CIRP Annals - Manufacturing Technology 68 (2019) 727–750

Contents lists available at ScienceDirect

CIRP Annals - Manufacturing Technology

journal homepage: http://ees.elsevier.com/cirp/default.asp

Development capabilities for smart products

Tetsuo Tomiyama (1)[a,*], Eric Lutters (1)[b], Rainer Stark (2)[c], Michael Abramovici (2)[d]

[a] Japan Educational Foundation, Shinjuku, Tokyo 160-0023, Japan
[b] Department of Design, Production and Management, University of Twente, 7500 AE Enschede, The Netherlands
[c] Industrial Information Technology, IWF, TU Berlin and Virtual Product Creation, Fraunhofer (IPK), 10587 Berlin, Germany
[d] Ruhr-Universität Bochum, 44801 Bochum, Germany

ARTICLE INFO

Article history:
Available online 18 June 2019

Keywords:
Design
Product development
Model-based systems engineering

ABSTRACT

Smart products supported by new step-changing technologies, such as Internet of Things and artificial intelligence, are now emerging in the market. Smart products are cyber physical systems with services through Internet connection. For example, smart vehicles equipped with advanced embedded intelligence are connected to other vehicles, people, and environment, and offer innovative data-driven services. Since smart products are software-intensive, data-driven, and service-conscious, their development clearly needs new capabilities underpinned by advanced tools, methods, and models. This paper reviews the status and trends of these emerging development technologies such as model-based systems engineering and digital twin.

1. Introduction

With such features as intelligent, connected, and autonomous, new types of products are appearing in the market, qualitatively differentiating themselves from previous product generations. In this keynote paper, these products are called 'smart products' [3,12,34,181]. Although 'smart' is used as a synonym for almost any concept to mean 'intelligent', 'clever', 'nifty' or even just 'advanced', the 'smart product' concept includes much wider connotations. This means that it is insufficient just to discuss their development only from the viewpoint of intelligence. This paper is an attempt to sharpen the smart product concept and to identify issues, challenges and research needs associated with the technologies, methods, tools, and models for the smart product development.

Smart products exhibit several distinctive functional capabilities and technical features that differentiate them from conventional products [3,12] (see Section 2.2). Some of these are listed below.

- *Intelligence*: Intelligence is critical for smart products, as this forms the basis for other (secondary) features. For example, when smart products interact with human users, their 'intelligence' features play the key role. In contrast, obviously intelligence is indispensable to autonomy of smart products.
- *Connectedness*: Connectedness addresses the connectivity of smart products with other products, humans, data and services through networks. This also implies potential threat of cyber-attacks and necessity of private data protection.
- *Service integration*: Intelligence and connectedness features are indispensable for smart products to create new 'services'. An example of new types of services enabled by intelligence and connectedness is mobility service with autonomous driving vehicles.
- *Data driven*: Smart products connected over the network form a platform to collect data and information through their sensing capabilities. The collected data can be processed with data analytics techniques [18,154] for both technical and commercial purposes. For instance, the data collected through the network could be used to make lifecycle-related decisions such as maintenance, while the latter can be used to extract user preferences and profiles that are commercially important.

Fig. 1 depicts the evolution of the smart product concept, starting from just mechatronics products, intelligent mechatronics systems, Cyber-Physical Systems (CPS) [19,129,169,170], and then to smart products. Mechatronics products rely on (mostly feedback) control of the mechanical system using electronics. However, as software executes control, sophisticated control beyond simple feedback control is possible. This leads to so-called intelligent mechatronics products. CPS extends intelligent mechatronics products with cyber communication with other CPS or any agents on the Internet. Smart products are defined as 'CPS, which additionally use and integrate Internet-based services in order to perform a required functionality' [3].

According to this definition, smart products are also smart product-service systems as an extension of PSS (Product-Service System) [28,86,231]. Therefore, smart products are part of a

* Corresponding author.
E-mail address: t.tomiyama@me.com (T. Tomiyama).

https://doi.org/10.1016/j.cirp.2019.05.010
0007-8506/© 2019 Published by Elsevier Ltd on behalf of CIRP.

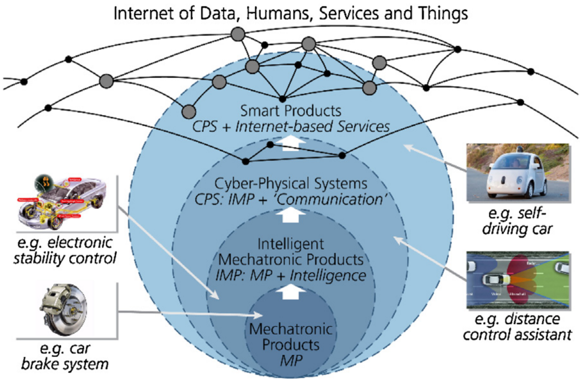

Fig. 1. Smart products as CPS integrating internet-based services.

complex business ecosystem (see Fig. 2) including different stakeholders (e.g., users, customers, manufacturers, service providers), physical environment (enterprise, surrounding physical infrastructure) and other connected product devices.

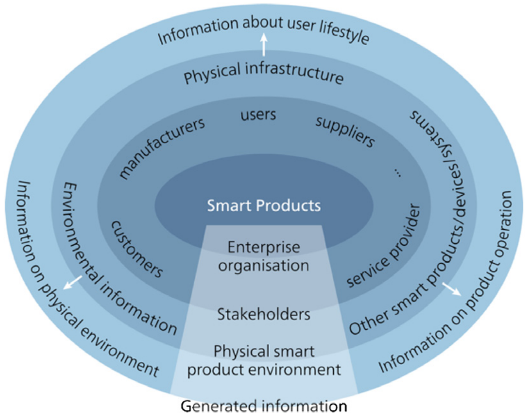

Fig. 2. The ecosystem of smart products.

This ecosystem supports two types of value generation mechanisms. The first one is traditional but explicit value for users generated by user experiences [92,172]. ISO 9241-210 defines user experience as 'a person's perceptions and responses that result from the use or anticipated use of a product, system or service' [104]. Through localised or personal modification of the product, it is also expected to improve its fit to the user or local community. For example, a smart phone can easily accommodate the needs and preferences of an individual user for superb experiences that can create value for the user.

The concept of (user) experiences of (smart) products is strongly advocated by the human–machine interface community [92]. In the context of product design, Dassault Systèmes emphasises 'experiences' in their tool set [89].

The second type of value creation of this ecosystem is business value generated by collecting data about numerous events that take place surrounding smart products. The data will be collected through distributed sensors and processed to extract further information. This idea shows the relevance of the IoT (Internet of Things) technologies [82], data mining techniques [18], and AI (Artificial Intelligence) technologies represented by deep learning [39,128,191]. The data will also describe how these smart products are used and potentially bring up with enormous business opportunities (i.e., business value), because they illustrate the life style of users.

Fig. 3 illustrates the data-driven value generation mechanism of the ecosystem. It suggests that developing smart products is not just about technically developing physical products but about covering the entire scope of the ecosystem including services, mechanisms to collect, process and generate data regarding the use of the products and user experiences. The development should also take business models and objectives into consideration.

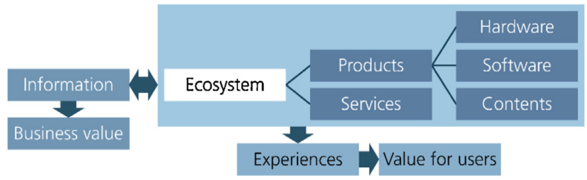

Fig. 3. Data-driven value generation of smart products.

This keynote aims at establishing an overarching concept of smart products as well as reviewing technologies needed to develop smart products. Following this introduction, Chapter 2 will briefly overview various types of smart products and then list up their features such as characteristics and architecture. Chapter 3 will overview enabling product technologies that are fundamental to build smart products. In contrast, Chapter 4 will examine methods, technologies and supporting tools to develop smart products and their services. Existing technologies might not be suitable for coping with the many new features of smart products related to, e.g., intelligence. Therefore, Chapter 5 will look into some research challenges of smart product development, including MBSE (Model-Based Systems Engineering) [262,83]. Chapter 6 will summarise the discussion and identify future research directions. Chapter 7 concludes the paper.

2. Smart products

2.1. Examples of smart products

As the number of smart products is growing rapidly, they are becoming ubiquitous [157]. In order to better understand the features and the challenges of this new product type the following section describes several examples of smart products:

- Smart phones (see Section 2.1.1)
- Smart speakers (see Section 2.1.2)
- Smart vehicles (see Section 2.1.3)
- Smart robots (see Section 2.1.4)
- Smart factory (see Section 2.1.5)

Further examples for single smart products are smart lights [282], smart thermostats [283] and smart meters [113]. An example of a smart product system is a smart home [43], smart grids [29,69,83,242], and for a complex smart system of systems a smart city [237,294].

2.1.1. Smart phones

In 2007, Apple introduced its first smart phone 'iPhone'. Originally, this was marketed as a combined product of phone, iPod (Apple's MP3 music player), PDA (Personal Digital Assistant) and camera, but without GPS, Apps or high-speed connectivity [238]. At that stage, it was merely a sum of these capabilities without integration. However, the 2008 model (iPhone 3G) made a step change by introducing the concept of Apps with possibility of intertwining these system capabilities, together with the introduction of iOS App Store. The system components of smart phones can be clustered as in Table 1.

Based on these capabilities, Apple created an ecosystem of delivering products and services including [51]:

Table 1
System components of smartphones.

Categories	Components
Mechanical hardware	Touchscreen that disinguishes 'smart phone' from other phones, vibrator
Electronics hardware	CPU, RAM, flash memory, speaker, microphone, battery
Communication hardware	Mobile communications, Wifi, USB, Bluetooth
Sensors	Cameras, accelerometer, GPS, light sensor, touch sensor
Software	iOS, Android, Apps

- iTunes Music Store (music, videos, movies)
- App Store (applications)
- Apple Music (music subscriptions)
- iCloud (cloud storage)
- Apple Pay

According to Fig. 1, iPhone as hardware is a CPS. Through services made available with the Internet connectivity, various Apps fulfil user needs purposefully [238] which converted iPhone into a true smart product. An example of such services is Google's traffic information on the Google Maps app when it is used as a navigation system [279]. Using GPS and other technologies, the phone determines its location. In a certain periodic interval, this information is sent to the central server, from which the speed of the vehicle is calculated. Information from a large number of phones using the application will be compiled to determine the average speed of the traffic on a certain road. This will be fed back to the map information with different colouring.

For this smartness of smart phones, Internet connectivity and collecting information from many anonymous users play a crucial role. The operating system (iOS) as well as application software is automatically updated through the Internet connection. Many applications take advantage of being connected to (cloud) servers, allowing users to access a variety of information as well as computing capabilities. In addition, it is important to note that the way smart phone applications are organised allows a high degree of individualisation in the choice of applications.

2.1.2. Smart speakers

So-called smart speakers [38] can exhibit highly sophisticated interaction capabilities with users, which allows them not only to play or control music but also function as an interactive bi-directional interface for information search services. The central function as a 'voice input/control device' is attained by advanced technologies for voice recognition and natural language understanding based on deep learning algorithms [128] made available through the Internet connection. Although obviously the speaker itself is no way smart, the smart speaker system as a system is able to learn about the user, her/his preferences, and behavioural characteristics.

The use history and user preferences obtained from these communications are valuable business data about user's life style. The smart speaker might be equipped with additional sensors (e.g., for temperature and brightness of the room), which will enable collection of characteristics of user activities, user profiles, and other types of business information. Therefore, smart speakers act as IoT sensing devices to collect information for the data-driven business ecosystem.

2.1.3. Smart vehicles

The automotive industry is now undergoing revolutionary transformations [126]. Sometimes captured as CASE (Connectivity, Autonomous, Shared & Services, Electric), these transformations include:

- Drive train change (more electrification)
- Autonomous driving
- Connectedness
- Shared (servicification such as MaaS (Mobility as a Service))
- Yearly updating

The drive train technology is shifting towards more electrification for a number of reasons, including sustainability which is becoming one of the top priorities of the industry (e.g., [149]). However, it has to be noted that autonomous driving vehicles for sharing services need to be electric driven for the sake of easy and safe refuelling.

Toyota Prius, the first hybrid electric automobile, launched in 1997, quickly became one of the bestselling cars in the world [285]. It aimed at improving fuel efficiency by running the ICE (Internal Combustion Engine) at the best fuel consumption range, while keeping the emission as clean as possible. All-electric vehicles (including lithium ion battery-based and fuel cell-based) with comparable performance to ICE-based vehicles did not appear until 2008 when Tesla Roadster was launched followed by Mitsubishi iMiEV in 2009 [281]. Tesla S introduced in 2012 was not only a luxury all-electric vehicle but also equipped with a number of features of smart vehicles including [284]:

- AWD (All Wheel Drive) with dual-motors
- Supercharging for 60–100KWh lithium ion battery
- Autopilot (semi-autonomous driving) using radar and cameras
- Connectivity to the Internet
- Automatic software upgrading

Autonomous driving is a technology currently looked into by a wide range of players from car manufacturers (such as GM, Audi, Tesla, and Toyota) to suppliers (such as Bosch, Denso and NVIDIA) to service providers (such as Waymo and Uber). While early efforts were mostly guided vehicles, truly autonomous driving cars controlled by visual information captured by stereo cameras were first developed by MITI's Mechanical Engineering Laboratory in Japan in 1977 [290]. The group at Carnegie Mellon University's NAVLAB funded by DARPA developed an autonomous van in the late 1980s and succeeded in trans-continental drive in 1995 [230]. This effort eventually led to Alphabet's (Waymo's) driverless vehicle that surprised the world [280]. Technological architectures for autonomous vehicles are proposed in Refs. [16,138,226]. Besides sensing technologies (such as laser-, vision-, ultra-sonic-, radar-based range scanning and vision recognition), advanced navigation based on SLAM (Simultaneous Localisation and Mapping) and AI-based recognition were breakthrough technologies.

Internet based services in transportation opened a possibility of new service called MaaS (Mobility as a Service) [286]. This takes a form of a range of mobility services including share ride, quick car rental, subscription-based car rental, and non-traditional taxi service. Replacing traditional public transportation vehicles with autonomous vehicles would increase convenience as well as efficiency in less-populated remote areas and aging societies. Already a number of companies entered into the market in various cities on the globe. Autonomous driving is considered a next chapter for this type of MaaS.

Internet connectivity enabled also 'yearly updating', which applies more regular software updates for the operating system and applications in a similar way as smart phones. Traditionally, IT equipment (e.g., PCs and handheld devices) and industrial machines have been maintained by upgrading the software in order:

- to improve functions of the software,
- to fix bugs, and more recently
- to tackle cyber security threats.

However, this was never the case for home appliances and cars, although future smart vehicles will become software intensive just like many other mechatronic products and CPSs. This trend of frequent updating will become dominant in case of more electric, autonomous, shared and connected vehicle to take advantage of their full capabilities [126].

Although it is not yet clearly advocated, besides those five trends, individualisation is also considered another advantage of smart vehicles. The Intelligent software-based control system can be flexible enough to be individualised according to the preferences of individual owners (see Section 2.2.10).

Not only drive train technology, but overall car technologies are shifting towards 'more electric'. Currently, a typical modern high-end passenger car may be equipped with more than 100 ECUs (Electronic Control Units) for, e.g., engine control and break control, but it is expected not to increase drastically due to the necessity of managing complexity and wiring issues [261]. The size of control software has significantly increased, however. Nowadays, a high-end passenger car is running more than 150 million lines of code [37]. Consequently, the control software architecture is moving from a 'federated architecture' to an 'integrated one" [60]; for example, AUTOSAR (AUTOmotive Open Software Architecture) is jointly developed by a consortium of various industrial partners to allow such strategy [26].

In response to this trend, Volkswagen recently announced the shift towards electric vehicles. The core is the newly developed architecture called MEB (Modular Electrification Toolkit) [260]. The number of ECU will be significantly reduced to deal with the complexity issues associated with the number of ECUs and programme size.

Fig. 4 illustrates how technological elements surrounding smart vehicles underpin the main trends (electrification, autonomous driving, connected, servicification, and yearly upgrading) and how these trends lead to future new service goals. (Fig. 4 depicts only four examples but there can be many more.) For example, MaaS will be supported by autonomous driving, connectedness, and servicification. Based on continuous monitoring using onboard sensors that collect data to be analysed using data analytics, continuous maintenance [190] will become feasible to monitor, diagnose and repair the vehicle proactively. Similarly, continuously collected operational data of the vehicle as well as the driver can be used to determine the option and premium of the insurance. This will lead to a customised insurance policy for individual drivers.

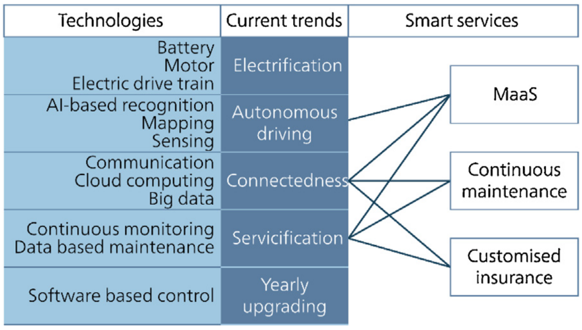

Fig. 4. New smart services of smart vehicles.

2.1.4. Smart robots

Robots [210] have made rapid proliferation in manufacturing industry to replace human labour in material handling, assembling, welding, and inspection. Early robots were mostly equipped with anthropomorphic arms of up to six degrees of freedom. These are typically used for highly dumb, dirty, dangerous, and demanding tasks in which pre-programmed repetitive behaviours without locomotion capabilities suffice. Robots are also finding applications in logistics, construction, maintenance, and many other service industries, as well as household. Advances in robotic technologies have made it possible to build robots with such features as redundant DoF arms, dexterous robot hands, and biped locomotion capabilities. The invention of Honda's Asimo [192] was epoch-making in that this biped humanoid robot opened up possibilities for applications of walking robots that are not bound to wheels.

However, robots are not necessarily manipulator-based. One can define a robot as 'a machine capable of carrying out a complex series of actions automatically, especially one programmable by a computer' (Oxford English Dictionary). Therefore, some robots are not meant for mechanical automation. For example, drones or UAVs (Unmanned Aerial Vehicles) have been put to practical use for picture/film shooting, surveillance, inspection, and light load transport [74]. The concepts of android and social robot focus on interactions with humans, so the appearance and conversational capability with humans are more crucial [75]. In this context, famous Asimov's three laws of robotics can be given contemporary meanings [24].

Recently, robotic applications are increasingly requiring intelligence for various purposes including:

- task planning
- navigation (for mobile robots)
- control and planning of locomotion and manipulation
- communication with other agents (human, robot, machines, and environment).

These capabilities will be executed using a variety of sensors to capture information from objects and the environment. For example, a vision-based part assembling robot should be able to locate and pick randomly placed parts. This robot will be able to deal with abnormality of the part it picked up by comparing the image on the camera and the shape information obtained from CAD. An 'intelligent caretaking robot' will be taking physical care of elderly people and patients. It can also take care of them mentally by conversing with them. An intelligent robot can be regarded as a mechanism to deliver such intelligent functions through various end-effectors and locomotion capabilities. From the viewpoint of smart products, this is the point where robots are interfacing between the physical world and the cyber world.

2.1.5. Smart factory

The 'smart factory' [162] (Fig. 5) is an example of a complex smart product system within the shop floor. Fig. 5 indicates that a smart factory consists of various elements including machine tools, robots, material handling systems (such as AGVs and conveyors), controllers, and mechanisms to collect, process and feedback information. These individual elements need to be 'smart', too. For example, a smart machine tool should be equipped with a variety of sensors to capture its status and performance data.

Fig. 5. Elements of the smart factory.

The smart factory is direct implementation of the so-called Industry 4.0 concept, which is a data-driven or -centric operational principle of industrial systems [106]. The smart factory is based on CPS or IoT advocating 'connectedness' starting from low level sensors, barcodes, and RFIDs [151,269,302] embedded everywhere in the factory including raw materials, workpieces, and components up to machines, material handling systems, and robots [107,127,132,299]. Robots can be integrated in the process with human workers in an assistive manner [56]. Information about all processes and every event is generated and stored, contributing to the realisation of a 'smart enterprise' [97,114,137,187,264]. This means that such data will be analysed with data analytics methods for the purposes of performance management, scheduling, machining quality control, and maintenance of the machine tool [135,208,293]. This bi-directional data flow to and from the machine tool underpins the smartness of the machine tool in the smart factory. Industrial Internet [100] advocates similar principles and concepts. A Reference Architecture Model for smart enterprises (RAMI) has been started as part of Platform Industry 4.0 in Germany [185].

The past efforts within the research community, such as holonic [1,136,249,251,252], bionic [243] and fractal manufacturing systems [270], all pursued data intensive, small-lot, customisable, flexible, responsive, and agile manufacturing [228]. In this context, these past efforts laid foundation for smart factory.

In Ref. [162], the concepts and the architecture of smart factories are analysed. The advantages of the smart factory are multi-fold, which include:

- Increased data generation that can lead to performance improvements through data analytics
- Agile responsivity to failures and abnormality
- Increased changeability and flexible reconfigurability for smaller production volume (eventually one-off).
- Improved traceability of components and quality.

2.2. Features of smart products

Smart products possess some functional capabilities in common that distinguish themselves from conventional products. These include intelligent interactions with human users, autonomy, dependability, and individualisation. These functional capabilities result from various technical features such as intelligence and resilience backed up by connectedness, sensing capabilities, and reconfigurability. Below, these functional capabilities and technical features are discussed in detail.

2.2.1. Intelligence

Although 'intelligence' is very casually discussed, no clear definition does exist because the notion is used in many fields. However, the intelligence feature is one of the most distinctive features of smart products and includes such capabilities as recognition (voice, vision, language, etc.), reasoning, and learning (or improving).

In the context of smart products, clearly intelligence has several meanings. The first is intelligent interactions with human users (see Section 2.2.5, e.g., natural language interfaces such as Alexa of Amazon and Siri of Apple). The second aspect is so-called intelligent control that goes beyond the traditional feedback control. For example, advanced robots use various types of modern high-level control algorithms. Third, intelligence is useful for other features such as autonomy and reconfigurability.

Traditionally, intelligence was implemented using so-called AI technologies (see Section 3.2) which follow two major approaches. One is the traditional symbolic logic-based methods to explicitly implement knowledge [191], which was experiencing limitations in knowledge acquisition and maintenance. A breakthrough came from so-called machine learning methods based on Deep Neural Network (DNN) [88], which were a sophisticated version of Artificial Neural Network (ANN). For specific applications, such as vision recognition, a special type of DNN called convolutional neural networks perform outstandingly. The learning capabilities facilitate continuous improvement of the performance, which is also favourable for smart product applications. The ownership of data and knowledge created as a result of this learning process is yet to be discussed.

However, there are two challenges associated with DNN. One is training of the network, which assumes clean high-quality teaching data. Preparing such teaching data can be extremely difficult, if not impossible. The other challenge is that reasoning outcomes do not have explanations as DNN is a black box. This is the difference between DNN and traditional AI approaches.

If real time responses are needed, as this may require substantial computing power, the Internet connection and the client-server architecture might be desirable. For example, the current implementation of natural language interfaces (such as Alexa of Amazon and Siri of Apple) assumes network connection.

2.2.2. Connectedness

Connectedness to other agents on the Internet constitutes the foundation for all the features of smart products. It will enable collection of data but also facilitate self-identification (i.e., from others) and determination of the location.

There are different types of computing paradigms related to connectedness that have evolved through historical development. Ubiquitous computing [276] (or pervasive computing) is a concept in which 'the nonintrusive availability of computers throughout the physical environment, virtually, if not effectively, invisible to the user' [277]. Ubiquitous computing is perhaps the most direct ancestor of IoT. Cloud computing is defined as 'shared pools of configurable computer system resources and higher-level services that can be rapidly provisioned with minimal management effort, often over the Internet' [203]. Cloud computing liberated computing from particular servers and virtually removed the boundaries of computing capabilities. For example, smart products do not have to be limited to locally available computing resources.

However, the introduction of the IoT technologies typically led to huge data traffic as well as heavy loads to the central servers, these computing and storage capabilities were transferred to distributed non-central (edge) nodes located closer to the user. This is called edge computing or fog computing [59,206] and it is expected to exhibit computing performances better than cloud computing which is relying mainly on centralised servers. Edge computing uses computing nodes that are distributed in the network to maximise the system throughput and to minimise delays.

The advances of wireless communication technologies are now pushing the boundaries of wireless communication to 5G and beyond [17,178]. With the anticipated communication speed of 5G (20 Gbps), it is expected that many step-changing technologies, such as (near) real time communication without latency and peer to peer communication, can be introduced. When applied to moving agents (e.g., smart vehicles, mobile robots, etc.), these will facilitate totally different level of functionalities such as high-resolution image link. Even immobile assets will become free from wiring. In addition, many applications will have an opportunity to shift from local to cloud/edge/fog computing. The smart product ecosystem in Fig. 2 requires such high-speed communication if it has to be really data-centric or data-driven.

Connectedness (with high-speed communication) has drawbacks as well. These issues will be discussed in Section 2.3.5.

2.2.3. Servicification

Smart products are by definition product-service systems. They collect operational and use data through sensors to enhance service purposes, such as maintenance and life cycle management.

DMG Mori has been deploying their MTConnect system to monitor, collect, maintain their machine tools in the market [62,164]. Originally, the system was developed as a remote monitoring system. It now plays the core role of their smart factory solutions. The system basically remotely monitors operations and conditional parameters of the machine and sends it to the server. This information is used for managerial purposes as well as maintenance. With these capabilities, the company now claims that they can provide their customers with Industry 4.0 solutions.

Rolls-Royce started power-by-hour services in the 1980s and this TotalCare service is now generating more than 50% of turnover of its civilian jet engine division. For airlines, TotalCare ensures engines' performance with predictable costs, while for Rolls-Royce it generates steady revenue. The service can include real time monitoring capabilities which collect real time engine data to be used for improving maintenance. Obviously, this smart product design will bring up with maximum benefits when aligned with service design. In 2002, the company launched a new programme that aimed at designing new products truly 'designed for service' [91].

Komatsu is known for its successful service called 'Komtrax' [109,116] which collects and reports the status of their construction machinery (such as bulldozers and excavators). The original purpose was to locate a machine using GPS but now it collects information including time when the engine started and shut down, total engine running time, service meter readings, and information about an eventual unauthorised machine use. These types of information are fully used for management and maintenance purposes, but the company is trying to extend its use to the design phase, too.

This type of capabilities can be critical for smart vehicles used for autonomous MaaS service. It is estimated that actual usage rate of these vehicles could be above 50%, while currently more than 90% of passenger cars are used for only 5% of the time [66]. Technically, this implies the need for much higher reliability and dependability. Service-wise, new methods for providing services much faster and more effective, including constant real-time health-monitoring, predictive maintenance and proactive maintenance.

2.2.4. Sensors and sensing

Smart products collect information about their environment as well as their own states and status through massively deployed sensors. These types of information are used in short term for the control purposes as well as in longer term, e.g., for autonomous operations.

IoT is very relevant to smart products [21,27,90,123,159,248,287]. In particular, the development of wireless RFID sensors [269] and IoT chips for wireless communication with low power consumption (which are called 'smart sensors') [289] are the game changer that allows massive deployment of sensors throughout, e.g., machines, buildings, factories, and even social infrastructure. These sensors have typically monitoring applications, e.g., pressure, temperature, and strain, e.g., for process data mining and maintenance purposes [80,227,256].

2.2.5. Smart human interaction

Smart human interactions require both verbal and visual interfaces, and haptic or other forms of interaction [48,171]. Note that autonomous systems are somehow contradicting to this feature of smart products, because autonomy requires least human interaction by definition. However, it should be interpreted that the best smart product is one that requires only minimum communication with human users in a very smart way.

The use of (especially) AR (Augmented Reality) technologies as a human interface to smart products is considered key to understanding information about user's behaviour. For example, traditional screen-based interfaces do not allow bi-directional interactions, unless the screen is a touch screen. However, even with a touch screen, the amount of information about the user's behaviour is limited. With AR technologies, information obtained through interactions can be richer, because of the availability of the user's focus.

2.2.6. Autonomy

A high level of autonomy is a critical feature of smart products. Autonomous systems are capable of sensing external information, intelligently making decisions without human intervention, and performing actions in response to the situation. They operate and survive by adapting to the external changes and internal deterioration, which is the feature of resilience. However, this also means lack of surveillance. Therefore, it is critical that an autonomous system is dependable (see the next section). For example, UAVs (often a drone or a deep-sea robotic submarine) can be self-driving (flying or swimming) without remote controls. They should not do any harm to any external parties or get lost in accomplishing their mission, either.

An autonomous system can also be self-learning, which represents non-deterministic behaviour. An example of autonomous self-learning is a modular product that configures itself after assembling the modules, which is self-organisation.

The interest in autonomy is growing in various fields. For instance, the UK government published a series of white papers on robotics and autonomous systems on such application areas as surgical robotics; space; robotics in social care; manufacturing; agriculture; robotics for emergency response, disaster relief & resilience; resilient infrastructure; urban robotics & automation; and AI & robotics [245].

2.2.7. Dependability

According to the International Electrotechnical Commission (IEC), dependability is an ability to perform as and when required [101]. It is 'a collective term for the time-related quality characteristics of an item overarching availability, reliability, recoverability, maintainability, and maintenance support performance'. In some cases, it may also include durability, safety and security. Similar concepts include trustworthiness, non-stop availability, and robustness.

In case of autonomous systems, for instance, their performance needs to be always delivered under any circumstances. High dependability can be supported by resilience against external turbulences and internal degradation. Intelligence contributes to dependability through self-monitoring, self-diagnosis, or even self-repair. An autonomous system has potential to increase dependability with the possibility of removing manual maintenance. In an extreme case, autonomous systems might even benefit from being subject to external stressors, as they can become antifragile [112].

Recently, safety and security are considered indispensable elements of dependability [152,189,212]. This is particularly true in the automotive sector in which functional safety of embedded software need to be certified [102].

2.2.8. Customization/individualization/personalization

Customisation, individualisation, or personalisation is key to more value to the customer. In the engineering design research community, 'mass customisation' [52,239,295] was first discussed in the context of modularisation, product family, and product platform [105,163,214]. Then, the concept of personalisation that satisfies implicit personal wants and needs was introduced as opposed to customisation that satisfies explicitly articulated requirements [125,240]. The advancement of additive manufacturing is facilitating the creation of really personalised products (e.g., [96,111,199,229,288]).

In the context of smart products, customisation, individualisation or personalisation to meet explicit or implicit user requirements could be supported by reconfigurability. We will examine this issue in Section 2.2.10.

2.2.9. Resilience

Resilience is originally a term in psychology. In the engineering context, it might be defined as the ability of a system to absorb external impacts without affecting its structure or behaviour [156]. This means in order for a system to exhibit resilience, it needs to have a special mechanism to absorb and counteract to the external impacts. The mechanism can be 'reconfiguration' that takes place at behavioural and structural levels discussed below [235].

2.2.10. Reconfiguration

The scale of reconfiguration can range from very minor adaptation to major changeover. The minor 'adaptation' or 'adjustment' include control responses to external stimuli and more major model tuning within the controller [184]. The major changeover can be morphological and topological changes or replacing units [155]. Reconfiguration, being minor or major ones, takes place at different levels of the system, i.e., structure, behaviour, and function [7,8,31,195,196]. An example of reconfigurability is modular architecture, which is supported by module-wise interoperability and standard interface between modules.

Reconfiguration can take place at any moment over the life time of smart products [258]. For instance, design-time reconfiguration is the modification of design to adapt an old model to new user requirements, while run-time reconfiguration can be any form of control to adapt the machine to changing external environments and deteriorating internal conditions. Life-time reconfiguration targets maximisation of the market value of the product by modifying part of the product over its life cycle or at the end of life. These three different modes of reconfiguration commonly use information obtained from sensors. While run-time reconfiguration is supposed to take place automatically or autonomously, the other two reconfiguration modes are not.

However, physical (or structural) reconfigurability is difficult to exercise (particularly, during run-time) for many product categories unless they have physically interchangeable modular structure

[7]. For this reason, software-based behavioural or functional reconfiguration is preferred for several smart products (e.g., smart phones). Therefore, dynamic real-time reconfiguration of smart products can be offered and implemented by using different software services [4,5,6]. Fig. 6 depicts life-time reconfiguration of a smart vehicle that goes through static reconfiguration in the middle of its life and later dynamic real-time reconfiguration. Static reconfiguration means non-personalized reconfiguration for all instances regardless of the environment in which these smart products exist, while dynamic reconfiguration is highly personalized based on the environmental data and thus instance-based [8,196].

Within manufacturing systems design, reconfigurable manufacturing systems have been well studied [64,122,130,155,267], Another good example of reconfigurability is modular reconfigurable robots [168,291,292]. Swarm robots are a group of robots not necessarily physically connected but loosely coupled with communication that performs tasks as a whole [15,155,160,274]. For instance, a fleet of autonomous drones exhibit useful group behaviours. Cellular machines are composed of homogeneous cells that exhibit simple functions, but as a whole, useful functions emerge (i.e., storage and transportation of packets) [120,193]. More recently, Amazon is operating warehouses with robots based on similar principles [87].

Reconfiguration is useful for the purpose of maintenance. Because of external disturbance or internal component deterioration, system's functions fail. By reconfiguring at parametric, state, or behavioural level, the system's function can be maintained. Table 2 compares various types of maintenance strategies through reconfiguration using the framework of the FBS (Function-Behaviour-State) modelling [246]. The top level is function which is an abstraction of a behaviour. A behaviour is a temporal sequence of state transitions. A state is a collection of parameters at a time containing. It is therefore a snapshot of the structure, which is a network of parameters.

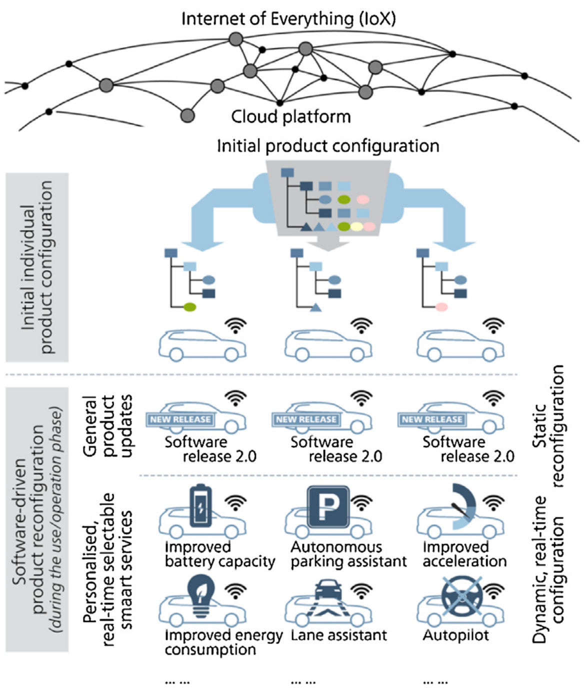

Fig. 6. Reconfiguration of smart products.

Table 2
Smart product reconfiguration during their use/operation phase.

FBS Modelling	Function-Redundant-type Self-Maintenance	Control-type Self-Maintenance	Adjustment through Control	Network Redundancy	Change Components
Function Abstraction	Maintain Function	Maintain Function	Maintain Function	Maintain Function	Maintain Function
Behaviours Temporal Transitions	Different Behaviours	Different Behaviours	Maintain Behaviours	Different Behaviour	Maintain Behaviours
States Snapshot at a Time	Different States	Different States	Different States	Different States	Maintain States
Parameter Network Topology	Different Topology	Same Topology	Tuning Parameters Values	Different Topology	Maintain Structure

The simplest maintenance is to diagnose the failure and change the failed component. This component changing strategy, however, does not change, other than the broken component, any of function, behaviour, state, and structure (represented by the parameter network topology).

Network-type redundancy is often used in transportation networks (railway, highway, etc.), infrastructure network (e.g., electricity distribution grids) and communication network (e.g., telephone networks and internet networks). In these networks, when one path fails, an alternative route is configured quickly, which results in a different topology. Although the capacity of this alternative route might not be the same as the broken one, the total throughput might be almost at the same level.

Parameter adjustment strategy is the control engineering approach. For example, set point adjustment will not change the parameter network topology but parameter values can be different. However, the goal of this strategy is to maintain the behaviours thereby maintaining the function. This can be contrasted with the control-type self-maintenance strategy [207], which is almost the same as set-point adjustment but when a set point exceeds, e.g., the physical limit, this strategy chooses a different control model. This means the physical parameter network does not change but the states and behaviours could be different. However, the total goal is to maintain the function.

The last function-redundant-type self-maintenance strategy will utilise a latent function of components existing in the system [247]. Therefore, the only thing to be maintained is the top-level function and anything else can be different.

2.3. *Contributions and challenges of smart products*

2.3.1. *Sustainability*

As discussed already, by monitoring status and environment parameters and later by processing the collected data using data analytics (see Section 3.3), smart products have a potential of better performances in maintainability, end-of-life treatment, and energy consumption, thereby contributing to sustainability [131,209,224]. For example, energy consumption might be reduced by understanding the machine condition better. Better capabilities of predicting failures early will prevent unnecessary stoppage of the operation, thereby guaranteeing the best use of resources as well as least waste. Understanding the usage over the whole life will help better decisions for the end-of-life treatment, thereby achieving circular economy [177,241].

2.3.2. *Multi-disciplinarity*

Smart products are mostly CPSs that are an outcome of an integration process over a variety of disciplines. They also interact with the physical world through sensors and actuators in a variety of ways. This physical reality brings up multi-disciplinarity, too. Multi-disciplinarity on one hand improves the function of the product significantly. On the other hand, it will add complexity [65,233]. The trade-off between functional advantages and complexity becomes a challenge.

2.3.3. *System of systems*

Large scale CPSs, in particular, are by definition a system of systems [150] not only for the number of subsystems but also for the multi-disciplinarity [233]. Being a system of systems means simultaneous co-existing events, interdependent behaviours, and consequently emergent functions. All of these add complexity at phenomenological, behavioural and functional levels.

2.3.4. *Complexity*

There are several sources of complexity of smart products. The first area to look into is software complexity. Smart products are CPSs controlled by software. As seen in Section 2.1.3, the software size has been increasing drastically recently due to ever increasing requirements. Not only size but also qualitative complexity of software architecture is also increasing [255].

Second, the process complexity of the product development processes is also very high. There are a couple of reasons. Capturing user requirements is one problematic area, as incorrectly captured requirements will lead to ineffective or inefficient product development processes. Related to this, the fact that the product development needs to consider the entire ecosystem (Fig. 2) significantly increases the complexity of product development. In addition, process complexity increases due to the multi-disciplinarity nature of smart products and the system of systems nature.

2.3.5. Cyber security and data protection

Just like any other ICT systems, cyber security has become one of the most concerning issues for smart products [44,47,186,218]. As seen in the famous Stuxnet case [200], cyber attacks through vulnerabilities of the system (even if it is a closed one) are a huge risk of a catastrophic or even life-threatening incident.

However, smart products as embedded systems have difficulty in dealing with cyber security. Traditionally, these systems were considered relatively safer compared with PCs connected that need consider cyber attacks, browser vulnerabilities, malwares, e-mail attachments with viruses, and infected USB devices. As they were considered isolated from the network, there was no protection. However, as smart products are indeed connected, it is now urgently necessary for them to be equipped with appropriate protection mechanisms. It is known that through the keyless entry system, vehicles can be stolen.

A related issue is data protection. Commercially sensitive data and information such as CAD data of a new product needs to be protected. However, PLMs are not different from any other systems. In addition, privacy data needs attention.

There is an EU regulation, General Data Protection Regulation (GDPR), for data protection and privacy [70] and similar measures are discussed internationally. In the case of commercially sensitive data, particularly, the ownership of collected data through smart products can become an issue. Suppose if a machine tool manufacturer collects data about the operation of a machine tool for monitoring and maintenance purposes. The problem is that such data can also be business sensitive and needs protection.

3. Enabling technologies for smart products

3.1. Contributions and challenges of smart products

There are emerging technologies that can underpin the smart product development. While many of these enabling technologies relevant to smart products are advanced ICT technologies, it is interesting to observe that advances in hardware- or material-oriented technologies are also enriching the concept of smart products. To name a few, the following can be suggested.

- Additive manufacturing [96,111,199,229,288] facilitates truly personalised production. There is even a possibility of simultaneous, integrated printing of both mechanical and electrical components. This will change the design and production of smart products significantly.
- Microelectronics is rapidly advancing: Semiconductor devices (CPU, memories, and sensors) are still making progress in performance, size, energy consumption and cost, which means the performance of mobile smart products in particular can improve. Foldable printed circuit boards and displays are already available, which allow denser implementation of the system.
- Battery technologies are also making progress. It is expected that soon the current lithium ion and lithium polymer batteries might be replaced with all solid-state batteries with much higher energy density and better safety features [183]. Flexible batteries are becoming possible, too [300]. Non-contact charging has been commercially available.
- New materials are emerging on the horizon, such as Cellulose Nano Fiber (CNF) with outstanding strength and light weight [63], 3D printable energy harvesting material [205], self-repair or self-healing metals, composites, polymer and ceramic that can significantly reduce life cycle cost [30,32].

The following ICT technologies are leading the way to so-called "Internet of Everything". These technical enablers are not necessarily innovatively disruptive any longer (some are indeed very classic) but they are technically mature enough to be integrated in a CPS product together with other technologies to yield a smart product.

- ◦ Communication technologies
- 5G/6G mobile networks
- NFC (Near Field Communication for contactless cards and RFID)
- Bluetooth Low Energy (BLE)
- M2M (Machine to Machine) networks
  - Internet technologies
- IPv6
- Social media
- Semantic web
  - Software technology
- AI technology such as machine learning and DNN (see Section 3.2)
- Big data and data analytics (see Section 3.3)
- NoSQL databases (to be used for big data)

These technologies together with AI technologies (Section 3.2) and data analytics (Section 3.3) form the core technologies to be embedded in smart products. Section 4 will look into technologies for the development of smart products. However, unfortunately, we do not have a silver bullet to tackle the challenges discussed in Section 2.3.

3.2. Artificial intelligence (AI)

Ever since IBM's Watson defeated human contestants in a TV quiz show 'Jeopardy!' in 2011, so-called deep learning-based AI has been recognised a breakthrough (IBM prefers to call Watson Augmented Intelligence, though) [73]. The role and value of classic symbolic logic-based methods and other types of reasoning techniques (such as fuzzy logic and genetic algorithm) [191] have become relatively less valued, primarily because the performance of deep learning-based methods was drastically superior to those classic ones. The fundamental difference between conventional ANN (Artificial Neural Network) and DNN (Deep Neural Network) is the number of the hidden (middle) layer(s). While ANN uses only one or a few layers, DNN (and its variants, such as Convolutional Neural Network, CNN) employs multiple (sometimes more than one hundred) layers.

In a broad sense, the learning (or training) algorithms for DNN are part of machine learning algorithms [88]. However, for machine learning, it was always 'teaching' meaning by providing data; a teacher needs to teach the answer. In some applications (e.g., image recognition) [124], DNN (e.g., Convolute Neural Network) does not require teaching provided with a sufficiently large amount of tagged image data. From a viewpoint of practical applications, this feature of learning without teaching makes a big difference and has become the biggest catch.

DNN exhibits outstanding performance, in particular, in pattern recognition, image recognition, and natural language processing. Other types of applications are now undertaken, too. However, obviously, a practical AI system cannot be built using solely deep learning methods. Many other classic reasoning methods are still valid and indispensable. If there is not much data available to train a DNN, it would be necessary to train the network through teaching, which becomes a tedious task. There is a trade-off between teaching time and effort against performance (e.g., output quality) of the system. Additionally, in principle DNN is a variant of ANN, therefore it is not possible to give explanations of the reasons of the reasoning outcomes. These are drawbacks of DNN.

While applications of classic AI techniques to product development are plenty within the research community, there are not many applications of deep learning (yet) in product development. One of the possible reasons for this is that product developers aim to maintain an overview of the network of interdependent and non-deterministic decision-making processes in development cycles. With an increasing number of decisions that cannot be underpinned – as is the case with deep learning – developers may feel less confident in governing the design process as the rationale is increasingly lacking.

However, as obviously seen in robotics and autonomous vehicles, these techniques are used to develop intelligent features of smart products. Within the CIRP community, deep learning techniques have been applied to the following design and manufacturing research topics; cutting process monitoring [78], cutting process planning [79], process planning for polishing [265], robotics [140,266], feature extraction for inspection [275], optimisation for process planning [180], planning for machine allocation [188], operation scheduling [271], diagnostics [278,296], planning and management of energy consumption [46], and design [268].

3.3. Big data and data analytics

The advances of computing technologies, especially of processor speed and storage capacity, combined with cheap communication means made the concept of 'big data' [18,154] available in many application areas. For instance, financial transaction data as big data allows for the extraction of customer purchasing preferences useful for future marketing. Similarly, smart products can function as a distributed sensor network that creates big data about product usage and users' behaviours [297]. Its example is the location information from smart phones used to create traffic information that will be used for navigation purposes (see Section 1).

Data analytics is a set of mathematical methods to model, capture, cleanse, analyse, and visualise big data. Among these, data mining is a set of statistical methods (such as principal component analysis and regression analysis) to model and discover knowledge to be used for predictive analysis. As illustrated in Figs. 2 and 3, smart products will be embedded in a data-centric ecosystem that will collect a huge amount of operational and life cycle related data. The collected data will be analysed with data analytics methods for technical and business purposes.

4. Development methods and tools for smart products

4.1. Currently available methods and tools

For the development of complex mechatronic products, a huge number of methods and supporting tools with a different level of maturity are available [234]. The majority of these methods and tools listed in Table 3 could also be used for the development of smart products. However, due to the high complexity described in Section 2.3.4, and due to the special characteristics of smart products such as intelligence, existing methods and tools are not powerful enough for the development [147].

Smart products have been discussed from a variety of aspects including product development methods (e.g., [12,76,108,142,223]), marketing (e.g., [57,110,133,202,250], and CPS based smart products [49,94,158]). Design methods and methodologies targeting a specific feature of smart products, for instance, 'design for reconfiguration', is useful to develop a smart product that exhibits dynamic reconfigurability during use phase (but unfortunately, this has not been established yet). Fuzzy front-end methods address early phases of product development including marketing [33], mapping business strategies of an enterprise to a business ecosystem (Fig. 2) and value generation processes (Fig. 3) offered by the smart product. This needs to end up with function, behaviour, and the physical embodiment of the smart product. Therefore, product development methods and service development methods identified in Table 3 cannot be separately discussed. The multi-disciplinary nature of CPS adds extra complexity to the smart product development process. Tools and methods of Model-Based Systems Engineering (MBSE) [84,273] are utilised for this purpose. We will examine MBSE in Section 4.4 in greater detail.

Table 3
Commonly used developing methods and tools.

Procedural approaches and generic methods
• Systems Engineering / MBSE (Model-based Systems Engineering). V-Model
• Agile methods adopted from the software development,
• Modular Design
• Collaborative Engineering, Lean Development, Concurrent Engineering, Simultaneous Engineering
• Design for X
System design
• Requirement engineering
• Functional modelling
• Behaviour modelling
• Product architecture design
• FMEA (Failure Mode and Effect Analysis)
• QFD
• Conjoint Analysis, Kano method
Domain-specific design tools and methods of components
• Mechanical CAD / rapid prototyping
• Electrical CAD
• Software Engineering (CASE)
• Service Engineering, PSS Engineering
• Simulation methods and tools (CAE)
System integration, verification and validation
• Multi-physics simulation
• Software in the loop, Hardware in the loop, Human in the loop
Product data and process management
• Product Lifecycle Management (PLM)
• Application Lifecycle Management (ALM)
• IoT integration platform
• Enterprise Resource Planning (ERP)

Accordingly, it is expected that the way product development of smart products is undertaken can be drastically different from the current practices from very early stages. Tools and methods for smart product development can also be different. For example, identifying user requirements for traditional products is depending on user research and market research comprising of a variety of methods that collects data actively. However, with smart products, market data can be collected 'actively' while collecting use data and operational data. This 'continuous real-time feedback' mechanism will make requirements definition phase totally different [9,10,11,13,14,54,182].

For example, a smart car equipped with massively deployed sensors can collect detailed information about how the car is operated. The information about the status of the car can be useful for the owner (who can be different from the driver) to perform predictive maintenance and to contract favourable insurance, for the manufacturer to improve the car design, and for public road planner to plan smoother city traffic. The car can also autonomously drive collaborating with other cars to offer a new type of mobility service and to increase the safety on the road. This will also minimise the congestion, thereby contributing to sustainability. This data collection requires careful design of sensors, data collection methods, data mining methods, as well as data protection strategies, which means seamless coordination between hardware design and cyber system design, beyond the traditional control engineering. In addition, obviously (at least part of, if not all) the collected data needs to be stored and properly managed. To this end, the concept of 'digital twin' is actually standing on the meeting point of these two directions [85] is becoming important. This issue will be examined in Section 4.3.

Similarly, resilience will increase dependability through reconfiguration. Resilient architecture contrasts to 'fixed' or 'rigid' architecture in which once designed, attributes, structure, behaviour and function will not change. Resilient architecture allows autonomous reconfiguration and changes, so that the system fits better to the environment or copes with its deterioration. This can be regarded as even growth or antifragile behaviour [112]. In this sense, product design for smart products is not only about the product itself but also about design of the mechanism for the product to change or grow. However, in existing literature, such 'growing architecture' is not discussed fully.

4.2. Value generation through services

Value generation through features of smart products as depicted in Fig. 3 will impact the services that smart products offer [20,58,167]. For example, a 'smart kettle' not only offers useful functions (e.g., automatic boiling of water) but also new service in the form of reporting abnormal behaviour by elderly users. This means smart products should take into consideration new PSS (Product-Service System) development, value chain, and business models. The smart kettle was developed by Zojirushi, Co., in Japan and has been in market since 2001 [301]. Due to the age, sudden illness, or symptoms of dementia, it is often necessary for family members to keep an eye on elderly people living alone. A direct solution is to set up a CCTV camera, but this is often problematic because it gives really an impression of violating privacy. As a resolution, the kettle not only boils water but also reports water boiling activities through mobile telephone connection. The usage information is sent from the server to a familiar member regularly, so that if there is anything wrong with the elderly person, it can be detected earlier. Similar products are now in the market, including 'smart gas meter', 'smart photo frame', and 'smart air-conditioner'. All of these sense human activities, although the responsibility of detecting abnormality is in the hands of family members.

In the case of a smart kettle, obviously not only the owner (elderly user living alone) but also additional stakeholders (who might be relatives, friends or neighbours) have to be considered in the value chain. The business model is different from that of traditional kettles, because of a new service element which is charged. However, value creation from available information is insufficient in the current model, because the system only reports usage data of the kettle. The system does not warn for unusual usages, lack of regular usage, or any other irregular activities. This is where perhaps some AI technologies can be introduced to generate more value for the user as well as for the business.

The value generation mechanism in Fig. 3 assumes a stronger role of service elements than of products themselves. The concept of XaaS (Everything as a Service) is a concept originally developed in cloud computing [277] that offers well-defined software components over the Internet. An example of XaaS is MaaS of smart vehicles as shown in Fig. 4. XaaS is also relevant here, because functions and services of smart products can be unitised for composition towards an entire solution. This compositional approach has a long history in software and Web services as SOA (Service Oriented Architecture) [67], but this can be analogically compared with PSS as below and potentially useful for PSS design.

- SaaS (Software as a Service) = Functional products
- PaaS (Platform as a Service) = PSS
- IaaS (Infrastructure as a Service) = Service infrastructure

In offering services, information flow plays a critical role. For example, MaaS requires such information flows as transportation requests, traffic information, vehicle information, and user information. Without these types of information, MaaS does not exist. This suggests that future business ecosystems cannot be designed without such concepts as XaaS and information flows associated with this XaaS.

One futuristic way of thinking about XaaS and future services for manufacturing is Supply Chain as a Service (SCaaS). Traditionally, within a supply chain for a manufacturer, logistics companies were considered only as a means for transportation and storage. However, if a logistics company like Amazon extends its services to 'provision of components', supply chains may disappear under the paradigm of cloud manufacturing [139].

4.3. Digital engineering

The notion of 'digital engineering' or 'virtual engineering' targets all engineering activities and processes from design, engineering, production, all through product life cycle, virtually performed on computer-supported systems, thereby reducing necessity of physical prototyping and testing [253]. While 'digital manufacturing' [50] refers to manufacturing processes, digital engineering may have an emphasis on narrower fields of processes and tools for design and engineering. However, e.g., it was demonstrated that digital engineering and digital manufacturing could be seamlessly integrated with help of the digital twin [222]. Digital engineering covers traditional mechanical engineering design support tools, i.e., CAD (Computer-Aided Design), CAM (Computer-Aided Manufacturing), CAE (Computer-Aided Engineering), CAPP (Computer-Aided Process Planning), manufacturing process simulation systems, and even systems engineering tools (e.g., Matlab/Simulink and Modelica). These tools are now integrated on a PLM (Product Lifecycle Management) platform for this purpose [219].

PLM is expanding its coverage from purely mechanical engineering and production engineering domains to electrical engineering and systems engineering domains. This contrasts with existing CAD tools in electrical engineering and electronics (e.g., those for circuit design) that seem to tend to stay within their own field, because mechanical CAD deals with physical shapes that are not necessarily limited to mechanical components but also relevant to the entire system for physical integration.

Manufacturing information frameworks, such as ERP [161], also integrate all necessary information about manufacturing processes and operations and used to control these. ERPs can be connected even to MES (Manufacturing Execution Systems) and even SCADA (Supervisory Control And Data Acquisition) for process control.

The integration of digital engineering tools can be seen from two perspectives. Tool-wise, the degree of PLM-based integration of all the design and engineering tools for smart products is not yet complete, although this should be the direction [121]. Data-wise, a wide range of data can be collected by ERP (rather than PLM) throughout the product life cycle, beginning with design and production preparation down to production, operation, maintenance, and end-of-life. The newly emerging concept of 'digital twin' is actually standing on the meeting point of these two directions [6,40,85,197,216,225,244,254]. A digital twin is a digital representation of an active unique product (real device, object, machine, service or intangible asset) or unique product service system (a system consisting of a product and a related service) that comprises its selected characteristics, properties, conditions and behaviours by means of models, information and data within a single or even across multiple life cycle phases [221]. It captures design information (i.e., to-be information), manufacturing information and operational information, which represents even minor differences of individual product (i.e., as-is information), thereby contributing to different types of decision making such as maintenance and end-of-life decisions.

Smart factories within Industry 4.0 will be inevitably software intensive. The software ranges from PLM (Product Lifecycle Modelling), ERP (Enterprise Resource Planning), MES (Manufacturing Execution Systems), SCADA, PLC (Programmable Logic Control) to servo controllers of individual equipment and devices on the shop floor. Among others, the increased number of sensors due to the IoT philosophy and finer control of processes will affect the complexity of software [248,259,263]. The nature of CPS adds another dimension of complexity. To deal with this, it is critical to establish a reference model of architecture of smart factory (e.g., see Ref. [185]) and to develop software using Model-Based Systems Engineering approaches (MBSE) [84,273].

The existence of vast use data will have very strong implications for PLM (Product Lifecycle Management) [2], ERP (Enterprise Resource Planning) [161] and other data management systems for product development. Traditionally, these tools were meant to be used during the design and production phases. However, the concept of digital twin [6,85] aims at storing vast operational and life cycle related data collected by smart products using the IoT technologies. Digital twins can be established on these PLM or ERP platforms, which extends the use of PLM and ERP across the entire product life cycle and is expected to serve as the basis for digital manufacturing [50] (see Fig. 7) in which smart products are expected to function as a mechanism to deliver smart services and collect massive life cycle related data. The roles and future research trends of digital twin in the context of digital manufacturing will be discussed further in Section 6.3.

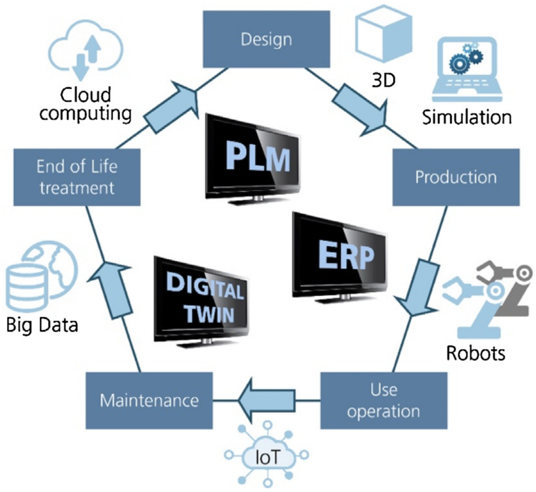

Fig. 7. Digital manufacturing and product life cycle.

4.4. Product development and systems engineering

Modern product development methods for multi-disciplinary mechatronics or CPS engineering have been discussed by a number of authors [22,45,93,94,118,119,141,165,179,204,253,298]. Some of them are influenced by traditional design process models [234], whereas others are based on systems engineering methodologies, such as the V-model [77] (Fig. 8). In Fig. 8, the left-hand side is the requirements (predicted models) whereas the right-hand side depicts actual implementation. It must be noted that the V-model is criticised for the discrepancies with the reality (e.g., [272]) and, therefore, V-model should be treated as a reference model rather than as a descriptive model.

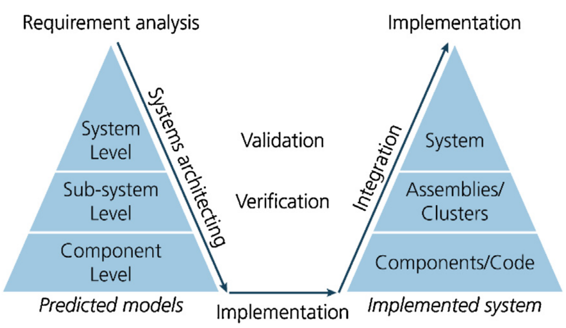

Fig. 8. V-model with predicted models and the implanted system.

The features listed in Section 2.2 can potentially influence the product development process, because these features rely heavily on software components. Ideally, the machine (hardware), electronics hardware, and the embedded control software should be developed as a whole as illustrated in Fig. 9. For instance, imagine a packaging machine. Fig. 9 reads that the first requirements were specified (e.g., packaging process description), which then would be transferred to a choice of the mechanism and a motion path of the mechanism (behavioural description). This would have to finally be converted to data for part making as well as a motor controller programme of the mechanism.

Fig. 9. Idealised product development process.

However, in practice, it is often the case that this ideal process in Fig. 9 splits in domain-wise (or discipline-wise) separated processes. There is a tendency that the control software development process comes last [22]. This means a realistic multi-disciplinary product development process would look like Fig. 10. After the initial product definition phase where functional requirements are determined, the mechanical and electronic design teams work almost independently until the design freeze.

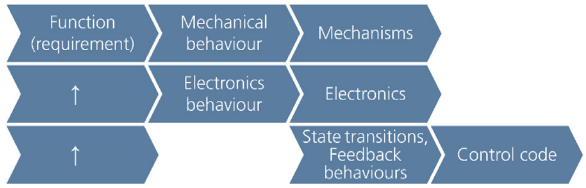

Fig. 10. Realistic multi-disciplinary product development process.

Then, the integration phase starts and in parallel the control software development follows up the mechanical and electronic design. This means the software development team waits for the mechanical and electronics teams to finish at least behavioural design. This also suggests that the product development will be undertaken as collaboration of mono-disciplinary teams [232]. In addition, practically speaking, the software development phase tends to delay, as this is the integration and verification/validation phases in which the only way to solve problems is code but not going back to hardware design. This means delay in product release. Therefore, improving this stage (verification and validation) with software-based methods could be beneficial.

In the near future, ideally, once the requirements are described and the mechanism is chosen, the behaviour of the mechanism as a motion path should be calculated automatically. The degree of automation of electronics CAD or software development is more or less the same and the control code can be semi-automatically generated. Fig. 11 illustrates this situation in which a light blue box indicates a potentially automated process. Due to these automated processes, project delays could be minimised or avoided.

In the future, eventually the smart packaging machine would (semi-)automatically programme itself from the packaging process description. This means that product development becomes integration of smart parts and we do not have to ‘programme’ the mechanism and design intelligence or embedded intelligence takes over the role of the designer. Embedded intelligence indeed is

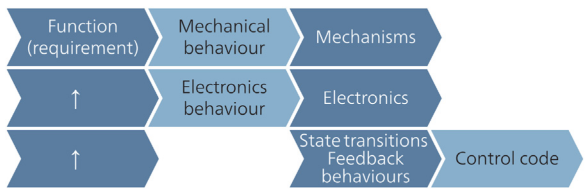

Fig. 11. Near future multi-disciplinary product development process.

a game changer in that the paradigm shifts from 'programmed behaviour' to 'learning for behaviour modification' (or growth). In addition, this also means a higher degree of discipline integration. Fig. 12 illustrates a future situation in which the three separated disciplines can be integrated with a help of high-level tools. Although this is just a vision at this moment, model-based systems engineering (see the next section) targets such automation. These research and development efforts in this direction will be looked at in Section 5.3.

Fig. 12. Future multi-disciplinary product development process.

4.5. Model-based systems engineering

Originating from software engineering, the concept of Model-Based (Systems) Engineering (MBSE) [84,273] (which was initially called Model-Driven Engineering (MDE) [198]) means the use of models to store, exchange, and develop information about the target software system being developed, rather than traditional document-based or code-based methods. The core philosophy of MBSE has a high degree of affinity with physical domains such as mechanical, mechatronic, and electronic engineering, because computer-based models are utilised more intensively than the software domain as observed in the rise of digital engineering (see Section 4.3).

Embedded software for smart products requires an appropriate level of abstraction about domain concepts to reduce the complexity of the domain, which was not sufficiently addressed in the traditional code-based development paradigm. For example, it often happens that requirements are recorded in a natural language in Microsoft Excel sheets. Instead, if requirements are recorded or illustrated in a graphical model like IDEF0 [98] or something that can be easily converted to a model in UML (Unified Modelling Language) [175] or SysML (Systems Modelling Language) [174], it would be possible to capture, describe, design, analyse and verify systems requirements, specifications, functions, and behaviours of the smart project in a formal manner. MBSE is expected to improve productivity of the code development process by facilitating reuse of models (i.e., eventually code). To develop smart products, MBSE is particularly suitable for controller design (e.g., [71,262]), as well as for programming the cyber part.

There are four significantly different pillars that form the basis of MBSE (Fig. 13). The first pillar is traditional systems engineering approach which is concerned about the development of large-scale complex systems. This approach contains rather top-down methods such as V-model as a prescriptive process model, formal definition decomposition methods, and specification methods including SysML and UML.

The second is the software engineering approach typically including CASE tools, some of which have links to descriptions in,

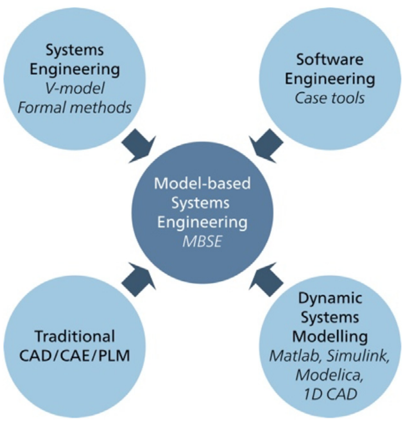

Fig. 13. Model-based systems engineering.

e.g., SysML, so that outcomes of systems engineering methods can be transferred into software codes.

The third is dynamic modelling of systems for which tools such as Matlab/Simulink and Modelica are gaining popularity to define, model and analyse systems dynamics behaviours. Recently, so-called 1D simulation or 1D CAE [211] is gaining interests, in particular, as a way to work with a simplified model which appears at very early phase of product development. Such a simplified model (often at physical phenomenon level) captures the essence of the function to be realised. Then, using a dynamic systems modelling tool, a conceptual level study (often with simplified differential equations) is conducted to determine key parameters (but very roughly). 1D modelling is often understood as a simplified method to speed up 2D or 3D CAE analysis processes but this is perhaps a too shallow interpretation. The true value of the 1D approaches is that they don't require too much details as is the case with 2D CAE or 3D CAE analysis.

Finally, the fourth is more recent recognition that physical models need to be incorporated for the smart products development. To do so, MBSE methods are being integrated into a PLM framework as illustrated in Section 5.3, which should allow easy transfer or conversion of information among models including even software code.

5. Research challenges for smart product development

5.1. State of the art of methods and tools for smart product development

To develop hardware, software, and contents of smart products, a wide variety of methods and IT tools (see Sections 4.2 and 4.3) is available. For example, we do know how to define and design 'connectedness' and 'sensor architecture' with systems engineering tools by representing information flow in SysML among various signal processing elements. However, we do not have generalised knowledge to build a 'reconfigurable system' or 'dependable machine', yet. Moreover, it is not obvious for any currently available model representation scheme within PLM or MBSE to represent semantic information needed to design services.

For the development of services, while there are a few methods available (e.g., [23,28,41,194]), there are not many strong IT tools available. Although the concepts of service CAD [23,117] have been proposed, they are not yet commonly used in industrial practice. In addition, the linkage between product design and service design is rather weak. Whereas general design methodologies to achieve a certain service goal in the context of technical architecture are yet to be fully understood and established, it is possible to identify attempts to develop a methodology that will represent service as a way to change the states of stakeholders [115,201,231]. In such a case, services are described as transitions among discrete events [117].

Smart products and their ecosystem generate information during their life cycle stages. This information should be processed to create higher level business value. To do so, techniques and tools of big data and data analytics can be used. However, feeding the obtained information back to smart product design is not yet practically exercised, whereas research is intensively conducted (e.g., [13]). At this moment, information obtained from product operations is mainly used for managerial and maintenance purposes [62,91,109,116,131,134,164].

The state of the art of methods and tools for the development of smart products could be summarised as follows.

- While there are still some deficiencies (e.g., immature integration of hardware and software design in MBSE, which will be discussed later), basic methods and IT tools for the development of smart products are reasonably mature and available.
- During the product development, the verification and validation subphases are always problematic. One possible solution is to

develop a software-based simulation tools for verification and validation. However, this is not a silver bullet, because the real root cause could be poor systems definition/decomposition, unpredicted phenomena [53,233], or impossibility of reversing design decisions when mechanical or electronics design is once frozen.
- Although methods for service design are available, IT tools for service design are undeveloped.
- There is no concrete method that links the smart product PSS design to user experiences [153].
- We do have a good amount of knowledge on how to deal with user experiences, but this is mainly focused on user interface design.
- Information generated from the smart product life cycle can be analysed adequately with a variety of methods and IT tools. But there are only a few formal methods to feed this information back to smart product development [9,10,11,13,14,54,182].

In the following sections, we look into some research challenges and current effort to tackle them.

5.2. MBSE challenges for smart product development

The use of MBSE to develop smart products is motivated in the first place by the complexity of smart products. The traditional document-based approach employs pure textual treatment of requirements, functional capabilities and behavioural properties in relation to the intended product and technical (sub-)system characteristics. Apparently, this approach cannot handle complex situations observed in smart product development, such as interactions with other products and devices on the network and completely amalgamated multi-disciplinarity.

Although MBSE offers many key solutions to the challenges above, there are still challenges to be solved. Table 4 compares different models in different domains relevant to smart products. From this table, it is clear that MBSE might be a promising approach, but there are still issues in model integration and data exchange among different models in different domains.

- Although it might be possible to represent functions in any domain using the “to do something” formalism or “input–output relationship” formalism, there is no trivial connection between functional models and other models at behaviour, state, or structure levels.
- Geometric model-based (i.e., PLM-based) integration of structure is possible for mechanical CAD and CAE applications. However, there are many other domain specific structural models that cannot be represented using only geometric models.
- Parameter value lists as well as state transitions using, e.g., IDEF, UML, or SysML, can be used to represent states in many domains.
- A variety of domain specific models exist for representing behaviours.
- There is no unified “MBSE model” for smart products, let alone service models. It is more pragmatic to develop translational approaches among these different types of models, such as ReqIF (Requirements Interchange Format) [173], system descriptions (SysML), interface standards for models (such as Functional Mockup Interface (FMI) [81]), and general linkages (such as OSLC (Open Services for Lifecycle Collaboration) [176]).
- This pragmatic translational approach (i.e., federation of mono-disciplinary models), as opposed to the ideal, unified, single model approach, seems valid, although obviously there are inevitable translation losses and inefficiencies.

Mosterman and Zander studied the architecture for Industry 4.0 from the viewpoint of software [166], which pointed out software challenges including:

- Emerging behaviour design: As a result of compositional approach, behaviours can emerge unexpectedly. How can these unpredicted behaviours be dealt with?
- Data sharing: Among models in different domains, the same data should be the same in a semantically correct manner, which is sometimes impossible.
- Functionality sharing: Control software needs to assume hardware limitations. This type of constraint needs to be correctly shared among different domains.
- Collaborative functionality testing: This is critically necessary for a dependable smart product, but such testing methods do not exist yet.

There are also incompatibilities between elements in the 1D CAE library (typically elements that describe differential equations for systems dynamics models) and functional decompositions necessary for systems definition. The latter does not consider time dependency which is critical in the 1D CAE library. This is because consistent theoretical foundation for MBSE is not yet fully developed. Instead, various *ad hoc* models are considered across various stages of the product development process.

With the introduction of services which are mostly defined by software elements, this incompatibility even exacerbates as services as software functions have nothing to do with physical functions. In addition, MBSE does not handle concepts like ‘adaptability’, or ‘customer experiences’ directly. These have to be first translated to parameters so that they can be modelled and manipulated with MBSE tools. This is perhaps extremely difficult if not impossible at all.

Therefore, MBSE for smart product development suffers from the lack of orchestrated models, if not a single unified model. This variety of concepts and model types is manifested, e.g., in different standards such as ReqIF, SysML, FMI, and, but all of them come with limited semantic expressive power. In addition, there is no formal standardised method to model and represent function models, system architecture, and information generated by the operation of the smart product.

If so many different types of models and standards are involved, traceability of data and decision making becomes extremely difficult to achieve due to organizational boundaries. In addition, due to a missing value model for data, information and digital models, it remains onerous to argue for investments in MBSE competence, education and model preparation.

INCOSE conducted a survey comparing various MBSE methods and tools including UML and SysML [68]. Comparing readiness for an integrated system approach, the concept of executable SysML was derived. In summary, the following limitations and shortcomings exist and need to be tackled for MBSE to become ready for the robust development of smart products.

Table 4
Examples of models in different domains.

	Behaviour	State	Structure
Mechanical	Motion diagram	3D geometric model Force diagram	3D geometric model
FEM analysis	Stress diagram	Force diagram	Mesh model generated from 3D geometric model
Electro-nic	State transitions	Parameter value list	Circuit diagram
Servo control	Parameter value diagram	Parameter value list	Block diagram
Soft-ware in general	State transitions	Parameter value list	• Data model • Flowchart

- There is no common understanding among practitioners about how MBSE works, for which purpose and in which cases its methods can and should be used. Most of them regard MBSE as a solution for current problems rather than a future direction. Many experts believe that there is a lack of maturity of the existing MBSE methodology. This can be interpreted in two ways; either the methodology really is incomplete, or the knowledge of practitioners is insufficient.
- As it is rather complicated to use many different digital models correctly, dedicated education and training are desirable. This should cover various PLM tools, from CAD (E-CAD, M-CAD), CAE (both 3D and 1D) and various specification models (such as SysML or ReqIF) as well as integration tools and standards (such as FMI or OSLC).
- Functional models and architecture models have not extensively been used in traditional engineering areas [236] although the use of functional models can be seen widely in precision engineering. This requires a new kind of system thinking away from typical embodiment design models. The anticipation of logical and physical dependencies out of those models remain unclear but offer significant opportunities to avoid late changes during the development process.
- The extensive tool offering makes it difficult to clarify the core capabilities that are needed from a skill-set and competence point of view.
- Today's implicit engineering collaboration (mostly based on e-mails and direct oral communication) needs motivation to engage a model-based approach. Cultural challenges and better team-oriented review mechanisms for model-based engineering are critical in moving forward. In addition, currently, development engineers tend not to share certain model knowledge with others.
- No powerful solutions exist yet in helping engineers to manage associated digital models without significant overhead. Therefore, it is a must to provide better and (partly) automated solutions for managing models. Engineers need a relief in orchestrating models in order to make an effective use of the new model richness for the anticipation of smart product behaviour and interactions.

In response to these limitations, efforts are being made, as depicted in the next section, to develop methods and toolsets that can address at least some of the issues addressed above. One of them is the architecture-centric approach (e.g., [232]). According to ISO/IEC 42010 [103], 'architecture is fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution'.

At this level of abstraction, e.g., behaviours will be represented by only parameters and their changes without any implementational level descriptions (i.e., mechanisms). This means architectural descriptions are domain-independent and units at architecture level will allow domain-independent compositional approach to design synthesis [71,118,119,215].

5.3. Industrial development of MBSE

In the following, we will look at two recent approaches developed by vendors of integrated tools (Dassault Systèmes and Siemens).

5.3.1. Dassault systèmes' approach to MBSE

The developer of the 3DExperience platform, Dassault Systèmes, published a white paper on the development of smart products [34], which advises steps towards successful smart product development.

- *Define the target*: Centralise and share information. Make cross-discipline requirements and design artefacts visible across the product development team to relevant stakeholders at each step of the development process.
- *Plan the product*: Establish the smart product conceptual model right as early as possible. Use solutions that support systems modelling and simulate systems behaviours in all operation modes.
- *Validate early and often*: Use early simulation at the product level with best visualization techniques. Facilitate informal and formal information sharing and interaction.
- *Design product architecture and validate to interfaces*: Use solutions that provide visibility and collaboration at both the domain level and across disciplines to eventually ease collaborative decision making based on shared KPIs.
- *Aggressively manage change*: change management must be effectively done in an integrated manner across disciplines. Extend change management scope beyond the current design. Use tools for traceability and change impact analysis.
- *Enable an integrated lifecycle view*: Use modern formal data/lifecycle management solutions as smart products demand more integration, transparency, and traceability.

A use case of UAV design on the 3DExperience demonstrated the power of the MBSE approach [42]. This used IEEE 1220 [99] to integrate requirements, functional, logical and physical views. Fig. 14 depicts the implementation of these views on the 3DExperience platform with the following features which helped a rather novice engineer to implement virtual flight simulation in the 3DVIA environment in a short period of time.

This demonstration achieved the following technical aspects.

- Traceable requirement management.
- Functional and logical design to define functional and components architectures with interfaces definition.
- Behaviour modelling to add dynamic and static models of various engineering fields: These models are described in Modelica or coming from external modelers (such as Simulia for CFD simulation and Dymola for dynamics simulation, Simulink or SW code for control) through FMI.
- Virtual physical prototypes to run experiences including 3D, model-in-loop, hardware-in-loop, and software-in-loop codes into in the verification and validation processes.
- Automated report generation. An automotive multi-disciplinary case study was conducted. This demonstrated that the combination of the 3DExperience platform and MagicDraw (No Magic Inc.) was useful for, especially, architectural level systems design [72].

- Virtual physical prototypes to run experiences including 3D, model-in-loop, hardware-in-loop, and software-in-loop codes into in the verification and validation processes.
- Automated report generation. An automotive multi-disciplinary case study was conducted. This demonstrated that the combination of the 3DExperience platform and MagicDraw (No Magic Inc.) was useful for, especially, architectural level systems design [72].

A comparative study compared design thinking [35] and complex systems engineering through experiments [61]. Design thinking focuses on creative aspects, while systems engineering focuses on vigorous modelling and precise simulation. It was proposed that a combinatory approach of design thinking and systems engineering was a desirable research direction.

5.3.2. Siemens approach to MBSE

Siemens develops and offers a comprehensive suite of PLM software, such as NX (CAD), Tecnomatix (manufacturing process planning and simulation), Teamcenter (PLM), Simcenter (CAE), Mentor (electronics CAD) and Mindsphere (IoT environment) [213], across the full product lifecycle targeting industrial companies in a variety of industries focussed on smart product development and commissioning. The Siemens PLM approach to MBSE is based on a digitalisation strategy beginning with concept design through multi-domain simulation and verification across

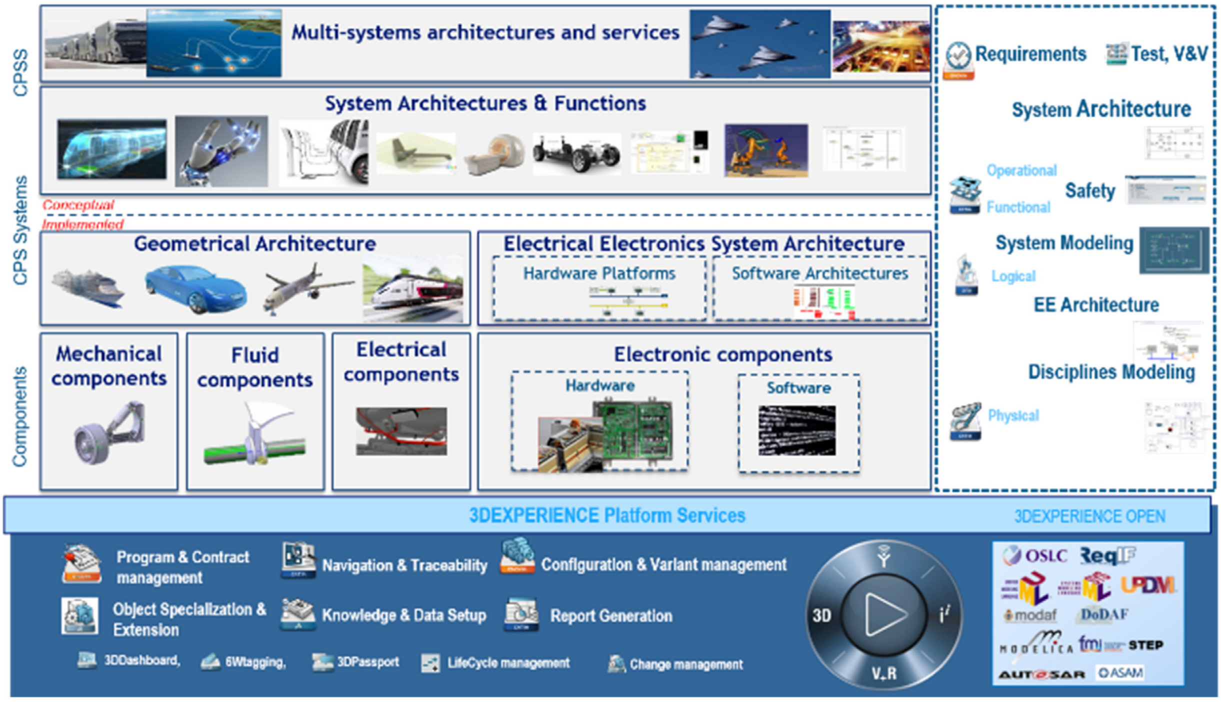

Fig. 14. The 3DEXPERIENCE Platform for Systems Engineering [42].

the full lifecycle of mechanical, electronics, and embedded software. Fig. 15 depicts the systems engineering methodology on the Siemens PLM platform. It is also based on the recognition that the digital twin as the virtual replica of the product can represent across a vast array of contributors. The digital twin is central to ensuring the product functions as intended and product behaviours can be verified to confirm performance, availability and reliability.

With regard to intelligence of smart products, such as Autonomous ADS (Automated Driving System) functions, the system definition is key to representing the scope of the ADS with respect to its functional requirements and relationships. The ability to model each aspect of the vehicle, its environment, its occupants, and the relationships is paramount in verifying if the resulting product can meet regulatory compliance while also delivering the consumer features expected. In Ref. [217], a virtual validation and testing framework is reported that is based on a co-simulation platform of vehicle dynamics and traffic environment tools. To validate algorithms, it is imperative to take into consideration both vehicle dynamics and the wide potential of traffic scenarios. By simulating the algorithms in a software environment, it was made possible to frontload the verification of control software in the absence of hardware.

In Ref. [257], optimisation design of powertrain architecture for hybrid vehicles was illustrated through a combination of automatically generated concepts with optimal control selection. This demonstrated the possibility of optimal design at the architecture level taking controls performance (e.g., energy management) into consideration. The increasing interest in hybrid electric vehicles imposes the need to provide computational support for design choices during the development of these vehicles.

Future challenges include the expansion of domains beyond mechanical, electrical, electronic, and software with classical MBSE methodologies, extending into smart service design (e.g., mobility, traffic control, etc.). To do so, work is underway with Mindsphere applications in collaboration with Siemens' Smart City Division

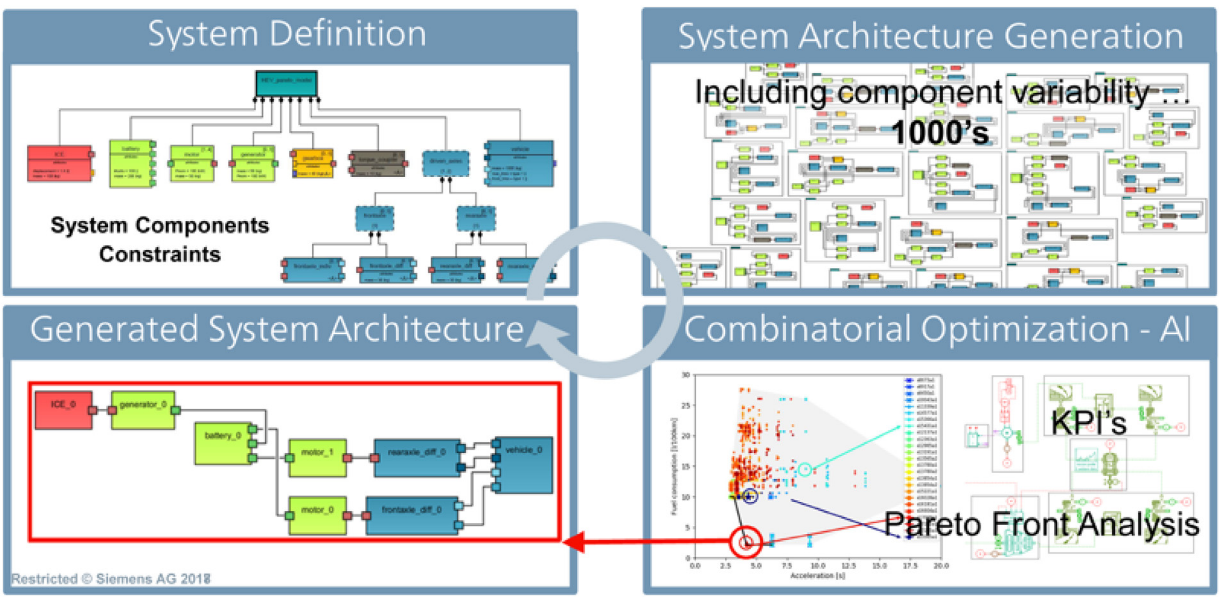

Fig. 15. Systems engineering methodology: Generative engineering to explore and optimize architectural combinations (courtesy of Siemens).

and eMobility inter-modal solutions to connect smart products to smart environments. It is also anticipated that the recent advances of IoT technologies will allow the collection of a massive amount of data at every stage of product life cycle, which will be stored in a digital twin creating a 'hybrid model' of products [254]. Fig. 16 depicts the digital twin concept used at Siemens, which covers the whole range of virtual product development processes.

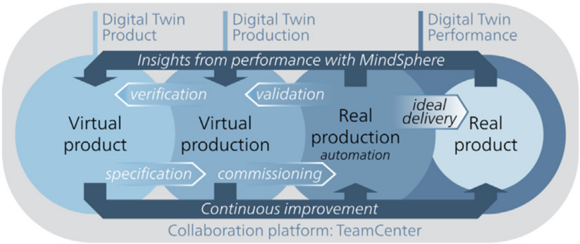

Fig. 16. Digital twin concept at Siemens.

6. Discussions and future research directions

6.1. Findings

This section discusses our findings about smart product development capabilities and future research directions. The paper began with Chapter 1 which defined the smart product concept (Fig. 1) and clarified the data-driven value generation mechanism of smart products (Fig. 3). In designing smart products, it is crucial to deal with experiences as well as information generated at every life cycle stage of smart products appropriately. Chapter 2 reviewed some existing smart products and summarised their functional capabilities and technical features of smart products (such as intelligence and resilience).

Chapter 3 overviewed technologies that need to be implemented in smart products, including AI technologies and data analytics. Chapter 4 discussed technologies to be used in developing smart products such as traditional product development methods, service development methods, systems engineering, digital engineering (including digital twin), systems engineering, and MBSE.

Regarding services, we seem to have sufficient knowledge to process information collected at various life cycle stages of smart products for the purposes of control and management of product life cycles. However, we have insufficient knowledge about how to convert that information useful in designing smart products and services. In addition to this, we do not have a formal methodology to develop smart products considering the entire scope of the ecosystem.

Chapter 5 focused on challenges of these developments, among others, of MBSE. Clearly methodologies to deal with services consistently with other domains are yet to be developed. In addition, it seems that multi-disciplinarity of models is best handled by developing translational standards.

In the following sections, we will discuss future research needs and trends identified in previous sections.

6.2. Survey on future research needs

Although innovation and business model potentials of smart products within Industry 4.0 are increasingly recognised by industrial companies and research institutes, only few research projects and activities address the development of smart products topics comprehensively. Two studies [4,5] in collaboration with about 100 international leading engineering experts have identified research deficiencies in the context of smart product development and defined following groups of research needs for the future:

- Procedural approaches and generic methods for smart product engineering.
- System design of smart products.
- Domain-specific design of smart product components.
- System integration, validation and verification of smart product components.
- Product data and process management.
- Smart product IT infrastructure.

In the first area of procedural approaches and generic methods, following main needs for research have been identified:

- Extension of the V-model of systems engineering to include service components as well as cooperative and agile work methods.
- Extension of the smart product development phase: During, e.g., the usage phase, some sort of design activities can take place for reconfiguration purposes. This means application of reconfiguration to all phases of the product lifecycle, in particular, to the usage phase (run-time reconfiguration and life-time reconfiguration).
- Establishing new disruptive approaches for the development of innovative smart products.

Regarding the system design of smart products, the following needs were identified:

- Development of 'design for smart X' in which X stands for a life cycle stage. This also includes IT tools to process feedback information from physical products [148].
- Consideration and modelling of business models, smart services, system interfaces and product cloud platforms for such business models during the smart product's system design phase.
- Use of feedback information from previous product generations in the system design stage.
- Extension of existing requirement engineering and management methods for the smart product design.
- Development of intuitive modelling and visualisation techniques for incomplete functional structures considering multi-layer abstractions.
- Development of models for a smart product's communicational, interactional and behavioural aspects.

Domain-specific design of product components lacks the following research topics:

- Designing smart human-machine interactions.
- Virtual functional simulation and validation of a component's behaviour including service components using multi-physics methods.
- Reflecting feedback information from downstream design phases during the development and optimization of individual product components, including new design for smart X methods and security requirements.
- Designing digital/virtual product twins, and of RAMI 4.0 administration shells [185] and product cloud platforms.

The main research needs for system integration, validation and verification of smart products are:

- Review and certification of a smart product with the objective to reduce the time required for validation and verification
- Field data and feedback information for validation purposes from the product's usage phase.
- Hybrid validation of real and virtual smart product components (hardware in the loop, software in the loop, human in the loop).
- Integration of customers and stakeholders into the product validation.

In the context of the product data and process management, the following research topics have been suggested:

- Integration of all virtual models (e.g. communicational, interactional and behavioural models) into PLM.

- Extension of existing product lifecycle management approaches with field data of real product instances.
- Instance-based management of virtual product models (management of the digital/virtual product twin).
- Integration of flexible and agile methods into engineering workflows.
- Integration of product reconfiguration concepts into the product lifecycle management.

Research needs regarding IT infrastructures are:

- Extension of existing neutral data formats (e.g. STEP and JT).
- Definition of comprehensive and application-specific, semantically rich data models (e.g., ontologies and semantic networks).
- Use of cloud platforms as partial smart product components as well as a part of engineering IT tools.
- Open software system architectures for engineering IT tools.

Additionally, engineering reference models and solution libraries for the development of smart products are yet to be addressed by scientific work. Furthermore, pervasively a high need for standardization in the context of smart product engineering and design has been identified.

6.3. Digital twin

In Section 4.3, it was argued that digital twin taking advantage of PLM should be combined with IoT in the future. Data is collected using the IoT technologies at every life cycle stage of smart products starting from development, production, logistics, operation, maintenance, and end-of-life. The data will be converted to more useful information using big data and AI technologies and should be used for other activities.

Fig. 17 illustrates the roles of digital twin in the context of digital engineering [145,146]. First, the smart product will be designed, and its product model will be stored in the PLM as a digital master. From the digital master, digital prototypes are generated for various types of evaluation through simulation (or analysis). The results of the evaluation will be reflected in design improvements. Smart products will be produced from the digital master, released, and operated as smart product service system. Its life cycle data will be collected and stored as an as-is model in the digital twin.

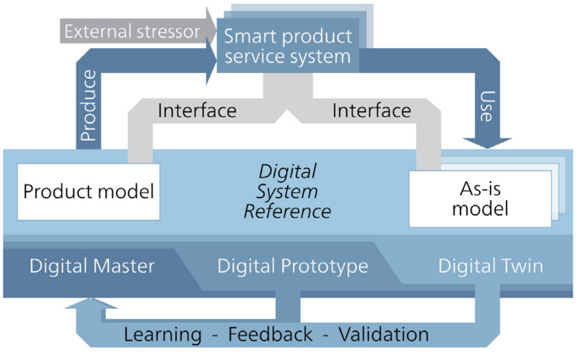

Fig. 17. Smart products and digital master, digital prototype and digital twin.

In Fig. 17, there is an interface between the product model and the smart product service system through which production takes place. The interface between the smart product service system and the as-is model is the IoT sensors and any other data collection mechanisms. The feedback mechanism, particularly the one from the as-is model to the product model, is not yet sufficiently understood, which is a future research topic. At the same time, the figure stresses the importance of the information backbone in development cycles as well as the necessity of 4D-based digital twin (i.e., including time scale) rather than only 3D-based digital twin.

6.4. Smart digital engineering

This section illustrates a new research topic of 'smart digital engineering', which is now tackled within one of the authors' laboratory [36,112,119,223]. Smart digital engineering represents technology and engineering capabilities required for smart product development and may include the following four types of Smart Digital Engineering Capabilities (SDEC) [223]. In each of these areas, possible research topics will be explained.

- Smart Digital Engineering
- Smart Virtual Engineering
- Smart Model-Based Systems Engineering
- Smart Lifecycle Engineering

6.4.1. Smart digital engineering capabilities

Digital engineering, in general, describes capabilities of creating digital product models, of handling digital models in the PLM in a consistent manner and of generating representations using the information sets within the PLM following digital workflows. This leads to the following definition.

SDEC will address new digital assistance and intelligence in creating, revising, exploring, collaborating, testing, signing-off and releasing diverse sets of traceable artefacts as part of digital engineering workflows, processes and activities.

From this definition, clearly digital engineering workflows, processes, and activities must be linked to behaviours of engineer (i.e., digital behaviour). Therefore, the first Smart Engineering Capabilities is:

- SDEC #1: *Identification of digital work behaviour* (type: digital)

Location within the EOS (Engineering Operating System): EOS is a system that controls and operates various types of a human engineer's cognitive activities during product development within the triangle of artefacts, processes, and (IT) tools [143,144] (see Fig. 18). A work behaviour of the engineer is located at the intersection among 'virtual and physical artefacts' (digital models and data), 'IT tools and systems', and 'process and organisations'.

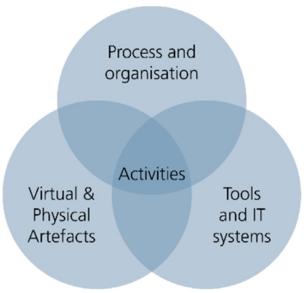

Fig. 18. Engineering Operating System (EOS) [144].

Problem description: Engineers and their management face problems to properly match the type of digital work with the digital computer application during typical engineering work patterns and associated collaboration scenarios.

Solution requirement description: In order to improve engineering activities patterns and styles with respect to quality, efficiency, relevance and compatibilities, it must be possible to seamlessly choose the best digital work behaviours for individuals and teams. Since the amount of digital work and its associated activities are steadily growing, optimised low-cost (time- and money-wise) IT solutions are needed, not only for individual and collaborative technical tasks but also administrative tasks.

Current research concepts and demonstrators: The influence of digitised work environments on engineering activities has been neglected so far, but it is necessary for the scientific research community to look into activities of information procurement, agreements and documentation in computer applications [143].

Potential first implementation experiences: Individual digital work behaviour patterns of engineers at a computer workstation can be capturable and related to specific development activities. This is achieved by automated identification and comprehensive analysis of such activities with respect to the development task, digital tools in interplay, work processes and digital models [144].

6.4.2. Smart virtual engineering capabilities

Virtual engineering addresses a range of scientific, technological, organizational and business activities using advanced information and communication technology and methods with a major focus on process and systems integration, immersive visualization and 'human-machine-human' interaction [95], potentially combined in so-called Synthetic Environments [55]. The overall aim of virtual engineering is the early, continuous, networked (process view) and integrated (system view) support of the entire product life cycle concerning collaboration, assessment, concretisation and validation of products and processes with participation of all partners using virtual artefacts.

- SDEC #2: *Smart hybrid prototyping* (type: virtual)

Location within the EOS: The idea of smart virtual engineering is validation as early as possible in the product development process ('process and organisation' in Fig. 18). For this purpose, the most suitable virtualization tools are selected in the 'tools and IT systems' area. Then, appropriate models are built in the virtual artefacts area to be validated with, e.g., user interfaces with VR technologies again in the 'tools and IT systems' area.

Problem description: The requirements for the new engineering capability of SHP (Smart Hybrid Prototype) is derived from higher-level objectives to fix industrial challenges of virtual engineering, e.g. the growing complexity of product and process requirements while drastically reducing time-to-market and classical physical product prototyping. Additionally, there is a need to 'physically' experience virtual models (e.g., CAD models). This can be achieved by using haptic devices, which enable force-feedback, thereby, interaction with virtual models.

Solution requirement description: The SHP approach is defined as a combination of physical prototypes and digital models in a virtual reality environment in order to enable a realistic experiencing of a mechatronic system [220]. Furthermore, it enables a multimodal experience of mechatronic products, which means that the human factor is given much greater prominence when validating the overall system in real-time.

Current research concepts and demonstrators: These include visualization technologies such as virtual, augmented and mixed reality, modelling and simulation technologies with physics-based game engines or professional tools such as MATLAB/Simulink or Dymola, interaction technologies and human machine interfaces (HMIs) like haptic interfaces and tangible user interfaces (TUIs).

Potential first implementation experiences: The SHP technology was validated by developing, assembling and testing a passenger car's tailgate device [25]. The tailgate test setup is shown in Fig. 19.

6.4.3. Smart (model-based) systems engineering capabilities

Systems engineering addresses issues such as requirements engineering, reliability, logistics, coordination of different teams, testing and evaluation, maintainability and many other disciplines necessary for successful system development, design, implementation, and ultimate decommission. Its scope needs to be expanded when dealing with large or complex projects. Systems engineering deals with work-processes, optimization methods, and risk management tools in such projects.

- SDEC #3: *Model intelligence for IoT and Industry 4.0* (type: model based)

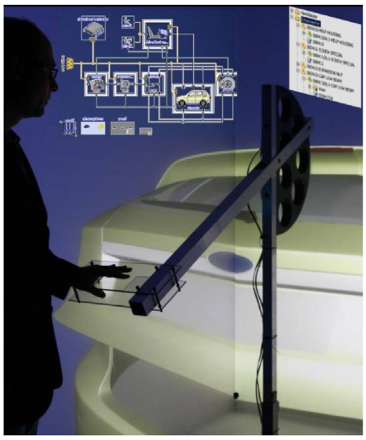

Fig. 19. SHP demonstrator [25].

Location within the EOS: From product development up to commissioning and final realization of the product/production system, each process step requires different types of digital models and their respective data. This means that MBSE sits in the 'activities' area in Fig. 17. MBSE operations need to be processed by IT tools and applications that allow lossless information transition and storage as well as accessibility during lifecycle phases, across engineering domains and across company borders. The new challenge here is to establish a meaningful feedback to design channel: information from product operation back into design and engineering phases and to provide services and capabilities to handle models and data appropriately.

Problem description: MBSE methodologies and tools are required because smart products are software intensive. This is even getting more so, because of the introduction of IoT technologies such as Industry 4.0 for smart factories. However, there are no commonly accepted and widespread technologies and formats of how to define interconnectivity and services of smart products during the product development phases.

Solution requirement description: MBSE calls for sophisticated multi-disciplinary models that integrate all domains of engineering. There can be two approaches to realise such an integrated model. One is to develop a truly integrated model that can cover many different disciplines, but this is impossible unless the level of descriptions is high-level without much details. One such a model could be the architectural level model discussed in Section 5.2. The other approach is to develop an integrated framework that can accommodate different types of models. PLM systems are typically based on this approach.

Current research concepts and demonstrators: An example demonstration cell 'Smart Factory 4.0' is being developed at Fraunhofer-IPK in Berlin, Germany. It is a miniature production line for conducting research and acts as a testbed for tool and component vendors. The Smart Factory itself can be considered smart to an extent that it manufactures individualized batch-size products right after the customer has entered his/her product definition into the system. The basis here is a set of simulation models that interact intelligently with each other and based upon sensor data created in the physical as well as in the virtual production line. Both worlds are inter-connected with IoT technology.

Potential first implementation experiences: The Fraunhofer-IPK implementation can be considered the first implementation of the kind.

6.4.4. Smart life cycle engineering capabilities

Smart products and services require new capabilities in connectivity of information and data sets across all lifecycle phases. Additionally, new IT platforms (e.g. IoT) and technologies (e.g. CPS) change common methodologies in engineering design, business development and lifecycle thinking. Some examples of this 'smart life cycle engineering' have been discussed in Section 2.2.3.

- SDEC #4: *Smart services* (type: life cycle)

Location within the EOS: Products are becoming smarter and services are offered across the entire life cycle. In the EOS in Fig. 17, the digital and physical artefacts are changing in the way they exist in the life cycle, which coincides with the evolution of the IT tools, applications and systems. This means that traditional engineering companies will have to adapt their engineering processes and organization. For instance, it is clear that, as the IoT technologies advance, products will be equipped with more sensors, which will generate more information of different types. These will force the IT infrastructure to change and eventually, which will change the business as well as products, which we have seen in the cases of Rolls-Royce [91] and Komatsu [109,116].

Problem description: Smart products and smart services require new engineering capabilities for development, manufacturing, use and end-of-life treatment. For these, dedicated data and connectivity management is required.

Solution requirement description: Solutions are needed for both engineering processes and life cycle processes. Harmonized interaction of artefacts requires synchronized data management and information standards up to business model including PDM/PLM and IoT platforms.

Current research concepts and demonstrators: In the future, data about business as well as consumer behaviours becomes the main resource for the data-driven business paradigm. Technically, this means big data technologies will play a key role here.

Potential first implementation experiences: Data driven engineering is the major challenge in smart life cycle engineering. Several data levels and data sets (e.g., sensor data, product system behaviour data) need to be combined, processed and analysed. Based on that, engineering conclusions can be drawn. Different heterogeneous data needs to be linked and be prepared for analysis. Consequently, semantic level data integration seems to be a key enabler for gaining serious results in the understanding of engineering.

7. Summary and conclusions

7.1. Summary

This paper attempted to shape the concept of smart products and to identify issues and challenges of their development from the viewpoint of engineering capabilities. Overall, the increasing population of smart products will drive significant changes to the conventional and to the current digital transformed engineering practice.

First, the concept of smart products was defined as CPS integrating internet-based services, which means smart products are software intensive, data driven, and service conscious. Smart products generate value in two ways (Fig. 3). One is through smart services that will increase the value of user experiences. The other is through operational data and information created at every life cycle stage. This has mostly business value.

In Chapter 2, various types of smart products were briefly reviewed, followed by identification and analysis of commonly found features of smart products.

Chapter 3 discussed product technologies that will be used or implemented within smart products. Among them, it is worthwhile to focus on AI, big data and data analytics.

Chapter 4 reviewed product development technologies (including those for service development). From recent development, the paper picked up some important technologies including service development methods, digital engineering, and model-based systems engineering (MBSE). While PLM and ERP are widely used in product development, it is important to note that data collected with IoT technologies during the life cycle of smart products needs to be stored and properly processed. For this purpose, the concept of digital twin seems promising. MBSE is useful to *co-develop* multi-disciplinary complex CPS, in particular, those software intensive smart products.

Chapter 5 discussed insufficiency and weak points of currently available development methods and tools. Among others, the following five weak points were identified.

- Many smart products specific aspects (such as resilience) lack design methods (i.e., design for resilience).
- Dealing with software intensive nature of smart products and resulting multi-disciplinarity, MBSE is considered promising from the viewpoint of *co-development,* but at this stage it still needs further development. Among others, at this moment, a single unified model about various aspects is not possible, which means conversion of models are left with *ad hoc* interfacing standards.
- Service design is not well connected to other elements of smart product development as PSS.
- Capturing and storing information generated at every life cycle stage of smart product are not that difficult. However, our knowledge on how to feed that information to design is still poor.

Chapter 6 first described the outcome of an industrial survey about smart product development. This largely matched our findings, especially regarding the immaturity of theoretical development, and the lack of unified theory integrated design. This chapter also proposed the concept of smart digital engineering.

7.2. Conclusions

The concept of smart products is now getting clearer, i.e., CPS with integrated internet-based services, which suggests software-intensiveness, data-driven, and multi-disciplinarity. However, the development of smart products is not just about introducing more software related capabilities. It is significantly different from the development of previous generations.

- Smart products are software intensive and the degree of multi-disciplinarity is much higher than, e.g., traditional mechatronics products, because the role of information is more than control. This suggests the architecture of smart products needs to be well established and clearly defined.
- The data-driven aspect of service elements coming from value creation through life cycle related information needs to be modelled, presented and utilised appropriately for smart product development.
- Information flow within the ecosystem of smart products is complicated and, consequently, the entire product development can be very complex.
- Decision-making processes during the product development become mutually dependent, thus controlling the product development cycle can be extremely difficult. This means the currently available prescriptive development models (e.g., V-model) can be inappropriate.
- The role of digital twins will be increasing and requires further research (e.g., to extend it to 4D models).
- MBSE seems a promising approach to tackle multi-disciplinary nature integrating four relevant approaches (i.e., traditional PLM, systems engineering, software engineering, and dynamic systems modelling). While this requires a unified single object model, this is currently replaced with a group of translational interface standards. Further research is needed to establish the sound foundation of MBSE for smart product development.
- From the viewpoint of education and training, solid competence in digital engineering, software engineering, AI algorithms, and MBSE will be needed for engineers and designers of smart products.

7.3. Future outlook

- It is not difficult to imagine that future smart products will be enjoying faster and better connectedness, intelligence, and sensors leading to step-changing improvements. For instance, far better vision capabilities will be installed in robots, thereby improving material handling capabilities. New applications of such robots will be more non-traditional situations like home, office, and society rather than factories.

- New capabilities that are not really present nowadays include autonomously reconfigurable products adapting to many unforeseen situations and extremely resilient products requiring no maintenance. For these products, validation and verification can be based on self-learning elements by adjusted algorithms.
- Research-wise, researchers and industry have to mobilize aligned efforts to sketch out a complete foundation consisting advanced theories about cross-domain models that can be used for future MBSE.

Acknowledgments

The authors would like to thank for the contributions and support from Dassault Systèmes, in particular, from Philippe Laufer, Eliane Fourgeau and Gauthier Fanmuy. They would like to acknowledge the contributions and support from Pete Lamoureux of Siemens AG.

References

[1] Abid A, Hammadi M, Choley JY, et al (2016) A Holonic-Based Method for Design Process of Cyber-Physical Reconfigurable Systems, 2016. *International Symposium on Systems Engineering IEEE*. http://dx.doi.org/10.1109/SysEng.2016.7753180.
[2] Abramovici M (2007) Future Trends in Product Lifecycle Management (PLM). *Proceedings of the 17th CIRP Design Conference* 665–674.
[3] Abramovici M (2015) *Smart Products, CIRP Encyclopedia of Production Engineering*, Springer. http://dx.doi.org/10.1007/978-3-642-35950-7_16785-1.
[4] Abramovici M (2018) *Engineering Smarter Produkte und Services Plattform (Engineering of Smart Products and Services)*, Industrie 4.0 Studie, Acatech, München. 46 pages. (In German).
[5] Abramovici M, Herzog O (2016) *Engineering im Umfeld von Industrie 4.0. (Engineering in the Context of Industry 4.0)*, Herbert Utz Verlag. (In German).
[6] Abramovici M, Göbel JC, Savarino P (2016) Virtual Twins as Integrative Components of Smart Products. *IFIP Advances in Information and Communication Technology* 492. http://dx.doi.org/10.1007/978-3-319-54660-5_20.
[7] Abramovici M, Göbel JC, Savarino P (2017) Reconfiguration of Smart Products During Their Use Phase Based on Virtual Product Twins. *Annals of CIRP* 66(1):165–168.
[8] Abramovici M, Göbel JC, Savarino P, Gebus P (2017) Towards Smart Product Lifecycle Management With an Integrated Reconfiguration Management. *IFIP Advances in Information and Communication Technology* 517. http://dx.doi.org/10.1007/978-3-319-72905-3_43.
[9] Abramovici M, Neubach M (2009) Knowledge-Based Representation and Analysis of Product Use Information for Feedback to Design. *Proceedings of 2nd CIRP International Seminar on IPS2 Shaker*. ISBN: 978-3-8322-8473-2.
[10] Abramovici M, Neubach M, Fathi M, Holland A (2009) Knowledge-Based Feedback of Product Use Information into Product Development. *Proceedings of 17th International Conference on Engineering Design* 8. ISBN: 978-1904670124.
[11] Abramovici M, Quezada A, Schindler T (2013) MEDA: Manufacturing Energy Demand Assessment Method for Future Production Planning and Product Development. *Proceedings of 11th Global Conference on Sustainable Manufacturing*.
[12] Abramovici M, Stark R (2013) *Smart Product Engineering*, Springer, Berlin, Heidelberg. ISBN: 978-3-642-30816-1.
[13] Abramovici M, Lindner A (2011) Providing Product Use Knowledge for the Design of Improved Product Generations. *Annals of CIRP* 60(1):211–214.
[14] Abramovici M, Lindner A, Walde F, Fathi M, Dienst S (2011) Decision Support for Improving the Design of Hydraulic Systems by Leading Feedback into Product Development. *Proceedings of 18th International Conference on Engineering Design 2011* 9. ISBN: 978-1-904670-32-2.
[15] Achtelik M, Achtelik M, Weiss S, Siegwart R (2011) Onboard IMU and Monocular Vision Based Control for MAVs in Unknown in- and Outdoor Environments. *Proceedings of IEEE ICRA 2011* 3056–3063.
[16] Aeberhard M, Rauch S, Bahram M, et al (2015) Experience, Results and Lessons Learned from Automated Driving on Germany's Highways. *IEEE Intelligent Transportation Systems Magazine* 7(1):42–57.
[17] Agiwal M, Roy A, Saxena N (2016) Next Generation 5G Wireless Networks: A Comprehensive Survey. *IEEE Communications Surveys and Tutorials* 18(3):1617–1655.
[18] Agrawal R, Imieliński T, Swami A (1993) Mining Association Rules Between Sets of Items in Large Databases. *ACM SIGMOD Record* 22(2):207–216.
[19] Ahmed SH, Kim G, Kim D (2013) Cyber Physical System: Architecture, Applications and Research Challenges, IFIP Wireless Days. *IEEE*. http://dx.doi.org/10.1109/WD.2013.6686528.
[20] Ahram TZ, Karwowski W, Soares MM (2012) Embedded Systems Engineering for Products and Services Design. *Work* 41:941–948.
[21] Al-Fuqaha A, Guizani M, Mohammadi M, Aledhari M, Ayyash M (2015) Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys and Tutorials* 17(4):2347–2376.
[22] Alvarez Cabrera AA, Foeken MJ, Tekin OA, et al (2010) Towards Automation of Control Software: A Review of Challenges in Mechatronic Design. *Mechatronics* 20(8):876–886.
[23] Arai T, Shimomura Y (2004) Proposal of Service CAD System—A Tool for Service Engineering. *Annals of CIRP* 53(1):397–400.
[24] Asimov I (1950) Runaround. *I, Robot (The Isaac Asimov Collection Ed.)*, Doubleday, New York NY. p. 40.
[25] Auricht M, Lindow K, Stark R (2013) *Erlebbare Entwicklung und Absicherung technischer Systeme, Stuttgarter Symposium für Produktentwicklung (SSP)*, Fraunhofer Verlag, Stuttgart8–17. ISBN: 978-3-8396-0573-8.
[26] AUTOSAR (Automotive Open System Architecture), http://www.autosar.org. (Accessed 15 April 2019).
[27] Atzori L, Iera A, Morabito G (2010) The Internet of Things: A Survey. *Computer Networks* 54(15):2787–2805.
[28] Baines TS, Lightfoot HW, Evans S, Neely (2007) State-of-the-Art in Product-Service Systems. *Proceedings of the Institution of Mechanical Engineers Part B* 221(10):1543–1552.
[29] Barbato A, Capone A, Rodolfi M, Tagliaferri D (2011) Forecasting the Usage of Household Appliances Through Power Meter Sensors for Demand Management in the Smart Grid. *Proceedings of 2011 IEEE International Conference on Smart Grid Communications*. http://dx.doi.org/10.1109/SmartGridComm.2011.6102356.
[30] Bekas DG, Tsirka K, Baltzis D, Paipetis AS (2016) Self-Healing Materials: A Review of Advances in Materials, Evaluation, Characterization and Monitoring Techniques. *Composites Part B: Engineering* 87:92–119.
[31] Bi ZM, Lang SYT, Verner M, Orban P (2008) Development of Reconfigurable Machines. *Advanced Manufacturing Technology* 39(11-12):1227–1251.
[32] Blaiszik BJ, Kramer SLB, Olugebefola SC, et al (2010) Self-Healing Polymers and Composites. *Annual Review of Materials Research* 40:179–211.
[33] Blois K, (Ed.) (2000), *The Oxford Textbook of Marketing Paperback*, OUP Oxford.
[34] Brown J (2014) *Mastering the Development of Smart Products, Tech-Clarity*, https://www.3ds.com/industries/high-tech/resource-center/white-papers/mastering-the-development-of-smart-products/ (Accessed 15 April 2019).
[35] Brown T (2008) Design Thinking. *Harvard Business Review* 86(6):84–92.
[36] Buchholz Ch, Vorsatz T, Kind S, Stark R (2017) SHPbench—A Smart Hybrid Prototyping Based Environment for Early Testing, Verification and (User Based) Validation of Advanced Driver Assistant Systems of Cars. *Procedia CIRP* 60:139–144.
[37] Burkacky O, Deichmann J, Doll G, Knochenhauer C (2018) *Rethinking Car Software and Electronics Architecture*, https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/rethinking-car-software-and-electronics-architecture (Accessed 15 April 2019).
[38] Butler A (2016) *Amazon Echo: The Ultimate Beginner's Guide to Amazon Echo*, CreateSpace Independent Publishing Platform. ISBN: 978-1539473589.
[39] Campbell M, Hoane AJ, Hsu F (2002) Deep Blue. *Artificial Intelligence* 134(1–2):57–83.
[40] Canedo A (2016) Industrial IoT Lifecycle via Digital Twins. *Proceedings Eleventh IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis* 29.
[41] Cavalieri S, Pezzotta G (2012) Product-Service Systems Engineering: State of the Art and Research Challenges. *Computers in Industry* 63(4):278–288.
[42] Chauvin F, Fanmuy G (2014) System Engineering on 3DEXPERIENCE Platform–UAS Use Case. *Proceedings of the Poster Workshop at the 2014 Complex Systems Design & Management International Conference co-located with 5th International Conference on Complex System Design & Management (CSD&M 2014)* 113–126, http://ceur-ws.org/Vol-1234/paper-11.pdf.
[43] Chan M, Estève D, Escriba C, Campo E (2008) A Review of Smart Homes-Present State and Future Challenges. *Computer Methods and Programs in Biomedicine* 91(1):55–81.
[44] Cheminod M, Durante L, Valenzano A (2013) Review of Security Issues in Industrial Networks. *IEEE Transactions on Industrial Informatics* 9(1):277–293.
[45] Chen Y, Liu ZL, Xie YB (2012) A Knowledge-Based Framework for Creative Conceptual Design of Multi-Disciplinary Systems. *Computer Aided Design* 44(2):146–153.
[46] Chen C, Liu Y, Kumar M, Qin J (2018) Energy Consumption Modelling Using Deep Learning Technique—A Case Study of EAF. *Procedia CIRP* 72:1063–1068.
[47] Cherdantseva Y, Burnap P, Blyth A, et al (2016) A Review of Cyber Security Risk Assessment Methods for SCADA Systems. *Computers and Security* 56:1–27.
[48] Choi S, Jung K, Noh SD (2015) Virtual Reality Applications in Manufacturing Industries: Past Research, Present Findings, and Future Directions. *Concurrent Engineering* 23(1):40–63.
[49] Cimino D, Ferrero A, Queirolo L, et al (2017) A Low-Cost, Open-Source Cyber Physical System for Automated, Remotely Controlled Precision Agriculture. *Lecture Notes in Electrical Engineering* 409:191–203.
[50] Chryssolouris G, Mavrikios D, Papakostas N, et al (2009) Digital Manufacturing: History, Perspectives, and Outlook. *Proceedings of the Institution of Mechanical Engineers Part B* 223(5):451–462.
[51] Cusumano MA (2012) Platforms Versus Products: Observations from the Literature and History. Kahl SJ, Silverman BS, Cusumano MA, (Eds.) *History and Strategy Advances in Strategic Management*, vol. 29. 35–67.
[52] Da Silveira G, Borenstein D, Fogliatto FS (2001) Mass Customization: Literature Review and Research Directions. *Production Economics* 72(1):1–13.
[53] D'Amelio V, Chmarra MK, Tomiyama T (2011) Early Design Interference Detection Based on Qualitative Physics. *Research in Engineering Design* 22(4):223–243.
[54] Damerau Th, Hayka H, Neumeyer S, et al (2014) Intelligent Information Technologies to Enable Next Generation PLM. *Product Lifecycle Management for a Global Market*, Springer: 485–495. ISBN: 978-3-662-45936-2.

[55] Damgrave R, Lutters E (2018) Enhancing Development Trajectories of Synthetic Environments. *Annals of CIRP* 67(1):137–140.
[56] Darvish K, Wanderlingh F, Bruno B, et al (2018) Flexible Human—Robot Cooperation Models for Assisted Shop-Floor Tasks. *Mechatronics* 51:97–114.
[57] Dawid H, Decker R, Hermann T, et al (2017) Management Science in the Era of Smart Consumer Products: Challenges and Research Perspectives. *Central European Journal of Operations Research* 25(1):203–230.
[58] Dutra De S, Silva Jr D. (2016) Product-Service Architecture (PSA): Toward a Service Engineering Perspective in Industry 4.0. *IFAC-PapersOnLine* 49 (31):91–96.
[59] De Brito MS, Hoque S, Steinke R, et al (2018) Application of the Fog Computing Paradigm to Smart Factories and Cyber-Physical Systems. *Transactions on Emerging Telecommunications Technologies* 29(4). http://dx.doi.org/10.1002/ett.3184.
[60] Di Natale M, Sangiovanni-Vincentelli AL (2010) Moving from Federated to Integrated Architectures in Automotive: The Role of Standards, Methods and Tools. *Proceedings of the IEEE* 98(4):603–620.
[61] Durantin A, Fanmuy G, Miet S, Pegon V (2017) Disruptive Innovation in Complex Systems. *Proceedings Complex Systems Design & Management (CSDM 2016)*, 41–56.
[62] Edrington B, Zhao B, Hansel A, et al (2014) Machine Monitoring System Based on MTConnect Technology. *Procedia CIRP* 22:92–97.
[63] Eichhorn SJ, Dufresne A, Aranguren M, et al (2010) Review: Current International Research into Cellulose Nanofibres and Nanocomposites. *Journal of Materials Science* 45(1):1–33.
[64] ElMaraghy HA (2005) Flexible and Reconfigurable Manufacturing Systems Paradigms. *Flexible Manufacturing Systems* 17(4):261–276.
[65] ElMaraghy W, ElMaraghy H, Tomiyama T, Monostori L (2012) Complexity in Engineering Design and Manufacturing. *Annals of CIRP* 61(2):793–814.
[66] Elsy D, Jennings P, Roy R, (Eds.) (2018), *Through-life Engineering Services for Mobility as a Service*, High Value Manufacturing Catapult, Warwick Manufacturing Group, Cranfield University32 pages.
[67] Erl T (2005) *Service-Oriented Architecture: Concepts, Technology, and Design*, Prentice Hall. ISBN: 978-0131858589.
[68] Estefan JA (2008) *Survey of Model-Based Systems Engineering (MBSE) Methodologies, INCOSE MBSE Initiative, INCOSE-TD-2007-003-02*, http://www.omgsysml.org/MBSE_Methodology_Survey_RevB.pdf (Accessed 15 April 2019).
[69] European Regulators Group for Electricity and Gas (2010) *Position Paper on Smart Grids, An ERGER Conclusions Paper*. Ref: E10-EQS-38-05 ERGEG, https://www.ceer.eu/documents/104400/-/-/3cf25df7-88cb-3ce3-f838-aa2d012ac45c (Accessed 15 April 2019).
[70] European Union (2016) Regulation on the Protection of Natural Persons With Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing, Directive 95/46/EC (Data Protection Directive). *EU Journal L119* ;(May)1–88.
[71] Eyisi E, Zhang Z, Koutsoukos X, et al (2013) Model-Based Control Design and Integration of Cyberphysical Systems: An Adaptive Cruise Control Case Study. *Journal of Control Science and Engineering* 2013678016.
[72] Fanmuy G, Petrauskas D (2018) An Automotive MBSE Case Study: An Interdisciplinary Role-Based Design With 3DEXPERIENCE Platform and MagicDraw. *No Magic World Symposium 2018 (NMWS2018)*. https://nmws2018.com/speakers/abstracts/item/an-automotive-mbse-case-study-an-interdisciplinary-role-based-design-with-3dexperience-platform-and-magicdraw (Accessed 15 April 2019).
[73] Ferrucci D, Brown E, Chu-Carroll J, et al (2010) Building Watson: An Overview of the DeepQA Project. *AI Magazine* 31(3):59–79.
[74] Floreano D, Wood RJ (2015) Science, Technology and the Future of Small Autonomous Drones. *Nature* 521(7553):460–466.
[75] Fong T, Nourbakhsh I, Dautenhahn K (2003) A Survey of Socially Interactive Robots. *Robotics and Autonomous Systems* 42(3–4):143–166.
[76] Forbes H, Schaefer D (2017) Social Product Development: The Democratization of Design, Manufacture and Innovation. *Procedia CIRP* 60:404–409.
[77] Forsberg K, Mooz H (1991) The Relationship of System Engineering to the Project Cycle. *Presented at the Joint conference sponsored by: National Council on Systems Engineering (NCOSE) and American Society for Engineering Management (ASEM)* 1–12.
[78] Fu Y, Zhang Y, Qiao H, et al (2015) Analysis of Feature Extracting Ability for Cutting State Monitoring Using Deep Belief Networks. *Procedia CIRP* 31:29–34.
[79] Fujishima M, Narimatsu K, Irino N, Ido Y (2018) Thermal Displacement Reduction and Compensation of a Turning Center. *CIRP Journal of Manufacturing Science and Technology* 22:111–115.
[80] Fujishima M, Ohno K, Nishikawa S, et al (2016) Study of Sensing Technologies for Machine Tools. *CIRP Journal of Manufacturing Science and Technology* 14:71–75.
[81] Functional Mockup Interface, FMI, https://fmi-standard.org. (Accessed 15 April 2019).
[82] Gershenfeld N, Krikorian R, Cohen D (2004) The Internet of Things. *Scientific American* 291(4):76–81.
[83] Gharavi H, Ghafurian R (2011) Smart Grid: The Electric Energy System of the Future. *Proceedings of the IEEE* 99(6):917–921.
[84] Gianni D, D'Ambrogio A, Tolk A (2015) *Modeling and Simulation-Based Systems Engineering Handbook*, CRC Press. ISBN 9781138748941.
[85] Glaessgen E (2012) The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. *Proceedings of 53rd AIAA/ASME/ASCE/AHS/ASC Structure, Structure Dynamics and Materials Conference—Special Session on the Digital Twin*, . 14 pages.
[86] Goedkoop MJ, van Halen CJG, te Riele HRM, Rommens PJM (1999) *Product Service systems Ecological and Economic Basics report submitted to the Dutch Ministry of Economic Affairs*. 132 pages.
[87] Gonzalez C (2017) *Amazon warehouse, Machine Design* (Accessed 15 April 2019), https://www.machinedesign.com/motion-control/changing-future-warehouses-amazon-robots.
[88] Goodfellow I, Bengio Y, Courville A, Bach F (2017) *Deep Learning*, MIT Press. ISBN: 978-0262035613.
[89] Gray P (2014) *Designing the User Experience, Dassault Systèmes White Paper Series* (Accessed 15 April 2019), https://www.3ds.com/fileadmin/Industries/High-Tech/Pdf/whitepapers/Designing-The-User-Experience-high-tech.pdf.
[90] Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. *Future Generation Computer Systems* 29(7):1645–1660.
[91] Harrison A (2006) Design for Service–Harmonising Product Design with a Services Strategy. *Proceedings of ASME Turbo Expo* 2006(2):135–143.
[92] Hassenzahl M, Tractinsky N (2006) User Experience—A Research Agenda. *Behaviour and Information Technology* 25(2):91–97.
[93] Hayes CC, Goel AK, Tumer IY, et al (2011) Intelligent Support for Product Design: Looking Backward, Looking Forward. *Journal of Computing and Information Science in Engineering* 11(2). 21007.
[94] Hehenberger P, Vogel-Heuser B, Bradley D, et al (2016) Design, Modelling, Simulation and Integration of Cyber Physical Systems: Methods and Applications. *Computers in Industry* 82:273–289.
[95] Herter J, Ovtcharova J (2016) A Model Based Visualization Framework for Cross Discipline Collaboration in Industry 4.0 Scenarios. *Procedia CIRP* 57:398–403.
[96] Hon KKB, Li L, Hutchings IM (2008) Direct Writing Technology-Advances and Developments. *Annals of CIRP* 57(2):601–620.
[97] Hwang G, Lee J, Park J, Chang T-W (2017) Developing Performance Measurement System for Internet of Things and Smart Factory Environment. *International Journal of Production Research* 55(9):2590–2602.
[98] IEEE (1998) *IEEE 1320.1-1998, Standard for Functional Modeling Language—Syntax and Semantics for IDEF0*.
[99] IEEE (2005) *IEEE 1220-2005, IEEE Standard for Application and Management of the Systems Engineering Process* (Accessed 15 April 2019), https://standards.ieee.org/standard/1220-2005.html.
[100] Industrial Internet Consortium, https://www.iiconsortium.org. (Accessed 15 April 2019).
[101] International Electronical Commission (IEC) (2015) *Dependability, Electropedia* (Accessed 15 April 2019)., http://www.electropedia.org/iev/iev.nsf/display?openform&ievref=192-01-22.
[102] International Standard Orgnisation (2011) *ISO 26262 Road vehicles Functional Safety Part 1–10*.
[103] International Standard Orgnisation (2001) *ISO/IEC/IEEE 42010 Systems and Software Engineering—Architecture Description*.
[104] International Standard Orgnisation (2010) *ISO 9241-210, Ergonomics of Human-System Interaction–Part 210: Human-centred Design for Interactive Systems*.
[105] Jiao J, Tseng MM (1999) Methodology of Developing Product Family Architecture for Mass Customization. *Intelligent Manufacturing* 10(1):3–20.
[106] Kagermann H, Wahlster W, Helbig J, (Eds.) (2013), *Recommendations for Implementing the Strategic Initiative Industrie 4.0: Final Report of the Industrie 4.0 Working Group*, http://en.acatech.de/publication/recommendations-for-implementing-the-strategic-initiative-industrie-4-0-final-report-of-the-industrie-4-0-working-group/ (Accessed 15 April 2019).
[107] Kemény Z, Beregi RJ, Erdos G, Nacsa J (2016) The MTA SZTAKI Smart Factory: Platform for Research and Project-Oriented Skill Development in Higher Education. *Procedia CIRP* 54:53–58.
[108] Kim WS (2011) Robotized Products. *Proceedings URAI 2011–8th International Conference on Ubiquitous Robots and Ambient Intelligence*, 612–614.
[109] Kitatani T (2010) Global Monitoring and Support Management System for Construction Machinery Using Satellite Communication. *Proceedings 2010 7th International Conference on Service Systems and Service Management*, 211–215.
[110] Knoch S, Deru M, Bergweiler S, Haupert J (2014) *Novel Consumer-to-Product Interactions with Context-Aware Embedded Platforms Lecture Notes in Computer Science*, 8515. Springer. 513–524.
[111] Gibson I, Rosen DW, Stucker B (2010) *Additive Manufacturing Technologies: Rapid Prototyping to Direct Digital Manufacturing*, Springer. ISBN: 978-1-4419-1119-3.
[112] Kennon D, Schutte CSL, Lutters D (2015) An Alternative View to Assessing Antifragility in an Organisation: A Case Study in a Manufacturing SME. *Annals of CIRP* 64(1):177–180.
[113] Khalifa T, Naik K, Nayak A (2011) A Survey of Communication Protocols for Automatic Meter Reading Applications. *IEEE Communications Surveys and Tutorials* 13(2):168–182.
[114] Kind S, Exner K, Neumeyer S, Stark R (2016) *Smart Engineering in the Context of Industrie 4.0: Modular Systems for Experiencable Design Validation of Assembly Systems, Gemeinsames Kolloquium Konstruktionstechnik 2016*, Shaker Publishing, Rostock251–258. ISBN: 978-3-8440-4732-5.
[115] Kölsch P, Herder CF, Zimmermann V, Aurich JC (2017) A Novel Concept for the Development of Availability-Oriented Business Models. *Procedia CIRP* 64:340–344.
[116] Komatsu (2013) *Komatsu Wireless Monitoring System* (Accessed 15 April 2019), https://www.komatsu.eu/-/media/sites/komatsu/brochures/komtrax/komtrax_eess018706_1302_86944.ashx?v=9886518fd9ad492f965cc01c0edb1c81&la=en?v.
[117] Komoto H, Tomiyama T (2008) Integration of a Service CAD and a Life Cycle simulator. *Annals of CIRP* 57(1):9–12.
[118] Komoto H, Tomiyama T (2011) Multi-Disciplinary System Decomposition of Complex Mechatronics Systems. *Annals of CIRP Annals* 60(1):191–194.
[119] Komoto H, Tomiyama T (2012) A Framework for Computer-Aided Conceptual Design and Its Application to System Architecting of Mechatronics Products. *Computer-Aided Design* 44(10):931–946.

[120] Kondoh S, Sato RK, Tomiyama T, Umeda Y (2000) Self-Organization of the Cellular Manufacturing System. *CIRP Journal of Manufacturing Systems* 30 (6):507–513.
[121] Königs SF, Beier G, Figge A, Stark R (2012) Traceability in Systems Engineering—Review of Industrial Practices, State-of-the-Art Technologies and New Research Solutions. *Advanced Engineering Informatics* 26(4):924–940.
[122] Koren Y, Heisel U, Jovane F, et al (1999) Reconfigurable Manufacturing Systems. *Annals of CIRP* 48(2):527–540.
[123] Kortuem G, Kawsar F, Sundramoorthy V, Fitton D (2010) Smart Objects as Building Blocks for the Internet of Things. *IEEE Internet Computing* 14(1):44–51.
[124] Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet Classification With Deep Convolutional Neural Networks. *Proceedings of 25th International Conf on Neural Information Processing Systems* 1:1097–1105.
[125] Kuksa I, Fisher T, (Eds.) (2017), *Design for Personalisation*, Taylor & Francis.
[126] Kuhnert K, Stürmer C, Koster A (2018) *Five Trends Transforming the Automotive Industry, PricewaterhouseCoopers GmbH* (Accessed 15 April 2019), https://www.pwc.com/gx/en/industries/automotive/assets/pwc-five-trends-transforming-the-automotive-industry.pdf.
[127] Kusiak A (2018) Smart Manufacturing. *Journal of Production Research* 56(1–2):508–517.
[128] Lecun Y, Bengio Y, Hinton G (2015) Deep Learning. *Nature* 521(7553):436–444.
[129] Lee EA (2008) Cyber Physical Systems: Design Challenges. *Proceedings of 11th IEEE Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing*, 363–369.
[130] Lee GH (1997) Reconfigurability Consideration Design of Components and Manufacturing Systems. *Advanced Manufacturing Technology* 13(5):376–386.
[131] Lee J, Ardakani HD, Yang S, Bagheri B (2015) Industrial Big Data Analytics and Cyber-Physical Systems for Future Maintenance & Service Innovation. *Procedia CIRP* 38:3–7.
[132] Lee J, Bagheri B, Kao HA (2015) A Cyber-Physical Systems Architecture for Industry 4.0-Based Manufacturing Systems. *Manufacturing Letters* 3:18–23.
[133] Lee J, Kao HA (2014) Dominant Innovation Design for Smart Products-Service Systems (PSS): Strategies and Case Studies. *Annual SRII Global Conference SRII IEEE* 305–310. http://dx.doi.org/10.1109/SRII.2014.25.
[134] Lee J, Kao HA, Yang S (2014) Service Innovation and Smart Analytics for Industry 4.0 and Big Data Environment. *Procedia CIRP* 16:3–8.
[135] Lee JY, Yoon JS, Kim B-H (2017) A Big Data Analytics Platform for Smart Factories in Small and Medium-Sized Manufacturing Enterprises: An Empirical Case Study of a Die Casting Factory. *Precision Engineering and Manufacturing* 18(10):1353–1361.
[136] Leitão P, Restivo F (2006) ADACOR: A Holonic Architecture for Agile and Adaptive Manufacturing Control. *Computers in Industry* 57(2):121–130.
[137] Leitão P, Colombo AW, Karnouskos S (2016) Industrial Automation Based on Cyber-Physical Systems Technologies: Prototype Implementations and Challenges. *Computers in Industry* 81:11–25.
[138] Leonard J, How J, Teller S, et al (2008) A Perception-Driven Autonomous Urban Vehicle. *Journal of Field Robotics* 25:727–774.
[139] Ren L, Zhang L, Tao F, et al (2015) Cloud Manufacturing: From Concept to Practice. *Enterprise Information Systems* 9(2):186–209.
[140] Liu H, Fang T, Zhou T, et al (2018) Deep Learning-Based Multimodal Control Interface for Human-Robot Collaboration. *Procedia CIRP* 72:3–8.
[141] Liu ZL, Chen Y, Xie YB (2012) A Knowledge-Based System for Multi-Disciplinary Conceptual Design Synthesis. *Proceedings of the Institution of Mechanical Engineers Part C* 226(12):2950–2966.
[142] Luetzenberger J, Klein P, Thoben K-D (2013) Using Knowledge Based Engineering to Support the Design of Smart Products. *Proceedings of 19th International Conference on Engineering Design (ICED13) Design Society* 6:63–72.
[143] Lünnemann P, Wang WM, Stark R (2017) Methodische Analyse Der Entwicklungsaktivitäten. *Proceedings of 15th Gemeinsames Kolloquium Konstruktionstechnik Interdisziplinäre Produktentwicklung* 89–98. ISBN: 978-3-940402-15-8.
[144] Lünnemann P, Wang WM, Ibánez Manteca P, Stark R (2017) Engineering Activities—Considering Value Creation from a Holistic Perspective. *Proceedings of 2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, . ISBN 978-1-5386-0774-9/17.
[145] Lutters E (2018) Pilot Production Environments Driven by Digital Twins. *South African Journal of Industrial Engineering* 29(3):40–53.
[146] Lutters E, Lange Jde, Damgrave RGJ (2019) Virtual Dashboards in Pilot Production Environments. *International Conference Competitive Manufacturing; COMA'19* 22–27, http://hdl.handle.net/10019.1/105429.
[147] Lutters E, van Houten FJAM, Bernard (2014) Tools and Techniques for Product Design. *Annals of CIRP* 63(2):607–630.
[148] Lützenberger J, Klein P, Hribernik K, Thoben K-D (2016) Improving Product-Service Systems by Exploiting Information from the Usage Phase. A Case Study. *Procedia CIRP* 47:376–381.
[149] Lv C, Zhang J, Nuzzo P, et al (2015) Design Optimization of the Control System for the Powertrain of an Electric Vehicle: A Cyber-Physical System Approach. *Proceedings of 2015 IEEE International Conference on Mechatronics and Automation* 814–819.
[150] Maier MW (1998) Architecting Principles for Systems-of-Systems. *Systems Engineering* 1(4):267–284.
[151] Marilungo E, Papetti A, Germani M, Peruzzini M (2017) From PSS to CPS Design: A Real Industrial Use Case Toward Industry 4.0. *Procedia CIRP* 64:357–362.
[152] Masrur A, Kit M, Matěna V, et al (2016) Component-Based Design of Cyber-Physical Applications with Safety-Critical Requirements. *Microprocessors and Microsystems* 42:70–86.
[153] Mattelmäki T, Roto V, Lee J-J, Zimmerman J (2018) Experience Design Meets Service Design-Method Clash or Marriage? Extended Abstracts of the 2018. *CHI Conference on Human Factors in Computing Systems (CHI EA' 18) ACM*. http://dx.doi.org/10.1145/3170427.3170626. Paper W26, 6 pages.
[154] Mayer-Schönberger V, Cukier K (2013) *Big Data: A Revolution That Will Transform How We Live, Work, and Think*, John Murray Publishers, London.
[155] Mehrabi MG, Ulsoy AG, Koren Y, Heytler P (2002) Trends and Perspectives in Flexible and Reconfigurable Manufacturing Systems. *Journal of Intelligent Manufacturing* 13(2):135–146. 1.
[156] Mehrpouyan H, Haley B, Dong A, et al (2015) Resiliency Analysis for Complex Engineered System Design. *AI EDAM* 29(1):93–108.
[157] Meyer GG, Främling K, Holmström J (2009) Intelligent Products: A Survey. *Computers in Industry* 60(3):137–148.
[158] Michniewicz J, Reinhart G (2016) Cyber-Physical-Robotics—Modelling of Modular Robot Cells for Automated Planning and Execution of Assembly Tasks. *Mechatronics* 34:170–180.
[159] Miorandi D, Sicari S, De Pellegrini F, Chlamtac I (2012) Internet of Things: Vision, Applications and Research Challenges. *Ad Hoc Networks* 10(7):1497–1516.
[160] Mohan Y, Ponnambalam SG (2009) An Extensive Review of Research in Swarm Robotics. *Proceedings of 2009 World Congress on Nature and Biologically Inspired Computing* 140–145. http://dx.doi.org/10.1109/NABIC.2009.5393617.
[161] Monk E, Wagner B (2012) *Concepts in Enterprise Resource Planning*, 4th edition. Cengage Learning. ISBN-13: 978-1111820398.
[162] Monostori L, Kádár B, Bauernhansl T, et al (2016) Cyber-Physical Systems in Manufacturing. *Annals of CIRP* 65(2):621–641.
[163] Moon SK, Simpson TW, Kumara SRT (2008) A Process Model and Data Mining to Support Designing Families of Services. *Proceedings of ASME DETC2007* 6 (Part A):591–600.
[164] Mori M, Fujishima M, Komatsu M, et al (2008) Development of Remote Monitoring and Maintenance System for Machine Tools. *Annals of CIRP* 57 (1):433–436.
[165] Mørkeberg Torry-Smith J (2012) Challenges in Designing Mechatronic Systems. *Journal of Mechanical Design* 135(1):11005.
[166] Mosterman PJ, Zander J (2016) Industry 4.0 as a Cyber-Physical System Study. *Software and Systems Modeling* 15(1):17–29.
[167] Mourtzis D, Fotia S, Vlachou E, Koutoupes A (2018) A Lean PSS Design and Evaluation Framework Supported by KPI Monitoring and Context Sensitivity Tools. *Advanced Manufacturing Technology* 94(5-8):1623–1637.
[168] Murata S, Yoshida E, Kamimura A, et al (2002) M-TRAN: Self-Reconfigurable Modular Robotic System. *IEEE/ASME Transactions on Mechatronics* 7(4):431–441.
[169] National Institute of Standards and Technology, https://www.nist.gov/el/cyber-physical-systems. (Accessed 15 April 2019).
[170] National Science Foundation (2017) *Cyber-Physical Systems (CPS)*Available at: https://www.nsf.gov/publications/pub_summ.jsp?ods_key=nsf18538. (Accessed 15April 2019) .
[171] Nee AYC, Ong SK, Chryssolouris G, Mourtzis D (2012) Augmented Reality Applications in Design and Manufacturing. *Annals of CIRP* 61(2):657–679.
[172] Nielsen Norman Group (2019) https://www.nngroup.com/articles/definition-user-experience/ (Accessed 15 April 2019).
[173] OMG (2010) *Requirements Interchange Format (ReqIF)* (Accessed 15 April 2019), https://www.omg.org/cgi-bin/doc?mantis/10-03-07.pdf.
[174] OMG (2017) *Systems Modeling Language (SysML) Version 1.5* (Accessed 15 April 2019), https://www.omg.org/spec/SysML/1.5/.
[175] OMG (2017) *Unified Modeling Language (UML) Version 2.5.1* (Accessed 15 April 2019), https://www.omg.org/spec/UML/.
[176] Open Services for Lifecycle Collaboration (OSLC), https://open-services.net (Accessed 15 April 2019).
[177] Pagoropoulos A, Pigosso DCA, McAloone TC (2017) The Emergent Role of Digital Technologies in the Circular Economy: A Review. *Procedia CIRP* 64:19–24.
[178] Palattella MR, Dohler M, Grieco A, et al (2016) Internet of Things in the 5G Era: Enablers, Architecture, and Business Models. *IEEE Journal of Selected Areas in Communications* 34(3):510–527.
[179] Peters DL, Papalambros PY, Ulsoy AG (2010) Sequential Co-Design of an Artifact and Its Controller via Control Proxy Functions. *IFAC Proceedings (IFAC-PapersOnline)* 43(18):125–130.
[180] Pfrommer J, Zimmerling C, Liu J, et al (2018) Optimisation of Manufacturing Process Parameters Using Deep Neural Networks as Surrogate Models. *Procedia CIRP* 72:426–431.
[181] Porter ME, Heppelmann JE (2014) How Smart, Connected Products Are Transforming Competition. *Harvard Business Review* 92(11):64–88.
[182] Preidel M, Damerau T, Stark R (2017) Web-Based Information Visualization for Early Customer Integration. *Proceedings of Stuttgarter Symposium Produktentwicklung SSP 2017*. ISSN: 2364-4885.
[183] Quartarone E, Mustarelli P (2011) Electrolytes for Solid-State Lithium Rechargeable Batteries: Recent Advances and Perspectives. *Chemical Society Reviews* 40(5):2525–2540.
[184] Rammig FJ, Grösbrink S, Stahl K, Zhao Y (2014) Designing Self-Adaptive Embedded Real-Time Software-Towards System Engineering of Self-Adaptation. *Proceedings of 2014 Brazilian Symposium on Computing Systems Engineering Manaus 2014* 37–42. http://dx.doi.org/10.1109/SBESC.2014.15.
[185] Plattform Industrie 4.0 (2015) *Reference Architecture Model Industrie 4.0 (RAMI4.0)* (Accessed 15 April 2019), https://www.bitkom.org/sites/default/files/file/import/150410-Umsetzungsstrategie-0.pdf.
[186] Ralston PAS, Graham JH, Hieb JL (2007) Cyber Security Risk Assessment for SCADA and DCS Networks. *ISA Transactions* 46(4):583–594.
[187] Rashid A, Masood T, Erkoyuncu JA, et al (2018) Enterprise Systems' Life Cycle in Pursuit of Resilient Smart Factory for Emerging Aircraft Industry: a Synthesis of Critical Success Factors (CSFs), Theory, Knowledge Gaps, and Implications. *Enterprise Information Systems* 12(2):96–136.

[188] Reger A, Dumler J, Lobachev O, Neuberger J, Steinhilper R (2018) Machine Allocation via Pattern Recognition in Harmonic Waves of Manufacturing Plants. *Procedia CIRP* 67:69–74.
[189] Riel A, Kreiner C, Macher G, Messnarz R (2017) Integrated Design for Tackling Safety and Security Challenges of Smart Products and Digital Manufacturing. *Annals of CIRP* 66(1):177–180.
[190] Roy R, Stark R, Tracht K, et al (2016) Continuous Maintenance and the Future–Foundations and Technological Challenges. *Annals of CIRP* 65(2):667–688.
[191] Russell S, Norvig P (2009) *Artificial Intelligence: A Modern Approach*, 3rd edition. Pearson. ISBN: 978-0136042594.
[192] Sakagami Y, Watanabe R, Aoyama C, et al (2002) The Intelligent ASIMO: System Overview and Integration. *Proceedings of IEE/RSJ Intl Conf Intelligent Robots and Systems* 3:2478–2483.
[193] Sakao T, Kondoh S, Umeda Y, Tomiyama T (1996) Development of a Cellular Automatic Warehouse. *Proceedings 1996 IEEE/RSJ International Conference Intelligent Robots and Systems (IROS 96), IEEE 96 CH 35908, IEEE, (1996)*, 324–331.
[194] Sakao T, Shimomura Y (2007) Service Engineering: a Novel Engineering Discipline for Producers to Increase Value Combining Service and Product. *Journal of Cleaner Production* 15(6):590–604.
[195] Savarino P, Abramovici M, Göbel JC (2018) A Methodological Approach for the Identification of Context-Specific Reconfiguration Options in the PLM-Context. *Proceedings of PLM18*.
[196] Savarino P, Abramovici M, Göbel JC, Gebus P (2018) Design for Reconfiguration as Fundamental Aspect of Smart Products. *Procedia CIRP* 70:374–379.
[197] Schleich B, Anwer N, Mathieu L, Wartzack S (2017) Shaping the Digital Twin for Design and Production Engineering. *Annals of CIRP* 66(1):141–144.
[198] Schmidt DC (2006) Model-Driven Engineering. *EEE Computer* 39(2):25–31.
[199] Schmidt M, Merklein M, Bourell D, et al (2017) Laser Based Additive Manufacturing in Industry and Academia. *Annals of CIRP* 66(2):561–583.
[200] Schneier B (2013) *To Kill a Centrifuge*37 pages, https://www.langner.com/wp-content/uploads/2017/03/to-kill-a-centrifuge.pdf (Accessed 15 April 2019) .
[201] Scholze S, Correia AT, Stokic D (2016) Novel Tools for Product-Service System Engineering. *Procedia CIRP* 47:120–125.
[202] Schweitzer F, Van den Hende EA (2016) To Be or Not to Be in Thrall to the March of Smart Products. *Psychology and Marketing* 33(10):830–842.
[203] Sehgal NK, Bhatt PCP (2018) *Cloud Computing: Concepts and Practices*, 1st ed. Springer. ISBN-13: 978-3319778389.
[204] Sell R, Seiler S (2012) Improvements of Multi-Disciplinary Engineering Study by Exploiting Design-Centric Approach, Supported by Remote and Virtual Labs. *International Journal of Engineering Education* 28(4):759–766.
[205] Seol M-L, Ivaškevičiūtė R, Ciappesoni MA (2018) All 3D Printed Energy Harvester for Autonomous and Sustainable Resource Utilization. *Nano Energy* 52:271–278.
[206] Shi W, Cao J, Zhang Q, et al (2016) Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal* 3(5):637–646.
[207] Shimomura Y, Tanigawa S, Umeda Y, Tomiyama T (1995) Development of Self-Maintenance Photocopiers. *AI Magazine* 16(4):41–53.
[208] Shiue Y-R, Lee K-C, Su C-T (2018) Real-Time Scheduling for a Smart Factory Using a Reinforcement Learning Approach. *Computers and Industrial Engineering* 125:604–614.
[209] Shrouf F, Ordieres J, Miragliotta G (2014) Smart Factories in Industry 4.0: A Review of the Concept and of Energy Management Approached in Production Based on the Internet of Things Paradigm. *IEEE International Conference Industrial Engineering and Engineering Management* 697–701.
[210] Siciliano B, Khatib O, (Eds.) (2016), *Springer Handbook of Robotics*, Springer.
[211] Siemens AG (2018) *1D CAE Software* . (Accessed 15 April 2019), https://www.plm.automation.siemens.com/global/en/our-story/glossary/mechatronic-system-simulation-1d-cae/13110.
[212] Sierla S, Tumer I, Papakonstantinou N, et al (2012) Early Integration of Safety to the Mechatronic System Design Process by the Functional Failure Identification and Propagation Framework. *Mechatronics* 22(2):137–151.
[213] Siemens, https://www.plm.automation.siemens.com/global/en/products/. (Accessed 15 April 2019).
[214] Simpson TW (2004) Product Platform Design and Customization: Status and Promise. *AIEDAM* 18(1):3–20.
[215] Singh VB, Benoit T, Braibant V (2018) Breaking Down Silos With Contract Based Design for Industrial Software Development: Illustrated Through an Aerospace Case Study. *Proceedings of 9th European Congress on Embedded Real Time Software and Systems*, , https://www.erts2018.org/uploads/program/ERTS_2018_paper_22.pdf.
[216] Söderberg R, Wärmefjord K, Carlson JS, Lindkvist L (2017) Toward a Digital Twin for Real-Time Geometry Assurance in Individualized Production. *Annals CIRP* 66(1):137–140.
[217] Son SD, Hubrechts J, Awatsu L, Bhave A, Van der Auweraer A (2018) A Simulation-Based Testing and Validation Framework for ADAS Development. *Proceedings of 7th Transport Research Arena 2018*.
[218] Sridhar S, Hahn A, Govindarasu M (2012) Cyber-Physical System Security for the Electric Power Grid. *Proceedings of the IEEE* 100(1):210–224.
[219] Stark J (2011) *Product Lifecycle Management: 21st Century Paradigm for Product Realisation*, 2nd ed). Springer. ISBN-13: 978-0857295453.
[220] Stark R, Beckmann-Dobrev B, Adenauer J (2010) Ein Interdisziplinärer Ansatz Zur Multimodalen Funktionalen Absicherung Mechatronischer Systeme Am Beispiel Einer PKW-Heckklappe. *An interdisciplinary approach to a multimodal functional validation of mechatronic systems tested at an automotive tailgate) Grazer Symposium Virtuelles Fahrzeug* 18–27.
[221] Stark R, Damerau T (2019) Digital Twin. in Chatti S, Laperrière L, Reinhart G, Tolio T, (Eds.) *CIRP Encyclopedia of Production Engineering*, Springer, Berlin, Heidelbergin print.
[222] Stark R, Kind S, Neumeyer S (2017) Innovations in Digital Modelling for Next Generation Manufacturing System Design. *Annals of CIRP* 66(1):169–172.
[223] Stark R, Lindow K, Reich D, Vorsatz T (2018) Smart Engineering Capabilities for New Generation Product Offerings. *Anais do 23 Seminário Internacional de Alta Tecnologia*. ISSN: 2175-9960.
[224] Stock T, Seliger G (2016) Opportunities of Sustainable Manufacturing in Industry 4.0. *Procedia CIRP* 40:536–541.
[225] Tao F, Zhang M, Liu Y, Nee AYC (2018) Digital Twin Driven Prognostics and Health Management for Complex Equipment. *Annals of CIRP* 67(1):169–172.
[226] Tas OS, Kuhnt F, Zollner JM, Stiller C (2016) Functional System Architectures Towards Fully Automated Driving. *Proceedings of 2016 IEEE Intelligent Vehicles Symposium (IV)*, 304–309.
[227] Teti R, Jemielniak K, O'Donnell G, Dornfeld D (2010) Advanced Monitoring of Machining Operations. *Annals of CIRP* 59(2):717–739.
[228] Tharumarajah A, Wells AJ, Nemes L (1996) Comparison of the Bionic, Fractal and Holonic Manufacturing System Concepts. *International Journal of Computer Integrated Manufacturing* 9(3):217–226.
[229] Thompson MK, Moroni G, Vaneker T, et al (2016) Design for Additive Manufacturing: Trends, Opportunities, Considerations, and Constraints. *Annals of CIRP* 65(2):737–760.
[230] Thorpe CE, Herbert M, Kanade T, Shafer SA (1991) Toward Autonomous Driving: the CMU Navlab. Part I—Perception. *IEEE Expert* 6(4):3–42.
[231] Tomiyama T (2001) Service Engineering to Intensify Service Contents in Product Life Cycles. *Proceedings of Second International Symposium on Environmentally Conscious Design and Inverse Manufacturing (EcoDesign 2001) IEEE* 613–618. http://dx.doi.org/10.1109/.2001.992433.
[232] Tomiyama T (2012) Architecture-Centric Model-Based Product Development. *Proceedings of 13th Mechatronics Forum International Conference* 2/3:434–443. ISBN: 978-3-99033-045-6.
[233] Tomiyama T, D'Amelio V, Urbanic J, ElMaraghy W (2007) Complexity of Multi-Disciplinary Design. *Annals of CIRP* 56(1):185–188.
[234] Tomiyama T, Gu P, Jin Y, et al (2009) Design Methodologies: Industrial and Educational Applications. *Annals of CIRP* 58(2):543–565.
[235] Tomiyama T, Moyen F (2018) Resilient Architecture for Cyber-Physical Production Systems. *Annals of CIRP* 67(1):161–164.
[236] Tomiyama T, van Beek TJ, Alvarez Cabrera AA (2013) Making Function Modeling Practically Usable. *AI EDAM* 27:3301–3309.
[237] Townsend AM (2013) *Smart Cities: Big Data, Civic Hackers, and the Quest for a New Utopia*, WW Norton & Company.
[238] Tracy KW (2012) Mobile Application Development Experiences on Apples iOS and Android OS. *IEEE Potentials* 31(4):30–34.
[239] Tseng MM, Jiao J, Merchant ME (1996) Design for Mass Customization. *Annals of CIRP* 45(1):153–156.
[240] Tseng MM, Jiao RJ, Wang C (2010) Design for Mass Personalization. *Annals of CIRP* 59(1):175–178.
[241] Tseng M-L, Tan RR, Chiu ASF, et al (2018) Circular Economy Meets Industry 4.0: Can Big Data Drive Industrial Symbiosis? *Resources Conservation and Recycling* 131:146–147.
[242] Tuballa ML, Abundo ML (2016) A Review of the Development of Smart Grid Technologies. *Renewable and Sustainable Energy Reviews* 59(C):710–725.
[243] Ueda K, Vaario J, Ohkura K (1997) Modelling of Biological Manufacturing Systems for Dynamic Reconfiguration. *Annals of CIRP* 46(1):343–346.
[244] Uhlemann THJ, Lehmann C, Steinhilper R (2017) The Digital Twin: Realizing the Cyber-Physical Production System for Industry 4.0. *Procedia CIRP* 61:335–340.
[245] UK RAS, 2019, https://www.ukras.org/publications/white-papers/. (Accessed 15 April 2019).
[246] Umeda Y, Ishii M, Yoshioka M, et al (1996) Supporting Conceptual Design Based on the Function-Behavior-State Modeler. *AI EDAM* 10(4):275–288.
[247] Umeda Y, Tomiyama T, Yoshikawa H, Shimomura (1994) Using Functional Maintenance to Improve Fault Tolerance. *IEEE Expert-Intelligent Systems and Their Applications* 9(3):25–31.
[248] Ungurean I, Gaitan N-C, Gaitan VG (2014) An IoT Architecture for Things from Industrial Environment. *Proceedings of 10th IEEE International Conference on Communications* 1–4. http://dx.doi.org/10.1109/ICComm.2014.6866713.
[249] Valckenaers P, Van Brussel H (2005) Holonic Manufacturing Execution Systems. *Annals of CIRP* 54(1):427–432.
[250] Valencia A, Mugge R, Schoormans JPL, Schifferstein HNJ (2015) The Design of Smart Product-Service Systems (PSSs). *International Journal of Design* 9 (1):13–28.
[251] Van Brussel H, Bongaerts L, Wyns J, Valckenaers P, Van Ginderachter T (1999) A Conceptual Framework for Holonic Manufacturing: Identification of Manufacturing Holons. *Journal of Manufacturing Systems* 18(1):35–52.
[252] Van Brussel H, Wyns J, Valckenaers P, Bongaerts L, Peeters P (1998) Reference Architecture for Holonic Manufacturing Systems: PROSA. *Computers in Industry* 37(3):255–274.
[253] Van Der Auweraer H, Anthonis J, De Bruyne S, Leuridan J (2013) Virtual Engineering at Work: The Challenges for Designing Mechatronic Products. *Engineering with Computers* 29(3):389–408.
[254] Van der Auweraer H, Donders S, Desmet W (2018) Simulation and Digital Twin for Mechatronic Product Design. *Proceedings of 28th International Conference Noise and Vibration engineering (ISMA2018)*, , http://past.isma-isaac.be/isma2018/proceedings/.
[255] Eds.).Van de Laar P, Punter T, (Eds.) *Views on Evolvability of Embedded Systems*, Springer, NetherlandsISBN 978-90-481-9849-8.
[256] Van Eck ML, Sidorova N, Van Der Aalst WMP (2016) Enabling Process Mining on Sensor Data from Smart Products. *Proceedings of 2016 IEEE Tenth International Conference Research Challenges in Information Science (RCIS)* 1–12. http://dx.doi.org/10.1109/RCIS.2016.7549355.

[257] Vanhuyse J, Murgovski N, Nicolai M, et al (2017) Closed-Loop Design for Hybrid Powertrains. *Procedings of ASME IDETC 2017* 3. http://dx.doi.org/10.1115/DETC2017-67404. Paper No. DETC2017-67404.
[258] Verriet J, Basten T, Hamberg R, Reckers F, Somers L (2013) *Adaptivity in Professional Printing Systems, Model-Based Design of Adaptive Embedded Systems*, Springer: 1–9. http://dx.doi.org/10.1007/978-1-4614-4821-1.
[259] Vogel-Heuser B, Diedrich C, Pantförder D, Göhner P (2014) Coupling Heterogeneous Production Systems by a Multi-Agent Based Cyber-Physical Production System. *Proceedings of 2014 12th IEEE International Conference Industrial Informatics INDIN 2014* 713–719. http://dx.doi.org/10.1109/INDIN.2014.6945601.
[260] Volkswagen AG, 2017, https://www.volkswagenag.com/en/news/stories/2017/02/the-e-mobility-module.html#. (Accessed 15 April 2019).
[261] Wall J., Forget, L. 2017, What's new in QNX SDP7.0, https://zapdoc.tips/what-s-new-in-qnx-sdp-70.html. (Accessed 15 April 2019).
[262] Wan J, Canedo A, Faruque MAAl (2017) Functional Model-Based Design Methodology for Automotive Cyber-Physical Systems. *IEEE Systems Journal* 11(4):2028–2039.
[263] Wan J, Tang S, Shu Z, Li D, Wang S, Imran M, Vasilakos AV (2016) Software-Defined Industrial Internet of Things in the Context of Industry 4.0. *IEEE Sensors Journal* 16(20):7373–7380.
[264] Wang L, Törngren M, Onori M (2015) Current Status and Advancement of Cyber-Physical Systems in Manufacturing. *Journal of Manufacturing Systems* 37:517–527.
[265] Wang P, Gao RX, Yan R (2017) A Deep Learning-Based Approach to Material Removal Rate Prediction in Polishing. *Annals of CIRP* 66(1):429–432.
[266] Wang P, Liu H, Wang L, Gao RX (2018) Deep Learning-Based Human Motion Recognition for Predictive Context-Aware Human-Robot Collaboration. *Annals of CIRP* 67(1):17–20.
[267] Wang WC, Koren Y (2013) Design Principles of Scalable Reconfigurable Manufacturing Systems. *IFAC Proceedings* 46(9):1411–1416.
[268] Wang Y, Mo DY, Tseng MM (2018) Mapping Customer Needs to Design Parameters in the Front End of Product Design by Applying Deep Learning. *Annals of CIRP* 67(1):145–148.
[269] Want R (2006) An Introduction to RFID Technology. *IEEE Pervasive Computing* 5(1):25–33.
[270] Warnecke HJ (1993) *The Fractal Company, A Revolution in Corporate Culture*, Springer-Verlag.
[271] Waschneck B, Reichstaller A, Belzner L, et al (2018) Optimization of Global Production Scheduling With Deep Reinforcement Learning. *Procedia CIRP* 72:1264–1269.
[272] Wasson CS (2015) *System Engineering Analysis, Design, and Development: Concepts, Principles, and Practices*, John Wiley & Sons . ISBN: 978-1-118-44226-5.
[273] Wymore W (1993) *Model-Based Systems Engineering*, CRC Press . ISBN 9780849380129.
[274] Wei H, Cai Y, Li H, Li D, Wang T (2010) Sambot: A Self-Assembly Modular Robot for Swarm Robot. *Proceedings of 2010 IEEE International Conference Robotics and Automation* 66–71. http://dx.doi.org/10.1109/ROBOT.2010.5509214.
[275] Weimer D, Scholz-Reiter B, Shpitalni M (2016) Design of Deep Convolutional Neural Network Architectures for Automated Feature Extraction in Industrial Inspection. *Annals of CIRP* 65(1):417–420.
[276] Weiser M (1991) The Computer for the Twenty-First Century. *Scientific American* 265(3):94–104.
[277] Weiser M (1993) Ubiquitous Computing. *IEEE Computer* 26(10):71–72.
[278] Wen L, Gao L, Li X, Wang L, Zhu J (2018) A Jointed Signal Analysis and Convolutional Neural Network Method for Fault Diagnosis. *Procedia CIRP* 72:1084–1087.
[279] Wikipedia, https://en.wikipedia.org/wiki/Google_Traffic. (Accessed 15 April 2019).
[280] Wikipedia, https://en.wikipedia.org/wiki/History_of_autonomous_cars. (Accessed 15 April 2019).
[281] Wikipedia, https://en.wikipedia.org/wiki/History_of_the_electric_vehicle. (Accessed 15 April 2019).
[282] Wikipedia, https://en.wikipedia.org/wiki/Smart_lighting. (Accessed 15 April 2019).
[283] Wikipedia, https://en.wikipedia.org/wiki/Smart_thermostat. (Accessed 15 April 2019).
[284] Wikipedia, https://en.wikipedia.org/wiki/Tesla_Model_S. (Accessed 15 April 2019).
[285] Wikipedia, https://en.wikipedia.org/wiki/Toyota_Prius. (Accessed 15 April 2019).
[286] Wikipedia, https://en.wikipedia.org/wiki/Transportation_as_a_Service. (Accessed 15 April 2019).
[287] Xu LD, He W, Li S (2014) Internet of Things in Industries: A Survey. *IEEE Transactions on Industrial Informatics* 10(4):2233–2243.
[288] Yasa E, Kruth J-P, Deckers J (2011) Manufacturing by Combining Selective Laser Melting and Selective Laser Erosion/laser Re-Melting. *Annals of CIRP* 60 (1):263–266.
[289] Yasuura H, Kyung C-MLiu Y, Lin Y-L(2017), *Smart Sensors at the IoT Frontier*, SpringerISBN: 978-3-319-55344-3.
[290] Yatabe T, Hirose T, Tsugawa T (1991) Autonomous Navigation of the Vehicle With Vision System. *Journal Society of Instrument and Control Engineers* 30 (11):1014–1028. (in Japanese).
[291] Yim M, Duff DG, Roufas KD (2000) PolyBot: a Modular Reconfigurable Robot. *Proceedings of IEEE International Conference on Robotics and Automation* 1:514–520.
[292] Yim M, Shen W-M, Salemi B, et al (2007) Modular Self-Reconfigurable Robot Systems [Grand Challenges of Robotics]. *IEEE Robotics and Automation Magazine* 14(1):43–52.
[293] Yoon SC, Um J, Suh S-H, et al (2019) Smart Factory Information Service Bus (SIBUS) for Manufacturing Application: Requirement, Architecture and Implementation. *Journal of Intelligent Manufacturing* 30(1):363–382.
[294] Zanella A, Bui N, Castellani A, et al (2014) Internet of Things for Smart Cities. *IEEE Internet of Things Journal* 1(1):22–32.
[295] Zawadzki P, Zywicki K (2016) Smart Product Design and Production Control for Effective Mass Customization in the Industry 4.0 Concept. *Management and Production Engineering Review* 7(3):105–112.
[296] Zhang J, Wang P, Yan R, Gao RX (2018) Deep Learning for Improved System Remaining Life Prediction. *Procedia CIRP* 72:1033–1038.
[297] Zhang L (2015) Specification and Design Method for Big Data Driven Cyber Physical Systems. *Proceedings of 23rd International Conference on Systems Engineering* 1089:849–857. 978-3-319-08422-0_124.
[298] Zheng C, Le Duigou J, Peter H, Matthieu B, Eynard B (2016) Multidisciplinary Integration During Conceptual Design Process: A Survey on Design Methods of Cyber-Physical Systems. *Proceedings of 14th International Design Conference DS 84* 1625–1634.
[299] Zhong RY, Xu X, Wang L (2017) IoT-Enabled Smart Factory Visibility and Traceability Using Laser-Scanners. *Procedia Manufacturing* 10:1–14.
[300] Zhou G, Li F, Cheng H-M (2014) Progress in Flexible Lithium Batteries and Future Prospects. *Energy and Environmental Science* 7(4):1307–1338.
[301] Zojirushi, C http://www.mimamori.net (in Japanese) (Accessed 15 April 2019).
[302] Zuehlke D (2010) SmartFactory-Towards a Factory-of-Things. *Annual Reviews in Control* 34(1):129–138.