavior semantics, the plant maintainer can improve his diagnosis and planning skills resulting in an increase of the production system availability.

Similar to Scenario 3, security issues are related to knowledge protection and production system safety leading to the same type of encryption and model validation requirements.

To enable the necessary definition of component-fault-risk models, production system-risk models, and change impact analysis for production system components, several needs for semantic capabilities arise. The knowledge on system requirements, fault risks, and change impact possibilities need to be explicitly represented (N1), the heterogeneous views of system engineers, operators, and maintainers need to be integrated (N2), the modeling of risks and change impact analysis require functions for engineering knowledge access and analytics (N3). In addition to structured data in databases or knowledge bases, the use of semi-structured data, e.g., technical fact sheets that include natural language text, or linked data, e.g., component information from a provider, in the organization and on the Web can help improve the automated support for risk and change impact analysis (N4). Based on these required capabilities, software systems to support risk and change impact analysis in an engineering and operation environment can be designed (N5) to improve the support for multidisciplinary engineering process knowledge (N6). The integrated engineering knowledge needs to be provided at runtime (N7) to enable decision support during diagnosis and replacement in the maintenance phase.

2.5 Needs for Semantic Support Derived from the Scenarios

Major semantic challenges come from the need to provide tool support for processes that build on the heterogeneous terms, concepts, and models used by the stakeholders in production system engineering and operation. Based on the selected scenarios, this section will derive a set of needs that illustrate key capabilities that semantic solutions for multidisciplinary engineering should address. Usage scenarios can be addressed with intelligent engineering applications supported with Semantic Web approaches, e.g., for engineering data integration and storage, for consistency checking, or for the organization and recommendation of engineering components for reuse in a specific target project. Table 2.1 lists the production system engineering usage scenarios (UCx) and derived needs for engineering knowledge modeling and integration capabilities (Nx), similar to the discussion on enterprise service ecosystems in (Oberle 2014).

N1—Explicit engineering knowledge representation. Domain experts in production systems engineering routinely use a wide variety of engineering models, which represent certain aspects of engineering knowledge explicitly (Newen et al. 2011). However, in many cases, the formality/expressiveness of the modeling approaches used does not support the sufficiently complete expression of

Table 2.1 Production system engineering usage scenarios (UCx) and needs for engineering knowledge modeling and integration capabilities (Nx)

Production System Engineering Needs & Use Cases		UC1	UC2	UC3	UC4
N1	Explicit engineering knowledge representation	✓	✓	✓	✓
N2	Engineering data integration	✓	✓	✓	✓
N3	Engineering knowledge access and analytics	✓	✓	✓	✓
N4	Efficient access to semi-structured data in the organization and on the Web	✓	✓		✓
N5	Flexible and intelligent engineering applications		✓		✓
N6	Support for multidisciplinary engineering process knowledge	✓	✓	✓	✓
N7	Provisioning of integrated engineering knowledge at production system runtime			✓	✓

knowledge needed to automate engineering processes for production system engineering. Therefore, there is a need for knowledge representation support, which allows analyzing the requirements for the knowledge to represent and providing the domain experts with appropriate tools for knowledge representation and design.

In all use cases (UC1 to UC4), example knowledge representations are engineering models; also the representation of semi-structured data, such as external documentation like technical fact sheets or component provider data on the Web; access to domain expertise, like expert networks, and collective intelligence systems, like recommender systems for reusable components. For multidisciplinary engineering, common concepts of several stakeholders across disciplines need to be represented to allow sharing engineering artifact views between engineering disciplines. In UC1, engineering process steps require the exchange of engineering tool results, responsibilities of project participants, and progress states of engineering objects. If domain experts use a variety of tools that have only limited knowledge of the engineering project and process, an integrated plant model is very useful to provide a complete view on the project progress. The plant model is based on the capability to store versions of engineering models and tool data to support parallel engineering processes. In UC2, production system engineering project and reusable engineering artifact characteristics support the selection and adaptation of candidate artifacts for reuse. In UC3 and UC4, access to run-time data in the semantic context of engineering models enables the automated interpretation of run-time data elements, even if the production system structure or processes are adapted.

N2—Engineering data integration (common concepts model). The engineering tool network in a typical production system engineering environment contains a collection of tools with heterogeneous data models, which use different terms and data formats for similar concepts (Biffl et al. 2012). Due to this semantic heterogeneity it is difficult, costly, and error prone to provide a consistent production system plant model for parallel engineering. In particular, in multidisciplinary engineering there is the need for an engineering data integration approach, which provides an integrated data model of the common concepts of stakeholders across disciplines to enable the linking of engineering knowledge across disciplines.

Example engineering data integration requirements are in all use cases (UC1 to UC4) including a process for identifying common concepts and for transforming data for the versioned exchange of engineering results in a tool network, e.g., identifying which parts of an electrical engineering plan are semantically linked to part of a mechanical engineering plan. Use cases UC1 and UC2 require data integration between business and engineering/automation levels, e.g., linking an engineering model version to a requirement in the engineering project plan as foundation for checking the consistency of these planning views. In UC2, an integrated plant view is useful to assess the impact of reusing a specific engineering artifact, e.g., understanding which components in different disciplines need to be considered, if a part of a previous system should be extracted for reuse. In UC3 and UC4, engineering plan knowledge has to be linked to run-time data access configurations, e.g., an automated process step to an OPC UA variable, to enable the effective and consistent adaptation of a production system and its associated engineering models at runtime.

N3—Engineering knowledge access and analytics. Knowledge access and analytics in production system engineering builds on the availability of formally represented (N1) and integrated (N2) engineering data in a multidisciplinary engineering environment. Domain experts and managers need basic functions operating on common data model, e.g., reports and analyses based on data coming from an integrated plant model to check the project progress and the quality of the results from parallel engineering. Therefore, effective and efficient mechanisms are needed (a) for querying of engineering models, including versions and changes; and (b) for defining and evaluating engineering model constraints and rules across several views. Example engineering knowledge management requirements in all use cases (UC1 to UC4) include analysis functions for an integrated plant model. In UC1, domain experts can collaborate more efficiently when supported by checks that reveal missing or inconsistent engineering results between two or more disciplines, e.g., a new device that has not yet been addressed in a partner discipline. In UC2, the recommendation of components for reuse needs the capability to analyze both target engineering plans and reuse candidate engineering artifacts from a set of reusable projects and to perform a matchmaking operation between needs and offers. In UC3 and UC4, business managers can plan production services more effectively, if they can quickly integrate engineering knowledge on the current state of the production system with external web resources, e.g., the production capability of the plant with changing customer orders and updated input from component providers and backup producers that all come in via web services or from web pages. The needs N1 to N3 are basic needs of production system engineering derived from the use cases in Sect. 2.4. Addressing the need for explicit knowledge representation (N1) is the foundation for addressing the need for engineering data integration (N2), which in turn is a foundation for addressing the need for engineering knowledge access and analytics (N3). Addressing the basic needs N1 to N3 is the foundation for addressing the advanced needs N4 to N7.

N4—Efficient access to semi-structured data in the organization and on the Web. Engineering process automation today is mostly based on structured data, e.g., in databases or documents that follow a metamodel. In addition to structured data in databases or knowledge bases, the use of semi-structured data, e.g., technical fact sheets that include natural language text, or linked data, e.g., component information from a provider, in the organization and on the Web can help improve the automated support for reuse processes. Therefore, there is a need for more efficient access to semi-structured data in the organization and on the Web. Example requirements for efficient access to semi-structured or Web data are found in UC1, UC 2, and UC4. In UC2, the tool support for reuse depends in many cases on information that comes from outside the organization, e.g., vendor information on components or issues and recommendations posted in software engineering discussion forums. In a similar way during maintenance in UC4, when changing a component for a different component with similar capabilities, domain experts can use the information on the experience of others to assess the impact of a change on the overall system.

N5—Flexible and intelligent engineering applications (automation of engineering). Assuming the capability of knowledge access and analytics (N3) on an integrated production system plant model, intelligent engineering applications, such as defect detection and constraint checking can be designed. In a production system context, these engineering solutions need to be flexible to adapt to the changes in the production system both at design time and at runtime. An intelligent engineering application goes beyond hard-coded programs and is driven by the description of the production system plant. Production systems, in general, are mission- and safety-critical systems, which require a very high level of correctness and completeness of results from the engineering applications. Therefore, the benefits of domain experts depend on the extent to which they can rely on the results coming from the engineering application. Important nonfunctional requirements are highly intuitive user interfaces and scalability. User interfaces have to be easy to understand and use for practitioners, e.g., many engineers want to continue using their well-known best-practice tools, which should be augmented with the knowledge they need in a project. Scalability becomes a major requirement for typical large engineering projects if several projects in a company need to be analyzed together.

In UC2, a reuse system in a company is an example for an engineering application that has to be easy to extend as new or changed types of production systems, components, and devices have to be considered. In UC4, requirements for an engineering application for defect detection and constraint checking are to enable the simple addition of another engineering discipline to extend the range of constraint checking, e.g., new kinds of constraints coming from a specific kind of simulation model. At runtime, flexibility can mean the change of the production system or the addition of new disciplines, such as plant maintenance.

N6—Support for multidisciplinary engineering process knowledge. A major goal of the stakeholders in production system engineering is improving the productivity of the engineering project in a repeatable way, e.g., by avoiding unnecessary work. Therefore, there is a need to support increasing the quality and efficiency of the multidisciplinary engineering process by representing engineering process responsibilities and process states linked to the production system plant model. This need extends N3, which focuses on knowledge regarding engineering artifacts, with respect to knowledge on engineering processes.

Example engineering process requirements can be found in all use cases (UC1 to UC4). In UC1, the maturity state of engineering results and the responsibilities of domain experts and organizational units enable more efficient planning and monitoring of the engineering process. In UC2, both projects and project-independent reuse processes need to be defined and need access to engineering knowledge for effective reuse recommendations. In UC3 and UC4, the inclusion of all relevant stakeholders from production system planning, engineering, operation, and maintenance, enables more effective system engineering and overall production system process management.

N7—Provisioning of integrated engineering knowledge at system runtime. In a flexible production system context, domain experts need engineering knowledge at runtime to assess in a situation, which needs changing the system, the set of options for a successful change. In addition, changes have to be documented in a way that supports future change analysis. Therefore, there is a need for providing integrated engineering knowledge at system runtime beyond simple pdf printouts of engineering plans. The knowledge has to be available in a sufficiently timely manner to support applications that depend on reacting in time to real-time processes.

Example requirements for engineering knowledge at system runtime can be found in UC3 and UC4. In UC3, if a flexible production system changes, the engineering knowledge on the system structure, available components, and wiring to signals has to be updated in a knowledge base to enable the correct linking of data from sensors to engineering objects. In UC4, a current engineering knowledge on the system is necessary to correctly assess the impact of a changed component on the overall system behavior.

The collection of needs is the foundation to investigate how well Semantic Web capabilities can address these needs for semantic support.

2.6 Summary and Outlook

This chapter introduced key elements of the life cycle processes of engineering production systems toward *Industrie 4.0*, focusing on heterogeneity in the engineering of production systems. We introduced four usage scenarios to illustrate key challenges of system engineers and managers in the transition from traditional to cyber-physical production system engineering environments. These scenarios allow semantic researchers and practitioners to better understand the challenges of production system engineers and managers, to define goals and assess solution options using Semantic Web technologies. Major semantic challenges come from the need to provide tool support for processes that build on the heterogeneous terms, concepts, and models used by the stakeholders in production system engineering and operation. From the challenges illustrated in the four scenarios, we derived needs for semantic support from the usage scenarios as a foundation for evaluating solution approaches.

Outlook. Chapter 3 will provide a basic assessment on how well Semantic Web approaches seem suitable to address these needs compared to alternative approaches. Table 2.2 summarizes the assessment discussed in Chap. 3 to inform potential users on the relevant Semantic Web capabilities and their match to the needs identified in Sect. 2.5. Table 2.2 considers how important each capability (Cx) is to address the set of needs to enable selecting a suitable set of Semantic Web technologies as starting point for planning a solution strategy.

The qualitative evaluation of Semantic Web capabilities shows good coverage of the previously identified production systems engineering needs. The conclusion

Table 2.2 Semantic Web technology capabilities (Cx) and needs for engineering knowledge modeling and integration capabilities (Nx); strong (++) and moderate (+) support of a need by a capability

Semantic Web Capabilities & Needs		N1	N2	N3	N4	N5	N6	N7
C1	Formal and flexible semantic modeling	++	+	++	+	+	+	+
C2	Intelligent, web-scale knowledge integration	+	++	++	++	++	++	
C3	Browsing and exploration of distributed data set			+	++	+	+	+
C4	Quality assurance of knowledge with reasoning					++	++	++
C5	Knowledge reuse	+	+	++			++	+

Chap. 15 will compare the strengths and limitations of Semantic Web approaches with alternative approaches that are also available to production systems engineers. Engineers and managers from engineering domains can use the scenarios introduced in this chapter to get a better understanding of the benefits and limitations coming from using Semantic Web technologies in comparison to alternative approaches as foundation to select and adopt appropriate Semantic Web solutions in their own settings.

Semantic Web researchers and practitioners, who need to define goals for intelligent engineering applications and to assess solution options using Semantic Web technologies, can compare their application scenarios to the usage scenarios introduced in Sect. 2.4 to derive needs that can be addressed well by the capabilities of Semantic Web technologies, see Chaps. 3 and 15.

Acknowledgments Parts of this work were supported by the Christian Doppler Forschungsgesellschaft, the Federal Ministry of Economy, Family and Youth, and the National Foundation for Research, Technology and Development in Austria.

References

AQUIMO. aquimo—Ein Leitfaden für Maschinen- und Anlagenbauer. VDMA (2010)

Biffl, S., Mordinyi, R., Moser, T.: Anforderungsanalyse für das integrierte Engineering—Mechanismen und Bedarfe aus der Praxis. ATP Edition **54**(5), 28–35 (2012)

Binder. J., Post, P.: Innovation durch Interdisziplinarität: Beispiele aus der Industriellen Automatisierung. In: Anderl, M., Eigner, U., Sendler, R., Stark, R. (eds.) Smart Engineering, acatech DISKUSSION, pp. 31–43 (2012)

BMBF. Produktionsforschung—57 erfolgreiche Projekte für Menschen und Märkte. Bundesministerium für Bildung und Forschung (2007)

Drath, R., Lüder, A., Peschke, J., Hundt, L.: An evolutionary approach for the industrial introduction of virtual commissioning. In: Proceedings of the 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE (2008)

Götz, O.: AutomationML—Effizienz für den Engineeringprozess. In: SPS-Magazin, 5/2013, pp. 43–45 (2013)

Grote, K.-H., Feldhusen, J. (eds.) Dubbel—Taschenbuch für den Maschinenbau. 24. aktualisierte Auflage. Springer Vieweg (2014)

Hundt, L., Lüder, A., Estévez, E.: Engineering of manufacturing systems within engineering networks. In: Proceedings of the 15th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE (2010)

Jacob, M.: Management und Informationstechnik. Eine Kompakte Darstellung. Springer (2012)

Kagermann, H, Wahlster, W, Helbig, J. (eds.): Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0—Deutschlands Zukunft als Industriestandort sichern. Forschungsunion Wirtschaft und Wissenschaft. Arbeitskreis Industrie 4.0 (2013)

Kiefer, J.: Mechatronikorientierte Planung automatisierter Fertigungszellen im Bereich Karosseriebau. Dissertation. Universität des Saarlandes. Schriftenreihe Produktionstechnik Band 43 (2007)

Lindemann, U.: Methodische Entwicklung technischer Produkte: Methoden flexibel und situationsgerecht anwenden. 3rd edn. Springer (2007)

Lüder, A., Hundt, L., Foehr, M., Wagner, T., Zaddach, J.J.: Manufacturing System Engineering with Mechatronical Units. In: Proceedings of the 15th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE (2010)

Lüder, A., Foehr, M., Hundt, L., Hoffmann, M., Langer, Y., Frank, S.: Aggregation of engineering processes regarding the mechatronic approach. In: Proceedings of the 16th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE (2011)

Lüder, A.: Integration des Menschen in Szenarien der Industrie 4.0. In: Bauernhansl, T., Vogel-Heuser, B., ten Hommel, M. (eds.) Industrie 4.0 in Produktion, Automatisierung und Logistik. Springer Fachmedien, pp. 493–507 (2014)

Maga, C., Jazdi, N., Göhner, P., Ehben, T., Tetzner, T., Löwen, U.: Mehr Systematik für den Anlagenbau und das industrielle Lösungsgeschäft—Gesteigerte Effizienz durch Domain Engineering. In: at—Automatisierungstechnik 9, pp. 524–532

Newen, A., Bartels, A., Jung, E.-M.: Knowledge and Representation. Centre for the Study of Language & Information (2011)

Oberle, D.: Ontologies and reasoning in enterprise service ecosystems. Inform. Spektrum **37**(4) (2014)

Pfrommer, J., Schleipen, M., Beyerer, J.: PPRS: production skills and their relation to product, process, and resource. In: Proceedings of the 18th IEEE Conference on Emerging Technologies and Factory Automation (ETFA). IEEE (2013)

Science. Engineers Council for Professional Development. Science **94**(2446), 456 (1941)

Schäffler, T., Foehr, M., Lüder, A., Supke, K.: Engineering process evaluation—evaluation of the impact of internationalization decisions on the efficiency and quality of engineering processes. In: Proceedings of the 22nd IEEE International Symposium on Industrial Electronics (ISIE). IEEE (2013)

Schäppi, B., Andreasen, M., Kirchgeorg, M., Rademacher, F.-J.: Handbuch Produktentwicklung. Hanser Fachbuchverlag (2005)

Schnieder, E.: Methoden der Automatisierung. Vieweg, 380 pp. (1999)

Terkaj, W., Tolio, T., Valente, A.: Focused flexibility in production systems. In: Hoda, A., ElMaraghy (eds.) Changeable and Reconfigurable Manufacturing Systems. Springer Series in Advanced Manufacturing, Springer, pp. 47–66 (2009)

Ulrich, K., Eppinger, S.: Product Design and Development. McGraw-Hill Higher Education (2007)

VDI. Entwicklungsmethodik für mechatronische Systeme. VDI-Richtlinie 2206. Verein Deutscher Ingenieure e.V (2004)

VDI. Digitale Fabrik—Grundlagen. VDI-Richtlinie 4499–1. Verein Deutscher Ingenieure e.V (2008)

VDI. Fabrikplanung. VDI-Richtlinie 5200. Verein Deutscher Ingenieure e.V (2009)

VDI. Digitale Fabrik—Blatt 2. VDI-Richtlinie 4499-1. Verein Deutscher Ingenieure e.V (2009)

VDI. Engineering of industrial plants. Evolution and Optimizations. Fundamentals and Procedures. International Standard. ICS 35.240.50 Part 1–Part 4. Verein Deutscher Ingenieure e.V.. VDI/VDE 3695, November 2010 (2010)

VDI/VDE. Industrie 4.0—Wertschöpfungsketten. VDI/VDE Gesellschaft Mess- und Automatisierungstechnik. Statusreport, April 2014 (2014)

Wegener, D.: Industrie 4.0—Chancen und Herausforderungen für einen Global Player. In: Bauernhansl, T., ten Hompel, M., Vogel-Heuser, B. (eds.) Industrie 4.0 in Produktion, Automatisierung und Logistik. Springer (2015)