

Article

# Leveraging Circular Economy through a Methodology for Smart Service Systems Engineering

Friedrich A. Halstenberg <sup>1,\*</sup>, Kai Lindow <sup>1</sup> and Rainer Stark <sup>1,2</sup>

<sup>1</sup> Division Virtual Product Creation, Fraunhofer Institute for Production Systems and Design Technology, 10587 Berlin, Germany

<sup>2</sup> Department of Industrial Information Technology, Technische Universität Berlin, 10587 Berlin, Germany

\* Correspondence: [friedrich.halstenberg@ipk.fraunhofer.de](mailto:friedrich.halstenberg@ipk.fraunhofer.de)

Received: 19 April 2019; Accepted: 14 June 2019; Published: 27 June 2019



**Abstract:** Product Service Systems (PSS) and Smart Services are powerful means for deploying Circular Economy (CE) goals in industrial practices, through dematerialization, extension of product lifetime and efficiency increase by digitization. Within this article, approaches from PSS design, Smart Service design and Model-based Systems Engineering (MBSE) are combined to form a Methodology for Smart Service Architecture Definition (MESSIAH). First, analyses of present system modelling procedures and systems modelling notations in terms of their suitability for Smart Service development are presented. The results indicate that current notations and tools do not entirely fit the requirements of Smart Service development, but that they can be adapted in order to do so. The developed methodology includes a modelling language system, the MESSIAH Blueprinting framework, a systematic procedure and MESSIAH CE, which is specifically designed for addressing CE strategies and practices. The methodology was validated on the example of a Smart Sustainable Street Light System for Cycling Security (SHEILA). MESSIAH proved useful to help Smart Service design teams develop service-driven and robust Smart Services. By applying MESSIAH CE, a sustainable Smart Service, which addresses CE goals, has been developed.

**Keywords:** circular economy; product service systems; smart services; systems engineering; model-based systems engineering; product development; sustainability; engineering methods; service economy

## 1. Introduction

In the 21st century, humankind is increasingly confronted with the limits of the capacities of our planet; natural resources are finite and greenhouse gas emissions foster climate change [1]. In contrast to this, large amounts of "waste" are deposited, incinerated and released uncontrolled into our ecosystems [2]. In addition to the use of renewable energies, Circular Economy (CE) is one of the central paradigms for meeting the global challenge of sustainable development. CE requires that material be circulated rather than incinerated or added to landfill [3].

Climate change, material-consumption and energy-consumption are closely entangled. Climate change is strongly driven by the incineration of fossil fuels [4]. In various sectors, such as electricity and heat production, transportation, industry or waste processing, energy consumption is dependent on the manufacturing [5], consumption [6] and material recovery [7] of products. While manufacturing, consumption and material recovery heavily influence energy consumption, and consequently, climate change, they are also a main driver for resource consumption. Production requires energy and resources in order to transform raw materials into the desired form. Manufacturing processes can be highly energy-intensive [5]. Raw materials and auxiliary materials need to be sourced before they can be used in manufacturing. Depending on the type of product, energy and material is consumed in the use phase

as well. A substantial impact of an automobile on its CO<sub>2</sub>-balance, for example, is caused through fuel consumption in the use phase. Material recovery processes show significant differences within their levels of energy and resource-intensity. Recycling processes may require high temperatures, which subsequently result in a high energy demand [7]. Remanufacturing or an increased product and component lifetime may increase the material intensity of products or components without changing their original form or purpose. Remanufacturing processes still require energy and material, but the levels can be lower than for recycling [8].

The impact of a product on material and resource consumption can occur in various lifecycle phases. The complex interdependencies between material, energy and climate impact need to be taken into account in the design of products. Entanglements and rebound effects need to be considered as early as possible in product design in order to have the highest impact. Here, the interactions with other design goals such as functionality need to be considered.

In order to address the challenges of climate change and limited resources, a product must provide a maximum benefit for a maximum period of time, while optimal decisions regarding material and resources must be made. Decision-making in this context is challenging, since the interrelations between different lifecycle phases and CE-goals occur. Some materials are critical for a product functionality and cannot be replaced. In certain cases, using more material or more energy-intensive production processes result in a lower environmental impact in e.g., consumption or the product's end of life. Nevertheless, adequate CE strategies must be identified and chosen for products at an early stage of product development [9].

CE is evolving into a substantial factor in society, governments, industry and academia. On the one hand, business model innovation for CE becomes fundamental to sustaining companies' competitive advantages in this context [10]. On the other hand, new environmentally and socially sustainable business models are needed in order to transform economy and society. Resource constraints, as well as increasing volumes of waste and pollution, are likely to impose increasing threats to welfare and wellbeing of the broad population. Business models based on CE have the potential create jobs and reduce the rate of unemployment [11].

Addressing climate change and extensive resource consumption represent central requirements for the industry. At the same time, the way value is delivered to the customer changes. On the one hand, products are increasingly combined and offered with services [12], and on the other, they include an increasing number of electronic elements and software [13,14]. These developments have been described both as enablers for a sustainable development if deployed right [15,16], and as barriers when used poorly [15,17].

An increasing number of products are no longer sold individually, but combined with services creating Product Service Systems (PSS). A PSS can be defined as consisting of 'tangible products and intangible services designed and combined so that they jointly are capable of fulfilling specific customer needs' [18]. PSS has been a significant research topic for decades. Some authors have put economic aspects at the centre of their work [19]. Others consider PSS to be solutions that have the potential to achieve sustainability improvements [20–22] and some even define PSS as offering models with a much lower environmental impact than traditional business models [23]. The relevance of PSS for the formation of CE manifests itself in an increasing number of validated industrial applications [24] as well as extensive subsidy programs. Nevertheless, PSS is not a guarantee for improving sustainability. For example, Tukker states that for many successful PSSs it is not clear whether they are sustainable. Furthermore, many purpose-designed sustainable PSSs have failed [14].

While products are increasingly offered with service components, the digital revolution is making an increasing impact on products and services at the same time. Industry 4.0 is characterized by the manufacturing of individualized products up to batch size one under the conditions of highly flexible production and the development of procedures for self-optimization, self-configuration and self-diagnosis [25]. In this context, products are combined with communication and processing capabilities to become Smart Products. Smart Products comprise Cyber-Physical Systems (CPS) and

Cyber-Physical Production Systems (CPPS) with additional internet-based services, thus including certain means of intelligence and communication [26]. Additionally, big data is analysed, interpreted, linked and supplemented and in this way refined into Smart Data [27].

The on-going digitization of products and production systems is a challenge for product development processes. Nevertheless, complex mechatronic systems have been developed over decades and respective design techniques have come a long way. A large number of our current products incorporate mechanical (physical), electrical and electronic (E/E-components) as well as software [28]. While these systems have advantages in their functionality, the dangers of e-waste to a sustainable development have been highlighted by e.g., Chancerel and Rotter [29]. Furthermore, electronic components require substantial amounts of critical raw materials, which are finite and sourced under critical circumstances [30,31]. A large number of critical raw materials, especially rare earths (lanthanides plus scandium and yttrium) as well as tantalum, gallium and indium show total end-of-life recycling rates of less than 1% [32]. This adds additional pressure to the challenge of finite resources.

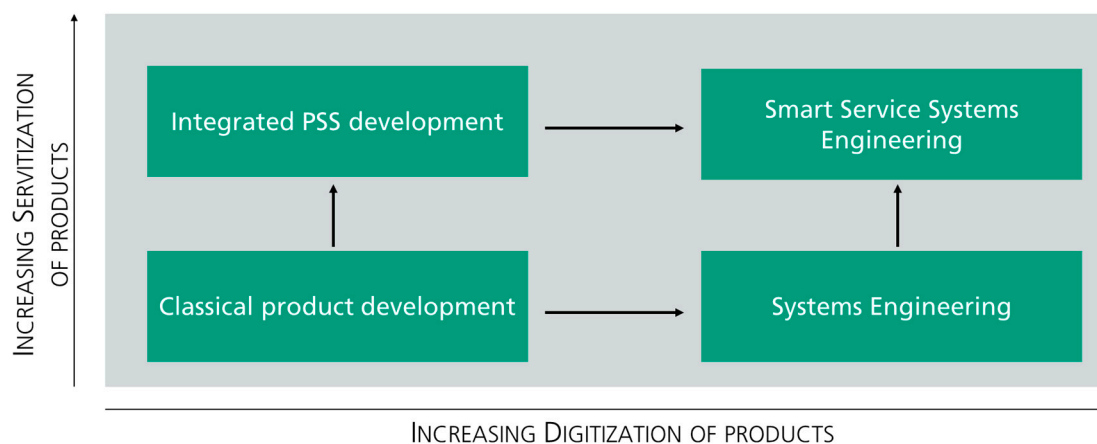
Complex mechatronic systems have to be developed in interdisciplinary teams. As a reaction to the increasing complexity in product development through the integration of different domains, the concepts of Systems Engineering (SE) and Model-based Systems Engineering (MBSE) have received considerable attention [33]. These disciplines are well developed, but focus rather on the development of technical systems and display a strong emphasis on the functionality of physical products and the interaction with software.

Servitization and Digitization transform traditional products into a new form of offering: Smart Services. Stark et al. have described Smart Services as “New digital data driven business models with high ambiguities around what and how to deliver new products to customers and users” [34]. The German National Academy of Science and Engineering (acatech) describes Smart Services as data-based services, which complement physical products and thus enable a flexible and individual orientation towards customer-specific requirements and needs [35]. Within the exploration field Smart Service Engineering of the Fraunhofer Institute for Production Systems and Design Technology, the following definition for Smart Services was developed building on the ideas previously mentioned: *Smart Services are platform-centred value creation systems that contain intelligent products and/or data-driven services and place the individual customer benefit at the centre of value creation.* This understanding is used throughout the scope of this article.

Smart Services combine Smart Products and data analytics capabilities with the service-orientation of PSS. In this context, Smart Services can be described as a specific manifestation of Product-Service Systems, which mainly refer to customer individual solutions based on the utilization of data derived from CPPS. In this regard, Smart Services utilize the capabilities of Smart Products and enable new business models and offerings for innovative digital value creation [5]. While the relevance of PSS for CE has been proven in a large number of industrial examples, information technology aspects also play a decisive role for CE. Data acquisition and analysis can be used to replace and simplify materials, logistics and physical maintenance work [3]. Nevertheless, a Smart Service is not sustainable by default. The production of vast amounts of IoT devices will account for a significant increase in energy and resource consumption. The dangers of e-waste to a sustainable development have been highlighted by e.g., Chancerel and Rotter [29]. It remains unclear whether the benefits of Smart Services outweigh the environmental costs, as long as these aspects are not considered and planned carefully. Developing integrated systems, which include physical components, E/E-components as well as service and software elements increases the complexity for systems design significantly. Developing these systems according to CE-goals requires robust engineering design methodologies in the early design phases.

Engineering disciplines have evolved from classical product development to PSS development on the one hand and to Systems Engineering on the other. The scientific community as well as industrial practitioners have presented a substantial number of methodologies and approaches for MBSE and PSS

development. Respective approaches for the development of Smart Services remain scarce. The authors propose the integration of PSS development and MBSE approaches in order to form methodologies and tools in the novel research field of Smart Service Systems Engineering (see Figure 1).



**Figure 1.** The evolution of product and system engineering disciplines.

MBSE represents a sophisticated approach for the integrated development of complex mechatronic systems and has been in development for several decades. Currently the service perspective is underrepresented in MBSE. The inclusion of service aspects in the procedures, models and tools of MBSE constitutes a next step towards the successful engineering of Smart Services. The central research question of this article can be formulated as: *How can the service perspective be integrated in current MBSE practices for the engineering design of Smart Services with the aim to address CE goals?* Within the scope of this paper and attempt to integrate service aspects in product and system engineering design practices is conducted. MBSE practices, in particular systems modelling methods and notations, are evaluated in order to address engineering design of technical systems and services.

## 2. Theoretical Background for the Development of Systems

The development of products has been in the focus of industry and academia for many decades. Product development is the cornerstone of a company's ability to innovate, compete and thus contributes significantly to ensuring its success [36,37]. The product development process describes the interdependent and often overlapping work steps during the design of a new product, from the idea to start of production [38]. A large number of generic and specific product development processes have been developed. Pahl and Beitz [39] provide a rough workflow for product development, which is staggered into clarification of the task, concept design, detailed design and elaboration. In addition, Ulrich and Eppinger [40] provide a design methodology for NPD (New Product Development) which covers the stages planning, concept development, system-level design, detail design, testing and refinement, and production ramp-up.

### 2.1. Systems Engineering & Model-Based Systems Engineering

Systems Engineering (SE) is a consistent, interdisciplinary approach to the development of multidisciplinary systems. It addresses not only the system to be developed, but also the associated project. It originates from system theory and has undergone continued further development. The catalyst has always been the increased complexity of problems [41]. The complexity can mainly be attributed to the diversity of the system elements. System engineering concentrates on the development of a system design in the early development phase [42]. The holistic way of thinking includes both technical and economic aspects and describes a concept from the production to the operating phase to the decommissioning of the system [43]. A system consists of several elements, which can themselves be systems and which have interfaces and interactions with each other. The basic objective is to

develop a common structure in order to align all system element. In order to fulfil this purpose, the interaction and the interfaces of all system elements must be determined. Therefore, the system elements are fragmented and structured hierarchically. This procedure starts with the perspective of the whole system before it is decomposed into parts and tasks. Then the individual parts are subject to decomposition again until respective business units can design them. This method allows to take the interactions between the individual elements into account [44].

Model Based Systems Engineering (MBSE) is aimed at developing a balanced system solution in response to diverse stakeholder needs [45]. While classical methodologies of systems engineering are paper- or document-based, MBSE is based on digital system models [41]. The methods of model-based systems engineering can help to describe a multidisciplinary product in an abstract way [46,47]. Accordingly, models can be either abstractions or representations of reality that facilitate the understanding of complexity. MBSE often uses repository-based multi-user modelling tools that provide an environment for defining and managing a precise and unique worldview of parts, the system, and their behaviours and interactions. This can be applied horizontally to support the SE lifecycle process and vertically from integration into the implementation disciplines [33]. The systems engineering process can be roughly divided in the RFLP framework, consisting of: Requirements Engineering (R), Functional Engineering (F), Logical Engineering and Prototyping (P) [48].

#### 2.1.1. Architectures in the Context of Systems Engineering

Eppinger defines the product architecture as the scheme according to which product components and functions are arranged into chunks or modules and by which they interact with each other [40]. The product architecture determines which function is to be performed by which function carrier (assembly, component) [49]. Lamm and Weilkiens speak in this context of the system architecture, which describes the combination of individual parts or groups and pursues the goal of optimizing the functional decomposition into components and their composition [50]. In the concept development phase, the functional requirements are converted into a physical product [51]. The product architecture determines the functions, properties, module structure and future variance. The goal of an optimally designed product architecture is to enable the reuse of components, to reduce the number of variants, to reduce assembly costs and product launch times, and to optimize supplier structures [51]. While the architecture is a standard model in software engineering, it has not been comprehensibly implemented into practice in the domain of classical mechanical product development. Often requirements are directly converted into domain models or even virtual or physical prototypes. This leads to coordination problems amongst different domains in the systems engineering process.

#### 2.1.2. Systems Modelling Procedures in the Context of MBSE

Systems modelling procedures aim to model a system step by step [52]. A big challenge consists in defining which information is modelled at which point in time and in what detail. The performance of the procedure should always be considered in proportion to its economic efficiency [53]. In Table 1, frequently used procedures are shown with their respective tools, languages and references.

**Table 1.** Overview MBSE Methods incl. Tools and Language.

No	Procedure	Tool	Language	Reference
1	Harmony SE	Rational Rhapsody	UML/SysML	[54]
2	OOSEM	-	SysML	[55]
3	RUP SE	IBM Rational Developer	UML/SysML	[56]
4	STRATA	CORE	SDL	[57]
5	SYSMOD	Cameo System Modeller	SysML/SYSMOD	[58]
6	CONSENS	iQUAVIS	SysML	[59]
7	ARCADIA	Capella	DSML	[60]
8	mecPro2	Cameo System Modeller	SysML	[61]
9	ALT	-	SysML	[62]

Harmony-SE uses a service-request-driven modelling approach modelled with SysML diagrams. Harmony Systems Engineering is a subset of a larger integrated system and software development process known as the Harmony Process, which is performed in eight steps. Harmony is based on the classical systems engineering V-model. The process assumes that model and requirements artefacts are managed in a centralized model, which is why the process can be implemented in the Rational Rhapsody tool [54]. The Object-Oriented System Engineering Method (OOSEM) was published by the INCOSE Initiative in November 2000. The special feature of OOSEM is that it is a tool-independent procedure, which is why a method-free tool is required for modelling. The mandatory modelling language is SysML. OOSEM follows a top-down approach and uses SysML to specify the analysis, design and verification of systems [55].

The Rational Unified Process for Systems Engineering (RUP SE) is a procedure that is both a process framework and a tool from IBM Rational and supports the development of large scalable systems including software, hardware and information. The procedure's aim is to help companies save time, reduce costs, reduce risks and improve the quality of the systems they build. The mandatory modelling language is now UML & SysML [56]. Vitech Corporation developed a procedure called STRATA, which can be integrated with the modelling tool CORE. The procedure is based on four primary SE activities that are linked and managed through a common system design archive. Each of these primary SE activities is linked to the corresponding "domains". The special feature of the STRATA procedure is that a System Definition Language (SDL) is used to manage model artefacts [57].

The System Modelling Method (SYSMOD) was designed by Weilkiens [58]. The procedure can be modelled with all SysML based tools. Cameo Systems Modeller offers a link in which only necessary SYSMOD elements are represented, whereby the language range is less user-friendly. The general procedure starts with the system idea, then follows the external view (black box) of the system, describing the context and finally the illumination of the internal system view (white box). The procedure is to be seen as a toolbox, using only the appropriate parts required for the system. The concrete procedure of the method can be described in 19 profiles by selecting those profiles that are relevant for the respective system. This depends on the intensity level selected above and whether modelling results are already available or not required at all [58]. iQUAVIS is a modelling software for product development, which is linked to the CONSENS procedure. iQUAVIS is not subject to any special modelling language. Only the use case diagram and the block definition diagram originate from SysML. iQUAVIS uses five different diagrams for modelling. Requirements, environment, application scenarios, functions, effect structure, form and behaviour are among the procedure steps. The aspects requirements, processes, resources and design of the production system are also taken into account, describing the principle of solving a production system [59].

The Architecture Analysis & Design Integrated Approach (ARCADIA) was developed by Thales and the PolarSys organization in cooperation with the Eclipse Foundation Group. Arcadia stands for a model-based construction method for designing systems, hardware and software. The procedure follows an architecture-oriented approach to analyze operational requirements and to structure and disassemble system components. This provides important information for decision-makers, allows a

holistic understanding of complex systems and makes the planning phase more efficient. The Arcadia procedure is fully dependent on the modelling tool Capella, for which a special modelling language has been developed, thus combining all three pillars of MBSE [60].

mecPro is a RFLP-based procedure. With regard to structure, mechatronic systems are based on four artefacts, requirements (R), functions (F), logical system architecture (L) and physical system architecture (P). The RFLP artefacts can be transferred to the left wing of the V-model for system modelling purposes [61]. The ALT procedure describes a functional and technical development. It is tool-independent and can be modelled with SysML. The modelling takes place in two architectures: technical-physical and the effect chain architecture. The technical-physical view contains the physical system elements and their assembly without considering the functional relationships of the components. The architecture has rather a horizontal arrangement over the first two layers. The chain-of-effect architecture, on the other hand, passes vertically through all the layers, which means that it deals with the functional relationships. The software components are modelled within the effect chain architecture with the help of signal and actuator chains [62].

### 2.1.3. System and Process Modelling Notations

A modelling language or notation is understood within this research as any artificial language that can be used to express information or knowledge or systems in a structure that is defined by a consistent set of rules. The rules are used for interpretation of the meaning of components in the structure. Modelling languages define the structure according to which conceptual models have to be designed. These languages can be formally described by meta models, which represent the language concepts and their graphical representation. The use of modelling languages can raise the process efficiency of conceptual modelling. They can also enhance the fluency of conceptual models which specify formalized, understandable and unambiguous constructs [63]. A substantial number of different modelling languages and notations have been developed for various purposes. In the following paragraphs, an overview of relevant modelling languages and notation for the fields of modelling PSS is given.

The Business Process Model and Notation (BPMN) specification provides a graphical notation for specifying business processes in a Business Process Diagram. Its goal is to support Business Process Modelling by providing a standard notation that is comprehensible to business users and yet represent complex process semantics for technical users. BPMN has become the de-facto standard for business processes diagrams. It is intended to be used directly by the stakeholders who design, manage and realize business processes, but at the same time be precise enough to allow BPMN diagrams to be translated into software process components. BPMN has an easy-to-use flowchart-like notation that is independent of any particular implementation environment [64].

Service blueprinting is a modelling notation specifically designed for service modelling. It consists of a detailed description of a service and precisely documents and describes the service, to communicate its details to the involved parties like providers, retailers and managers. This provides a background and defines what the service is and helps as a guide taking decisions regarding the service. The objective of blueprinting is to map the functions and sub-functions of the service, also inputs, outputs, activities and roles or people involved [65]. The components shown in an Integrated Definition 0 (IDEF0) model can represent people, information, software, functions, equipment, products, and raw materials. The objective of the model is to depict the component integration as well as the operation and association of the functions. The structure of the model consists of hierarchical diagrams. These provide different levels of detail in order to describe the functions and related interfaces along the system [66]. The Unified modelling language (UML) is an integrated notation intended to design, analyze, and implement software-based systems, modelling business processes and processes in general. The main goal is to facilitate an interoperable visual object-modelling tool. The structure of a UML model has three main categories of elements. These elements are used to describe different individual parts within the system. The categories are classifiers, events and behaviours [67].

SysML is intended to support the specification, analysis, design, verification and validation of complex systems. These can be hardware, software, information, processes, people, facilities and combinations. The objective of this language is to unify the modelling languages used in the SE community under a common one [68]. SysML reuses a subset of UML 2.5 and supports the application of model-based system engineering [47]. This language provides a simple construct for modelling SE problems. It is useful to specify: requirements, structures, behaviour, relations, and constraints on system properties [68]. Lindow et al. present the Data-flow Architecture. This data-oriented architecture is applicable throughout the entire product lifecycle. It can be used to represent actual processes and to derive optimization potential. The added value is depicted in cooperation with all relevant stakeholders, taking into account the value-adding activities, executing persons, tools used and models or data generated in the process creation [69].

Each modelling notation was developed for a specific set of reasons and has its advantages and disadvantages. BPMN is specifically designed for business processes. Through its very wide diffusion in industry, it is well-known amongst industry practitioners. This is a concise advantage since models in product development have to be understood quickly by different stakeholders within the organization, including middle and upper management. A draw-back for BPMN is the focus on sequences and processes. The notation cannot express technical structures of higher complexity. Similarly, service blueprinting is oriented towards services. Here, the sequence-based model is dominant as well and the notation is not able to express complex systems or software architectures. An assessment of Service Blueprinting was provided by Kazemzadeh et al [70]. IDEF0 on the other hand, was designed in order to depict organizational activities and decisions in a simple and quickly manner. Neither complex processes nor systems can be modelled in detail. UML provides an extensive array of different models for software modelling. Software architectures can be depicted very well, the sequence-based activity diagrams are simple and not suited for complexity. Similarly to UML, SysML provides a number of different diagrams for modelling an entire (mechatronic) system. The activity diagrams show very few modelling elements. The Data Flow Architecture was designed for developing and optimizing IT-landscapes. It can express IT systems, activities and data flows well. It was not designed for the development of products, which makes it difficult to apply in systems development. A detailed assessment of the suitability of integrated Smart Service modelling is presented in Section 4.2 of this article.

## 2.2. Circular Economy

Engineering Design significantly influences the future product and its characteristics regarding sustainability and CE specifically [71]. Engineering designers and engineering teams usually focus on product functionality primarily. In most cases, these teams have not been trained specifically in the disciplines of CE and are often unable to consider respective aspects. This chapter focuses on the methodological support for design teams regarding CE. At first CE frameworks and strategies are discussed. In the second section, concrete methods and tools are presented.

### 2.2.1. Frameworks and Strategies for Circular Economy

Ideas of CE have been based in different research fields such as the functional service economy [72], Cradle to Cradle design [73], Biomimicry [74], the Industrial Ecology [75], Natural Capitalism [76], and the Blue Economy systems approach [77]. According to the Ellen McArthur foundation, CE is an economy which is restorative and regenerative by design [24]. The European Union defines the following basic CE strategies: (1) product life extension through maintenance and repair, (2) reuse, (3) refurbishment (Remanufacturing & Refurbishing), (4) repurposing, and (5) recycling. These solutions are arranged from the most attractive to the most unattractive solution [9].

The Ellen McArthur foundation also provides the best-known framework for CE, naming sharing, repair/overhaul, reuse, remanufacturing and recycling as central CE strategies. Even though the model of the Ellen McArthur foundation has been accepted as a quasi-standard model for CE, it is not free



from criticism. Van den Berg and Bakker argue that the various interpretations of the terminologies used (reuse, remanufacturing etc.) can lead to misunderstandings [78]. According to Hansen and Schmidt, the cycle-oriented concept requires a stronger focus on services for maintenance, repair, reprocessing and recycling [79]. For this reason, an adapted CE product design model is presented by van den Berg and Bakker [78], which distinguishes between five essential features:

- **Future Proof / Sustainability:** reducing the need for new products, e.g. by developing durable products suitable for prolonged use
- **Disassembly:** non-destructive dismantling, destructive dismantling
- **Maintenance:** maintenance covers all aspects relating to the provision of services in order to extend the service life of products as much as possible. This includes cleaning, repair, upgrading and lifetime predictions. From a design point of view, optimal maintenance also includes designing a product with a lifetime prediction that can predict future product performance.
- **Remake / Redesign:** the redesign represents the prolonged use of components and consists of all actions performed when the customer returns a product. Product design for reprocessing is made possible by business models and can be viewed on two levels [80]:
  - Business model and product strategy: including sales, marketing, service support, and reverse logistics.
  - Detailed product design and engineering: including core collection and functional design.
- **Recycling:** end-of-life material recovery

Rose et al. as well as Gehin et al. highlight that basic End-of-Life strategies such as reuse, reprocessing and recycling should ideally be used as a hybrid form [81,82]. Recycling can take place with or without dismantling. In addition, disposal and service are also considered as important End-of-Life (EoL) strategies [81]. The study by Ghisellini et al. provides a comprehensive overview of the literature over the last two decades with the aim of covering the most important CE characteristics and perspectives, including origins, basic principles, advantages and disadvantages, modelling and implementation of CE at different levels (micro, meso and macro). It is noted that the implementation of CE worldwide is still in an early stage and is mainly focused on recycling and not on reuse. Experience has shown that the transition to CE depends on the involvement of all actors in society and their capacity, as well as on appropriate forms of cooperation and exchange [83].

Based on the idea of a comprehensive CE framework, an implementation strategy is proposed by Lieder and Rashid using both a top-down and a bottom-up approach. In combination with the strategy of modularisation and designing products with less material, this leads to increased material efficiency [84]. Four important strategies to reduce material demand through material efficiency are discussed by Allwood et al.: products that are more durable, modularization and remanufacturing, component reuse, design of products with less material. In industrialized countries, these strategies have received little attention due to the economic, regulatory and social barriers that have been studied. However, the findings from waste management and the quest for energy efficiency suggest that these barriers could be overcome. The authors outline possible mechanisms for change [85].

Well-developed supporting methodologies for CE exist. Life Cycle Assessment aims for creating a complete balance of a product's environmental impact [86]. When conducted correctly and with robust data, it is a precise tool for assessing the ecological impact of product or system. Within the last years, efforts have been made in order to expand the purely ecological scope of the LCA in order to include the societal impact of products and systems into the assessment. While Social Life Cycle Assessment focuses on societal aspects [87], Finkbeiner et al. present Life Cycle Sustainability Assessment, taking into account all three elements of the triple bottom line (ecology, economy and society) [88]. Material Flow Analysis (MFA) aims for quantifying material flows in a given system [89]. While LCA focuses rather on products, MFA was originally intended for process analysis.

### 2.2.2. Design for Circular Economy

The research field of Design for CE focuses on the development of methods and tools, which help design teams in addressing the resulting complexity. Various approaches, tools and methods addressing this issue have been described in the scientific literature and are summarized within the following paragraphs. Different methods are used to select a suitable strategy. The End-of-life Design advisor (ELDA) is a software that first evaluates technical product characteristics and then classifies the products according to appropriate EoL strategies [90]. End-of-life improvements are achieved either through design innovation or by improving the value chain. By better understanding the product, the proposed methodology can help the respective company in systematically developing appropriate and profitable end-of-life strategies for its unique position [81]. Repro2 (remanufacturing product profiles) was developed by Rose. Here, different situations are characterized and evaluated according to selected criteria in order to finally generate profiles for situation groups [82]. Tools such as design for environment checklists, environmental benchmarking and product passports can be integrated [81,84]. Design-for-Environment checklists are a helpful tool as a guide for material considerations. Furthermore, environmental benchmarking is regarded as an effective technical and environmental tool for comparing products with similar functions or market segments [81]. A product passport serves to increase material efficiency. It is a collection of information on inherent product materials and components and related alternatives to dismantling and recycling [84]. Furthermore, in the context of product end-of-life management, the product can be evaluated in an appropriate manner with the help of data from downstream life cycle phases [91]. Early eco-design tools and decision support tools are identified as a key strategy for the future, where sustainable thinking can also be combined with traditional design methods.

Romme and Endenburg propose a science-based organisational design that uses construction principles and design rules as guidelines for the application of academic projects. They conclude that design principles are important in a circular design process to create new design rules. In addition, a deeper understanding of the systems and practices that emerge from these rules can be developed. Furthermore, explicit principles and rules for organizational design seem to facilitate the transfer of information between different projects [92]. Vanegas et al. propose the method "eDiM" (Measure of Disassembly Metric) to calculate the disassembly time based on the Maynard Operation Sequence Technic (MOST). In eDiM, a simple calculation scheme is used to calculate the disassembly time based on the sequence of actions and basic product information. This makes the results uniquely fully verifiable, making eDiM suitable for policy action as opposed to the results of previously developed methods. eDiM categorizes disassembly tasks into six categories that are suitable for providing insight into which disassembly tasks are most time-consuming and how product design can be improved [93].

Knapp et al. describe the development of various quantitative and qualitative methods for estimating pollutant transfer in recycling processes. The authors developed an evaluation matrix, which enables the qualitative risk potential to be evaluated. The assessment matrix is suitable for quickly reviewing waste streams used for recycling and assessing their potential environmental and health risks [94]. Sultan et al. have developed a new approach to prioritising the recycling of products based on a recycling desirability index [95]. Aguiar et al. propose a diagnostic tool to be used during the product design phase. It evaluates the recyclability of products and serves as a supporting tool for designer decisions [96]. Mangun and Thurston have developed a model for considering long-term planning while reusing components in product design. The model uses a product portfolio approach based on market segmentation rather than a single product [97].

Ceha not only considers product design, but also identifies uncertainty as the main reason for companies' distrust of the environmental service branch. The author describes three main building blocks (partnerships, value contribution and revenue stream) which are not limited to a unique business model prototype and should be a strong focus for any company that wants to work towards CE [98]. Asif et al. created a simulation model based on System Dynamics (SD) and Agent Based (AB). The multi-method simulation technique is capable of considering the mutual interactions between critical

factors of the business model, the product design and the supply chain when designing a dynamic simulation model [99]. Kjaer et al. present a two-step framework that aims to support analyses of PSS and their potential to lead to absolute resource decoupling [100].

Moreno et al. present a concise analysis of the literature regarding Design for Sustainability (DfS) and circular business models. The authors synthesize categorizations of the respective research fields to form a conceptual framework for Circular Product Design [101]. Wastling et al. present work on Design for Circular Behaviour. Here, key user behaviours are analysed in terms of their requirement for circular behaviour. A theoretical framework for the design of products and services is presented by the authors [102]. Okorie et al. have conducted a Systematic Literature Review on data-driven approaches for CE. The findings indicate that research on digital technologies enabled CE remains scarce [103]. Bocken et al. set the focus on product design and business model strategies for CE. They have developed a framework of strategies for companies that want to move towards CE [104]. Den Hollander et al. argue that there is a fundamental distinction to be made in between eco-design and Circular Product Design. A new set of concepts and definition is presented and a typology of approaches for Design of Product Integrity is developed. This typology may lead to a deeper understanding of CE as a concept [105]. Lewandowski highlights the industrial benefits of CE and the growing interest by major global companies. The author analysed the traditional business model canvas and added two new components leading to the conceptualization of the circular business model canvas [106]. Linder & Williander argue that we do not see a widespread adoption of circular business models in industry. They present a hypothesis-testing framework of business model innovation and show that circular business models imply substantial challenges to proactive uncertainty reduction for entrepreneurs [107].

### 2.3. Development of Product-Service Systems

The PSS lifecycle can be divided roughly in five phases: PSS planning, PSS development, PSS implementation, PSS operation and PSS dissolution or End of Life (EOL) [108]. The majority of product and PSS properties are set in the PSS development phase. Systematic engineering design methodologies provide support for exploring “solution fields” for design tasks. Here, different solution variants must be identified and selected on a conceptual level for further concretization. “It must be assumed that the often cited benefits of PSS over traditional sales, particularly ecological ones, will not occur by themselves, but have to be designed into the PSS” [109]. Dealing with complexity and uncertainty at early stages of product development is challenging. Similar to Design for CE, designing products and services in an integrated manner adds new layers of complexity to the tasks of engineering teams. Frameworks for systematic engineering of PSS can help to deal with complexity and effectively support design teams. Along the different phases of PSS development, different methods have been developed [110]. In the following paragraphs a brief overview is given:

Müller et al. have introduced a generic PSS development process model. It constitutes an extension of the classical V-model utilized in Systems Engineering (SE) and divides the PSS development process in separate PSS lifecycle phases [111]. Halen et al. released a PSS development guideline that implements a modular PSS design toolset and application sequence for the respective tools [112]. To support the PSS design process, Muto et al. proposes a PSS design policy based on Software Engineering Methods and Theory (SEMAT). The proposed policy provides designers with the PSS design perspective, milestones through the design process, and the way to manage the design process.

The “PSS Layer method is a systematic, model driven approach to clarify elements of the PSS architecture to trace and generate new system level requirements”. It is intended to support thinking in terms of system lifecycle, architecture, customer value, and interdependencies of product-service. The layer method helps collect, recognize and employ unconstrained customer feedback on the system over the complete lifecycle [111]. According to Welp and Sadek 2008, the modelling of PSS concepts in the early development phase is crucial, especially in order to consider the partial substitution of

product and service artefacts. They propose an extension of the heterogeneous modelling approach in mechatronics and the PSS concept model [113].

Maussang et al., 2009 found that designers must carefully consider the interactions between the physical product and the technical services early in the design phase to develop a competitive PSS. The goal of their proposed methodology is to provide designers with technical specifications for the requirements of the entire system that are as precise as possible for the development of physical objects involved in these systems [114]. Müller introduced the PSS requirements checklist, a novel method for requirements driven engineering. The main intention of this method is to support the PSS requirements generation. The requirements checklist is generic and contains more than 100 entries, arranged across in six different cluster pairs. [111].

The methodological approach presented by Uhlmann and Bochnig 2013 to integrate the development of products and services is based on the modelling and visualization of the interdependencies between PSS elements. Crucial here is the PSS-CAD support system. Throughout the design and development process, this software module is responsible for creating, visualizing and managing the PSS files and models [115]. Alexopoulos et al. developed a multi-criteria resource planning methodology and a tool to optimize the production, delivery, and installation of IPSS [116]. Pezzotta et al 2016 provides a complete overview of the applicability of Service engineering methodology (SEEM) in an industrial context. SEEM aims to help companies adopt PSSs in their portfolio and suggests a structured decision-making process [117].

Since functions are playing a key role in the approach developed within this research, we would like to discuss the works from the PSS engineering and design community at this point. Tukker presents eight types of PSS business models. The functional result-oriented business model is characterized by a high (intangible) service content and low (tangible) product content. The customer agrees with the provider on the delivery of a result or function. The provider is free in choosing a way of delivering the service. In other works, the agreement is made on what is delivered and not on which means are employed to deliver it [15]. In contrast to this, the function in SE (see chapter 2.4) is understood similarly (as a certain result or purpose) but refers to a state in systems development. Consequently, functional engineering can be regarded as a particularly important step in the development of PSS, especially when they are result-oriented. Haber and Fargnoli present a method called Functional Engineered Product-Service System (FEPSS) which addresses this need [118]. The authors do not employ approaches from SE or MBSE in their method. Sun et al. argue that modularization plays a key role in PSS development to support individual design. Here, functional requirements of PSS can be identified and then classified into different clusters using a fuzzy clustering algorithm [119].

Joore and Brezet discuss the Multilevel Design Model (MDM) regarding PSS [120]. Fadeyi et al. have identified modular product design as a suitable product design strategy for improved remanufacturing PSS integration [121]. Service Blueprinting was introduced by Shostack. The modelling methodology was inspired by PERT (Program Evaluation and Review Technique), a technique that produces robust proposals for delivery delays in new development and manufacturing projects [65]. Its applicability to PSS has been validated [109,122]. Schendel et al. integrated functional structures, as they are known from the engineering design in the service blueprinting approach [109].

To support the PSS adaptation in the early design phase, Song and Sakao suggest a design framework that includes a design process. The proposed design framework is module-based and thus flexible according to the needs of the user [123]. A framework developed by Kuijken et al is premised on the assumptions that the customer value is included in the PSS. The framework builds on the idea that products and services differ in terms of the value created by the material elements and the interaction moments between manufacturers and customers [124]. For further details on model types, model relationships, conceptual illustrations, and other experiments, please refer to: Sakao and Shimomura [125], Sakao et al. [126], Tomiyama 2001 [127].

Smart Services can be described as a specific manifestation of PSS. Exner et al. give a concise overview of the relation of data-driven business models and Smart Services [128]. Within this paper,

we define Smart Services as “platform-centred value creation systems that contain intelligent products and/or data-driven services and place the individual customer benefit at the centre of value creation”. Methods specifically designed for the development of Smart Services remain scarce.

Exner et al. present a method to design Smart Services based on information categorization of industrial use cases. The DAU (Data – Application – Use cases) flow method is intended to support companies in their design process, especially to understand their current situation regarding the utilization of their existing and potential available data [128].

A large number of different methods and tools for the support of PSS design have been developed over the past decades. A gap in research exists in the support of conceptual design and at the architecture definition level in particular. In order to address this issue, Integrated Product and Service Architectures (IPSSAs) have been presented by Halstenberg and Stark in previous research [129]. This paper presented a first study on the applicability of certain modelling notations for expressing PSS and specifically the relationships in modelling products and services on the architecture level. The authors conducted a further study on the feasibility of modelling notations for expressing IPSSAs. Here, models of SysML were combined with BPMN models and elements of the data flow architecture.

### 3. Materials and Methods

In this chapter, the problem statement is presented, which specifies the concrete research gaps which are to be addressed. Based on the problem statement a solution approach was developed. This chapter concludes with the description of the applied research methodology.

#### 3.1. Problem Statement

Two central research gaps motivated the present article. These gaps were identified through the analysis of the state of the art as well as derived from experience of the authors in industrial practices. They are described in the following paragraphs:

##### 3.1.1. Research Gap I: Lack of Methodological Support for PSS and Smart Services in Concept Design

PSS Design describes the integrated development of products and services. It is a relevant research field and numerous methodologies and tools for the support of PSS development teams have been described in the academic literature. These methods, however, predominantly focus on the upper phases of the V-model. In conceptual design and specifically in the step "Structuring into realizable modules", little support has been provided so far. Especially, no equivalent model to the function and product structure (i.e. product architecture) on the service side currently exists.

The field of Smart Service Engineering extends the PSS concept to include the development of database-driven services. So far, such methods have only been researched sporadically. The integrated development of data-driven product and service services is not yet methodologically supported. Especially for the activity of *Partitioning and concept design for (integrated) sub-systems*, no adequate supporting methodologies for design teams are available.

##### 3.1.2. Research gap II: Lack of Suitable Design Methods for CE Focusing on the Conceptual Design Phase

Design for CE describes the development of products that address CE strategies. Numerous methods have been described within this discipline to support product development. Yet, the methods developed and described are poorly validated and tool-supported. A corresponding diffusion into industry is not yet discernible. In addition, methods and methodologies are poorly supported in architecture development to date. A support in the product architecture draft was not described so far. Circular design methods have been developed, but have not yet been diffused into practice. A formalization and systematization of the approach offers great potential, especially in the definition of product architecture. Additionally, methodologies which focus on the integrated development of PSS and Smart Services taking CE goals and strategies into account are currently missing.

### 3.1.3. Synthesis of Research Gaps

For each of the disciplines CE Design and PSS Design, numerous methods have been developed in the scientific literature. Product and system architectures represent a decisive advantage in the development of mechatronic products, but are not yet sufficiently used across disciplines. Design for CE methods have been developed, but have not yet been diffused into practice. For integrated PSS development, methodological support was provided for the design steps in the upper phases of the V-model. However, methodological support for circular PSS development in concept design and specifically in the step "structuring into realizable modules" has not been provided. The integrated development of Smart Services is not yet supported.

### 3.2. Scope of the Research

This research focuses on the integration of the service perspective in product and system engineering design practices under utilization of MBSE systems modelling methods and notations with the aim to better include CE strategies in Smart Services. The solutions developed in this article aim for application at the phases of functional and logical engineering within the SE process. In traditional product design this point in the product development process can be compared to the tasks of concept design and system-level design according to Eppinger [40] or conceptual design according to Pahl and Beitz [39]. These are comparably early phases in the product and system development process. Main goals and outcomes of these tasks are (1) gaining a better understanding of the system, (2) reaching a maturity level that is satisfactory for downstream development, (3) breaking down design activities, (4) definition of work packages, (5) forming system modules and PSS interfaces.

We emphasize here, that at this point the principal solutions are defined, but no virtual or physical models are created yet. In later phases in the design process, such as detailed design, the solutions gain more maturity. This research focuses on methodology for architecture definition and respective support mechanisms such as notations and knowledge systems. The developed results should support design teams in leveraging CE goals through the general composition of the system. In this way, engineering design teams gain a better understanding of the system and its internal and external interaction. It provides a framework in which further methods should be applied in order to find ideal solutions. We do see a large potential for the methodologies of LCA, S-LCA, LCSA and MFA to be applied in addition. Depending on the maturity that the system components have, simplified or streamlined methods will have to be used. The concrete interaction of these methods will not be in the scope of this article, but will be investigated in further research. Strong efforts are currently made in order to integrate MBSE methods and tools in Product Data Management (PDM), Product Lifecycle Management (PLM) and Enterprise Resource Management (ERP) systems. Since the proposed solution in this research aims to use MBSE methods and notations, we also see potential to integrate the proposed solution in PDM/PLM and ERP systems. Nevertheless, the further development of the proposed methodology into a functional IT-tool and its potential integration in larger IT-systems is considered out of the scope of this research and will be subject to subsequent work. This paper focuses rather on the design of non-stationary goods such as PSS and Smart Services. The methodology could be expanded for the application on facilities and buildings as well. In this case an integration to Building Information Modelling (BIM) systems will also be of interest. Nevertheless, adaption to buildings and facilities is considered out of the scope of this article.

### 3.3. Solution Approach

Based on the problem statement, the conclusion was drawn that a novel methodology for the development of Smart Services, which addresses CE strategies has the potential to fit the identified research gaps. The authors consider it important to address CE strategies in a methodology, which also gives the user the opportunity to incorporate further, "regular" design goals. Therefore, a methodology for supporting a broader range of design goals was developed in a first step. For the specific task

of addressing CE strategies, an specifically adapted methodology was developed. For both of these methods, the authors followed the hypothesis that MBSE procedures, notations and tools provide a suitable basis. The research field of MBSE has been receiving substantial attention for several decades and respective MBSE methodologies have been refined and proven useful through both research and industrial practice.

### 3.4. Research Methodology

Following the hypothesis, MBSE procedures and notations have been analysed according to defined criteria. As a scientific method for this analysis, the utility value analysis was chosen. As no quantifiable analysis method (e.g. net present value method, annuity method, internal rate of return method) can be used, an alternative option had to be chosen due to the absence of quantifiable conditions [130]. The utility value analysis, a scientific method for analysing several possible alternatives, was selected to ensure a factual comparison of the methods based on established evaluation criteria [131]. These are not quantifiable characteristics, but the qualitative criteria are considered in the utility value analysis. As a result, factors such as technical, ergonomic, communicative or organisational characteristics are included in the evaluation. Thus, the utility value analysis offers a profitability analysis, but rather under the aspect of functional utility [130]. This makes the comparability of non-economic but decision-relevant criteria more transparent [132].

Goal of these two studies was the identification of means to incorporate the service perspective into MBSE practices. Based on these two studies, requirements for the Methodology for Smart Service Architecture Definition (MESSIAH) were derived. In a next step, MESSIAH was designed in a prescriptive study. Based on the set of requirements, MESSIAH was developed in a creative process and several design workshops. It consists of four central elements: the MESSIAH Modelling Language System, the MESSIAH Blueprinting framework, the MESSIAH procedure and MESSIAH CE.

The MESSIAH Modelling Language System was developed first, since it is the logical backbone of the MBSE-inspired methodology. The MESSIAH Modelling language System is a framework consisting of several different models, tailored to the individual needs of the domains involved. The system allows expressing the layout and relationship of the different elements of Smart Services. The principle of Integrated Product and Service Architecture (IPSSAs) has been coined by the authors in previous work [129]. Within this research, IPSSAs were developed further in order to support MESSIAH. IPSSAs can be regarded as a combination of models, which are created with respective notations. The analysis of MBSE notations was used in order to determine requirements for IPSSAs. Based on these requirements, suitable models were combined by the authors using creative workshops and design sprints.

In a second step, the MESSIAH Blueprinting system was developed. It supports MESSIAH by facilitating access and utilization of models with the aim of making design knowledge on Smart Services more accessible for the industry. For creating the MESSIAH procedure, suitable steps were identified through the analysis of systems modelling procedures. Then they were combined and complemented with new steps designed by the authors. The last elements of MESSIAH constitutes MESSIAH CE. CE strategies needed to be addressed specifically by MESSIAH. It uses CE blueprints, which were identified by the analysis of CE literature and refined through application on case studies.

In a last step, MESSIAH and MESSIAH CE were validated by application to the example of a smart street lighting system, developed by Fraunhofer IPK and the Technische Universität Berlin.

## 4. Results

This chapter presents the results of the different studies conducted in the scope of this article. At first, two utility analyses are presented. They served to identify an understanding of the current systems modelling procedure and notations in MBSE as well as to derive requirements for the development of MESSIAH. In Section 4.3, the modelling language, blueprinting framework method

and MESSIAH CE are presented. This chapter closes with the validation of MESSIAH through the example of a smart street lighting system.

#### 4.1. Analysis of Systems Modelling Notations in MBSE

MBSE practices rely on diagrams and models, which are expressed through languages or notations. The first study conducted within this research is the utility-value analysis of MBSE languages and notations. Motivation was to gain knowledge on how current MBSE practices express the necessary information and specifically which notations they use in order to do so. After defining evaluation criteria, the weighting factors are determined. Here, it was of particular interest for the authors, which notations were able to express elements of classic product and systems architectures, and which ones were able to express business processes and software architectures.

The evaluation criteria for the analysis of notations were dependent on whether the notations were able to express: (1) Functional requirements, (2) non-functional requirements, (3) function structures (trees & nets), (4) functional sequences, (5) product structures, (6) business processes/data flow, (7) software architectures. The criteria were scored in percentages from 0% to 100%, depending on their ability to express the respective type of information.

The results (see Table 2) indicate that IDEF0 is only suitable for modelling simple processes without a great deal of detail. SysML is well suited for developing mechatronic products, but contains fewer functionalities for mapping processes or services. Especially the functionality for modelling function processes is not well developed within SysML. UML is well suited for developing software, but hardly contains any functionalities for mapping services. BPMN is very well suited for the representation of processes and services, but can represent neither systemic function and product structures, nor software functionalities.

**Table 2.** Utility value analysis for MBSE notations.

Assessment Criteria	Weighting	Data Flow Architecture							Petri Nets		Function Structure		OMEGA
		IDEF	UML	SysML	BPMN	eEPK	Mo2Go						
Functional requirements	12%	20%	80%	100%	0%	0%	0%	0%	0%	0%	0%	100%	
Non-functional requirements	5%	0%	0%	100%	0%	0%	0%	0%	0%	0%	0%	0%	
Function structures	10%	60%	80%	80%	0%	0%	0%	0%	0%	20%	60%	0%	
Function processes (sequential)	14%	40%	60%	20%	100%	60%	100%	80%	0%	0%	0%	100%	
Product structures	12%	20%	0%	60%	0%	0%	0%	0%	0%	20%	0%	0%	
Services	26%	40%	60%	60%	100%	60%	100%	40%	20%	0%	0%	100%	
Software	21%	40%	80%	60%	0%	0%	0%	0%	0%	0%	0%	0%	
Sum	100%	35%	59%	69%	40%	24%	40%	22%	5%	4%	6%	52%	

#### 4.2. Analysis of Systems Modelling Procedures in MBSE

The second study presented in this paper is the analysis of currently available systems modelling procedures. Within this analysis, systems modelling procedures were compared with each other and evaluated according to predefined criteria in order to collect important findings for the development of a new procedure for the modelling of Smart Services. As each individual criterion has a different significance for the achievement of an optimal systems modelling procedure, the analysis gained a stronger and more differentiated significance. This was followed by the evaluation of the systems modelling procedures; the evaluation scheme is usually expressed in percentages in order to guarantee easy traceability [4].



Then, a ranking of the systems modelling procedures was established, which results from the total score achieved. During the evaluation, the individual values achieved play an equally important role, as the weak points are identified in a more detailed analysis of the individual evaluation criteria [3]. This is why the results are presented and evaluated graphically at the end of the application of the procedures. The aim of the analysis was not to decide on one of the procedures in order to continue working with this one procedure, but rather to provide an overview of the strengths and weaknesses of the described procedure. In this way, potentials of the existing procedures can be utilized within the new smart service modelling procedure and possible weak points are analyzed more explicitly, in order to develop solutions for them. An assessment of systems modelling procedures via utility value analysis has not been performed in the scientific literature yet.

The following evaluation criteria have been chosen to evaluate the existing systems modelling procedures:

- User friendliness (Is the procedure easy to understand, learn and use for applicants with different backgrounds?)
- Inclusion of requirements (How good can the procedure process requirements?)
- Early service orientation (Does the procedure address the customer perspective in functional engineering?)
- Utilization of a function structure (Does the procedure model functions in a systemic diagram such as a function structure?)
- Capability for modelling hardware (Does the procedure include steps for physical product architecture definition?)
- Capability for modelling software (Does the procedure focus on software development sufficiently?)
- Capability for modelling services (Does the procedure focus on modelling services in logical engineering?)

Each of the evaluation criteria has certain characteristics that positively affect a model-based approach to smart services. However, it must be taken into account that the different characteristics have different effects on the achievement of objectives. The following scores were applied in the analysis:

- 0.5 points: for very slight positive effects
- 1 point: for slight positive effects
- 2 points: for strong positive effects
- 3 points: for very strong positive effects

The evaluation criteria are based on extensive literature research and modelling experience of the authors. The same applies to the scores awarded for the characteristics of the evaluation criteria. The following figure shows the results of the utility analysis. In the first column (A) of the respective systems modelling procedure, an evaluation of the fulfilment of the respective evaluation criterion was carried out. The evaluation scale ranges from zero to 6 points. In column B, the weighted evaluation can be seen.

The results of the study (please see Table 3) show that a number of effective procedures for MBSE have been developed. In terms of the envisioned procedure for Smart Service architecture definition, the current systems modelling procedures are only partly suitable. The highest scoring procedure identified is SYSMOD with 64%, followed by ARCADIA with 60% and Harmony with 54%. In terms of addressing the identified needs and respective criteria these values cannot be regarded as sufficient. Especially regarding the criteria *early service orientation* and *capability for modelling services*, the procedures scored particularly low. It can be concluded that current systems modelling procedures are not yet suitable for Smart Service architecture definition. Nevertheless, some of the procedures subject to this study include certain process steps and elements, which can be useful. In order to develop a procedure, which supports design teams in Smart Service architecture definition, some

of these elements can be used. Particularly regarding the *early service orientation* and *Capabilities for modelling services*, new procedure steps and supporting models should be developed and included in the new methodology.

**Table 3.** Utility-Value Analysis of MBSE procedures.

Assessment Criteria	Weight	Harmony	OOSEM	RUP SE	STRATA	SYSMOD, FAS, VAMOS	CONSENS	ARCADIA	mecPro2	ALT
User friendliness	14%	28	14	42	55.8	62.8	48.8	84	56	14
Inclusion of requirements	7%	28	28	28	42	42	28	42	42	14
Early service orientation	12%	23	23	23	12	23	12	23	23	0
Utilization of a function structure	16%	40.7	65	0	65	98	49	81	65	33
Capability for modelling hardware	7%	14	14	7	14	28	14	28	14	14
Capability for modelling software	21%	115.1	42	63	0	42	21	42	42	21
Capability for modelling services	23%	23	23	23	0	23	0	0	23	0
Total	100%	54%	42%	37%	38%	64%	34%	60%	53%	19%

#### 4.3. Development of the Methodology for Smart Service Architecture Definition

Addressing the above-mentioned research gaps and findings from the analyses of systems modelling procedures and notations, the authors developed a corresponding solution. The three pillars of MBSE are language, procedure and tool. Following this approach, the authors decided to focus this research on language and procedure first, since the implementation of a tool should be based on a functioning and validated procedure and language. As preliminary tools, simple and commercially available modelling tools such as Microsoft Power Point and Visio were used. The development of a tool, which supports the needs of the developed procedure and language better, will be subject to further research and development outside the scope of this article.

The solution developed within this research was named Methodology for Smart Service Architecture Definition (MESSIAH). It encompasses four main elements:

1. The **MESSIAH Modelling Language System**: A generic system of different diagrams with respective notations for the description of the essential elements of a Smart Service
2. The **MESSIAH Blueprinting Framework**: A knowledge system for preserving, accessing and making use of knowledge and valuable generic smart service blueprints
3. The **MESSIAH Procedure**: A systematic procedure which can be followed by design teams in order to realize development of Smart Services in the concept design phase
4. **MESSIAH CE**: A complementary element of the methodology, which can be used for addressing CE goals when designing Smart Services

The solution developed within this research is focused on the concept phase of Smart Service development. Here, the steps of functional engineering and logical engineering are addressed in particular. Requirements engineering has been considered as outside the scope of this methodology, since enough academic work has been centred on this phase.

##### 4.3.1. The MESSIAH Modelling Language System

The language modelling system presented in this sub-section is based on the rationale that a complex system (such as a PSS or a Smart Service) cannot be described through a single model in concept design. Different disciplines, roles and teams are involved in the development of such a

system. The system itself is composed of far too many components of different nature. Especially in the cases of PSS and Smart Services, the system elements can be physical carriers, digital solutions or services, amongst others. The diverse nature of these elements makes their representation in a single model challenging. Furthermore, people involved in Smart Service design assume diverse roles and different demands for information. In order to quickly understand, create and modify this information, a number of different models and respective diagrams are more suitable than a single one. Even though these individual models express different information in a certain way, they still refer to the same Smart Service to be designed.

The MESSIAH Modelling Language System has been developed to a concept stage within this research. It is envisioned to be realized in an IT-tool, in which all models are derived from one holistic Smart Service data model. Following the principle of Single Source of Truth (SSOT), the information is stored in one common database, where every data element is stored exactly once. The different models only reference these specific elements. Since the goals of the MESSIAH Modelling System aims to support design teams in developing Smart Services in a consistent way, cross- and tracelinking between different models, elements and design stages is essential. In this way, the interdependencies between different domains (e.g., mechanical design and service design) can be recorded and made comprehensible. Ultimately, the modelling system is designed to enhance the integrated understanding of the Smart Service in this early phase of development. This understanding, especially of the different design interdependencies, will eventually lead to a better Smart Service design. As described above, the MESSIAH system is based on the RFLP framework and focused on functional engineering and logical engineering.

The MESSIAH Modelling System consists of five individual models, which refer to the same Smart Service. Each of the models uses a specific modelling language. The five models were identified through the analysis of systems modelling procedures and notations (Sections 4.1 and 4.2). The dependencies of these five models are represented via tracelinks. Elements of every model are tracelinked to the elements of every other model, given the fact that respective dependencies or relations exist. In Figure 2 an overview of the MESSIAH Modelling System can be seen.

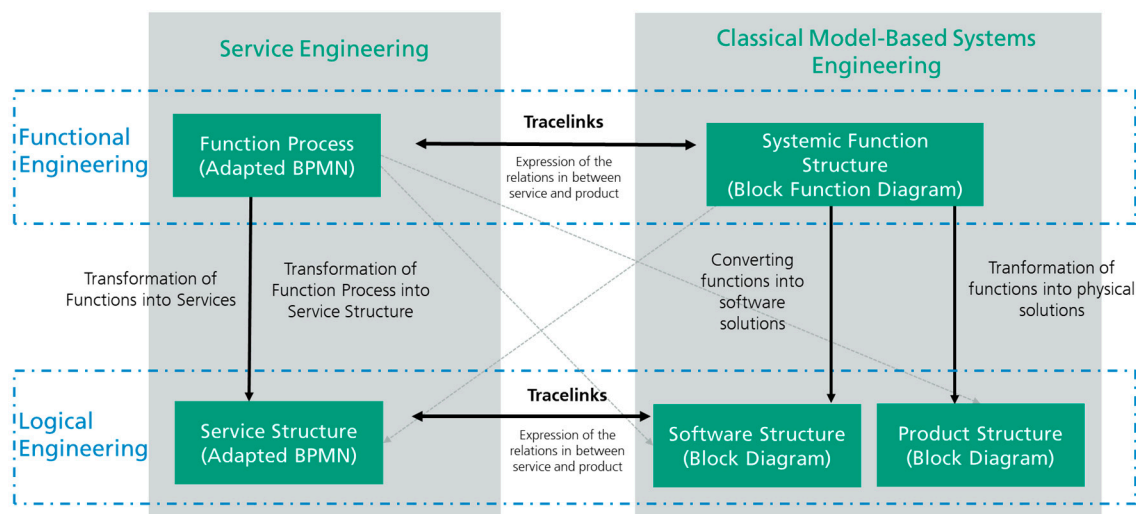
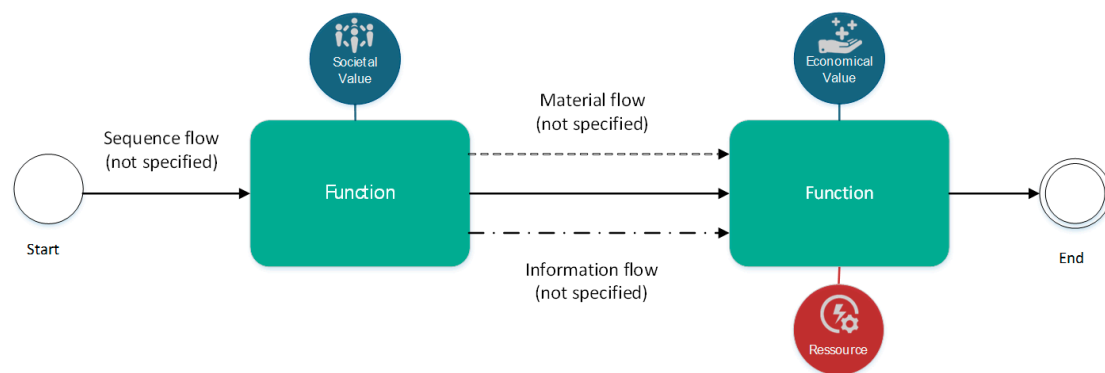


Figure 2. MESSIAH modelling language system.

Designing PSS and Smart Services requires a much more service-oriented approach than MBSE, which is centred on the function structure in functional design. For the function structure, a more systemic diagram, usually a block definition diagram, is used. It is suitable to depict functionalities from a systemic and technical point of view. We argue that a Smart Service should not be driven by technology. The design of Smart Services should be focused on the customer experience and should adapt technology in order to fulfil customer needs. Starting with the technological solution and trying

to sell it would not be suitable for a market characterized by increasing Servitisation and growing demands on service quality as well as user experience. The analysis of systems modelling procedures and notations identified that no suitable support for the service-perspective is included in current MBSE practices. As a reaction to this, two novel models with respective modelling languages have been developed and integrated into MESSIAH.

The first new model developed within the scope of this research is the Function Process. It is expressed notation for functional engineering in the MESSIAH Modelling Framework. We propose to start with this diagram, before a systemic, classical function structure is used. The Function Process focuses on the exact customer process first. It is sequence-based like most process models, which is more suitable for expressing the nature of a service and the activities of the customer. In Figure 3 the graphical representation of the Function Process can be seen.



**Figure 3.** Concept of the Function Process with modelling elements.

The Function Process represents a process-based view on the overall Smart Service System. Two general types of processes are represented. At first, the customer process is modelled, which aims to model the key activities that are conducted by the customer, when she or he uses the service. In many cases, there can also be more than one customer process, depending on how many concrete customer needs the system is intended to satisfy. Secondly, the core processes, which the organization conducts in order to fulfil the customer's needs, are modelled. These processes may or may not be visible to the customer. It is important that these processes remain on a functional level. No solutions are defined yet, since the Function Process is still used in functional engineering. The design team should keep the solution space open at this point in order to make better decisions when more information is available. The elements, which are modelled, are:

- Function (a natural purpose for fulfilling customer needs and requirements)
- Material flow (an indication of where material has to be exchanged. It does not necessarily have to be specified which material)
- Information flow (an indication of where information has to be exchanged. It does not necessarily have to be specified which information)
- Resource (an indication of where the organisation will have to deploy a resource. This resource can be of physical nature or require human work or capabilities)
- Economical value (the delivery of one of the central customer values, which can create economical value for the provider)
- Societal value (the delivery of value to the society, which is not inherent to the economic business models of the Smart Service)

The Function Process is designed in order to identify and specify the central value-adding activities of the Smart Service in a form where no concrete solutions are fixed yet. By utilizing it, the design team gains a better understanding of the environment of the system, the customer and the system itself. It helps to identify where resources are needed and where exactly the core value is delivered to the

customer. The model shows which functions are needed in order to fulfil customer needs. Additionally, a first understanding of how the company may deliver these services is gained. It provides an ideal base for designing a system that addresses customer needs and requirements in an optimal way.

For the systemic function structure, a traditional block definition diagram is used within the MESSIAH Modelling System. This diagram is designed for depicting systems in a structural way. It is suitable for planning the layout of the system. It can be used to gain a better understanding of how the system's elements are arranged and how the system may look.

On the logical engineering level, the Service Structure, the second novel diagram in MESSIAH, is introduced (see Figure 4). Similar to how the product structure refines the idea of the product in comparison to a representation via the function structure, the Service Structure specifies information from the Function Process. Within this process of specification, functions are transformed into activities. The function describes the fulfilment of a requirement without the specific solution. The activity is the specific solution for a function, which is fulfilled by a service. In order to specify the system further, the material and information flows are defined. This means a concrete type of material or information is assigned.

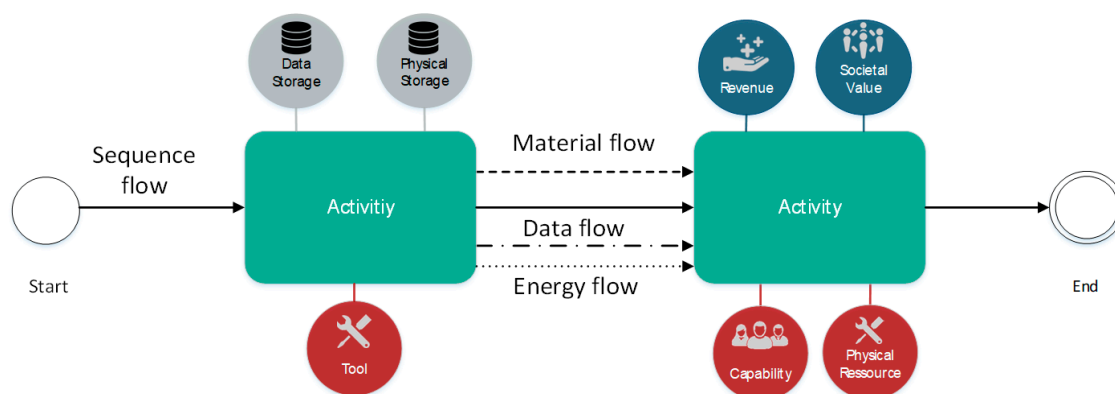


Figure 4. Concept of the Service Structure with modelling elements.

The Service Structure gives a concise overview of the services planned within the Smart Service in development. It represents the equivalent to the product structure and the software architecture on the logical engineering level. The modelling notation used here is an adapted BPMN with the elements:

- Activity (the specific process steps, which need to be carried out in order to provide the service)
- Data flow (a specification of data exchange within the delivery of the Smart Service)
- Material flow (a specification of material exchange of the Smart Service)
- Societal value (the delivery of value to the society, which is not inherent to the economic business models of the Smart Service)
- Revenue (an indication of exactly where revenue is created for the organisation)
- Tool (devices, means and IT-programs which are necessary in order to deliver the Smart Service)
- Capability (knowledge and competence required)
- Storage (facilities and means to store data, material or energy)

Resources are specified to tools, capabilities and physical resources. Tools represent all different IT- or physical aids, which are used for fulfilling certain tasks. Capabilities are the individual skills an organization needs in order to perform respective activities. Physical resources comprise raw materials and adjuvants (such as fuel of lubricant) which are needed in order to fulfil the service. It is important to identify these points in order to be able to know and plan how the organization will address requirements. Especially capabilities are a critical issue, since the lack of a capability can often not be resolved quickly. People with data analytics capabilities, for example, are currently rare. Given this, an organization wants to build up a service incorporating data analytics capabilities and is not

equipped with expertise in this field, will most likely not be able to deliver on these services quickly. Within the process of specifying the Service Structure, another specific focus put on data flow. Here, the information flows defined in the Function Process are transformed into data flows, including type of data and format. Since Smart Services per definition include databased functionalities and business models, understanding the nature of data flows within the system is crucial. While data storages depict needs for databases, physical storages show where it is required to store materials and resources.

Elements of the Service Structure are linked to the respective elements on the functional engineering level as well as to the elements of the other models on the logical engineering level. Tracelinking the models is important in order to describe the system from a service perspective. It is crucial to record all interdependencies in order to align the different parts of the smart service with one another. For expressing the physical components of the Smart Service, MESSIAH utilizes a classical product structure in a block definition diagram on the logical engineering level. Software elements are modelled with a software structure. Similar to the product structure, a block definition diagram is used here.

#### 4.3.2. The MESSIAH Blueprinting framework

A current challenge within the development of Smart Services consists in quick prototyping already in an early stage of Smart Service design such as concept design. In order to deal with this challenge, the MESSIAH Blueprinting Framework was developed within this research. Blueprints are generalized processes of a certain kind. They capture the knowledge collected on a certain subject and try to store it in an abstract yet comprehensible way. The blueprint needs to be as abstract as possible in order to make it suitable for the highest possible number of reutilization opportunities. It also needs to be comprehensible in order to transmit the knowledge in an optimal way. When external people can understand the blueprint and the information transmitted with it, chances are high that it will be used in further instances.

Two general types of blueprints are presented in this research: The Function Process Blueprint and the Service Structure Blueprint. When design teams start to develop one of the models, they can analyze the respective blueprints first in order to see how Function Processes and Service Structures with similar requirements and/or challenges have been modelled previously. When the respective model is developed further, the design can constantly be compared to the ones on the blueprint for validation, monitoring and failure detection.

A special focus is set within this paper on CE Blueprints. Since CE practices still did not diffuse into industry properly, distributing knowledge on CE practices into design processes can potentially have a big effect on industry and society. Within this research, a holistic CE Function Process was created. It is based on the analysis of different CE strategies described in chapter 2.2. Using the holistic CE Function Process, different CE strategies can be identified and prototyped at a very early age of smart service and concept design.

#### 4.3.3. The MESSIAH Procedure

The authors did not regard the development of a procedure solely for design for CE as useful for actual industrial practice. We strongly believe that design for CE always has to be viewed in the context of traditional product, PSS and Smart Service design. CE goals will always have to be aligned with traditional design goals (such as cost, quality and functionality related goals). Consequently, the authors chose to develop a procedure for PSS design in general, which allows taking into account diverse design goals first. A complementary methodology, MESSIAH CE addresses CE goals in particular and can easily be combined with MESSIAH. Within this chapter, the procedure for MESSIAH will be presented. In Section 4.3.4, MESSIAH CE is described. Table 4 shows the flow chart of MESSIAH as well as MESSIAH CE. MESSIAH starts after the requirements analysis, which was considered as outside the scope of this research. We consider the methodological support for PSS requirements analysis as sufficient (i.e., Müller has suggested the PSS requirements checklist, see also chapter 2.3).

MESSIAH is divided into three sections (functional analysis and modelling, creating logical architecture and performing structural analysis and optimization), which are each divided into several steps.

**Table 4.** Flowchart of MESSIAH and MESSIAH CE.

No.	MESSIAH	MESSIAH CE
0		Requirements analysis
1		Function analysis and modelling
1.1		Create function process
1.1.1	Analysis of requirements	Analysis of CE requirements
1.1.2	Analysis of Function Process Blueprints	Analysis of CE Function Process Blueprints
1.1.3	Transformation of requirements into functions	
1.1.4	Modelling the core business processes	Developing holistic CE strategy by using holistic CE blueprint
1.1.5	Modelling supporting business processes	Pre-defining sufficient CE processes for all components
1.2		Creating a Systemic Function Structure
1.2.1	Analysis of requirements	
1.2.2	Transformation of requirements into functions	
1.2.1	Arranging functions in a systemic function structure	
1.2.2	Modelling dependencies and interactions among the individual functional carriers	Modelling CE dependencies and interactions Design for disassembly
1.3		Set trace links between function process and systemic function structure
2		Creating Logical Function Architecture
2.1	Analysis of functions and assignation to solution	Preferring material-less solutions (software, service)
2.2		Creating a Logical Service Structure
2.2.1	Analysis of Service Structure Blueprints	Analysis of CE Service Structure Blueprints
2.2.2	Transformation of functions into services	
2.2.3	Transformation of Function Processes into Service Structures	Transformation of CE Function Processes into CE Service Structures; Closing material cycles in the models
2.3	Creating Product Structure	Aligning the Product Structure according to the necessities of the CE Service Structures
2.4		Creating Software Architecture
2.5		Linking the Services to corresponding functions/components
2.5		Tracelinking of Data, Services and Components
3.		Perform structural analysis and optimization (regarding CE and IoT)
3.1		Analysis of tracelinks between different models
3.2	Analysis along the process steps and structures	Analysis regarding the strategies
3.3		Analysis of Blueprints
3.4		Choosing optimal solutions for CE, taking into account strategies in step 3.2

The first section of MESSIAH is *Functional Analysis and Modelling*. Here, the Smart Service is outlined on a functional level. No solutions, such as physical components, specific services and software solutions are defined here. Requirements are transformed into functions and the general layout of the system is sketched. The section starts with task 1.1, the *Creation of the Function Process* (the description of the model can be found in Section 4.3.1). Within the Function Process, core business processes are modelled on a functional level. PSS and Smart Service Blueprints can be used in order to make use of experience from similar Function Processes and for rapid prototyping of the Smart Service. Requirements are analyzed, transformed into functions and integrated into the respective Function Process. After the core business processes are modelled on a functional level, the same is done for supporting business processes.

While the system functionality is modelled from a customer perspective in a sequential manner in the Function Process, the Systemic Functional Structure is created in order to outline the technical system. The Function Process is more suitable for expressing and understanding the customer and service perspective. The Systemic Function Structure on the other hand is used for a rather physically oriented view. Analogous to the procedure for the Function Process, requirements are transformed into functions while creating the Systemic Function Structure. Contrary to the Function Process, they are arranged systemically (and not sequentially). Especially modelling the interactions between different functional carriers is an important requirement fulfilled in this step. The Systemic Function Structure resembles the physical arrangement of the final system more and is rather interesting for physical elements of the system, compared to the Function Process. It can give better support for planning a systems layout and the arrangement of the systems elements. Dependencies between Function Process

and Systemic Function Structure are modelled via tracelinks. This is crucial, since MESSIAH aims at modelling one system and taking into account the interdependencies between product, service and IT. All different diagrams represent a certain view on the same system. In the end, an optimal system, which regards all different requirements and functions from different domains, should be created.

The second section of MESSIAH is *Creating the Logical Architecture*. Within this section, the Logical Service Structure is created first. Service Structure Blueprints are studied and analyzed. They are related to the Function Process Blueprints and can provide important sequences used and derived from other Smart Services. In the next steps, functions are converted into services and Function Processes into Service Structures by concretizing them. Following the modelling of the Service Structures, the further respective domain models, such as the product structure and the software structure, are created. While the product structure represents all physical carriers, software components of the Smart Service is depicted within the software structure. Both structures are derived both from Function Process and from Systemic Function Structure, by identifying and converting the respective functional carriers. Finally, the services are linked to physical components and software solutions as well to the respective functions. Tracelinks express the respective relationships.

In section 3 of MESSIAH, a *Structural Analysis and Optimization* is performed. Here, tracelinks between different models are analyzed. Then, the individual process steps and structures, as well as Blueprints, are analyzed for possible optimization. It is crucial within the MESSIAH procedure that the different views and tracelinks are considered and an ideal Smart Service is created via integrated optimization and alignment of the whole System.

#### 4.3.4. The MESSIAH Circular Economy Methodology

MESSIAH CE is specifically designed to be combined with MESSIAH and to address the integration of CE goals in the development of Smart Services. In certain steps of the MESSIAH procedure, MESSIAH CE complements MESSIAH with CE-relevant activities. For developing Smart Services, which address CE goals and requirements, the process steps of MESSIAH as well as the process steps of MESSIAH CE should be carried out.

In step 1.1.1, the analysis of requirements regarding CE is performed. The design team needs to clarify here if and how CE strategies will be regarded within the system design. It is also crucial to pinpoint which (economic) expenses the organization is willing to shoulder in order to achieve CE goals with their system. CE Function Process Blueprints are an essential part of MESSIAH CE. The blueprint library can be accessed for CE Function Process Blueprints, which describe previously used Function Processes. Through their analysis, means and effects can be identified as to how they were able to address CE goals and which strategies were used. In step 1.1.4 of MESSIAH, the core business processes of the system are modelled on a functional level. Here, the holistic CE blueprint can be used in order to develop a CE strategy for the Smart Service to be developed.

While the supporting business processes are modelled on a functional level, sufficient CE processes for all components need to be designed. This means that for every functional carrier, respective CE strategies have to be considered. Since no concrete solutions such as materials are chosen yet, the final CE strategy (such as recycling, remanufacturing etc.) does not have to be chosen. Nevertheless, possible CE strategies have to be identified, analyzed and pre-defined in order to eventually close all material cycles and maximize product lifetime. In step 1.2.2, where dependencies and interactions among the individual functional carriers are modelled, CE design goals are considered and integrated. A special focus is set on the interfaces between different carriers. Here, design for disassembly has to be considered in particular, due to the many implications on CE goals.

In section two, *Creating Logical Function Architecture*, material-less solutions such as software and services are preferred compared to material-intensive solutions, when functions are analyzed and assigned respective solutions. Similar to the analysis of CE Function Process Blueprints, CE Service Structure Blueprints are analyzed within MESSIAH CE. Especially, the CE Service Structure Blueprints, which have been derived in further examples from the same CE Function Process Blueprints as in



the current system development, are of particular interest. In step 2.2.3, CE Function Processes are transformed into CE Service Structures. At this point, the physical solutions are chosen and suitable CE-strategies must be chosen for every component of the product. Consequently, the required services for realizing these strategies are modelled (e.g., Services for recycling, remanufacturing, reverse logistics or maintenance). Here, measures should be planned to enhance product and component life as much as possible. More importantly, suitable end-of-life strategies should be chosen for every physical carrier. This step predominantly serves to develop the strategy of closing material cycles in the sense of CE. In step 2.3, the product structure is aligned according to the necessities of the CE Service Structures.

Section three of MESSIAH, *Perform structural analysis and optimization*, is of particular interest for the CE methodology. In step 3.2, Analysis along the process steps and structures, the whole system is analyzed regarding the strategies: Regenerate, share, optimize, loop, virtualize and exchange. In the step *integrated alignment and optimization*, optimal solutions for CE are chosen. Only here can the optimal solution be chosen since the whole system with its tracelinks and interdependencies has been modelled. This enables a holistic view of the system and facilitates optimization regarding the above-mentioned strategies.

## 5. Validation

After MESSIAH was developed in a prescriptive study, the authors chose a mode of validation for the methodology. Since the methodology is aimed at application in industrial practice, the authors considered validation by application on a practical example suitable.

### 5.1. Motivation for the Development of SHEILA

The development of the Smart Service used for validation in this chapter was motivated to increase the situation for cyclists. The cycling strategy of the city of Berlin envisages increasing the proportion of cycling to at least 18 to 20% of all routes by 2025, increasing the average distance travelled by 25% from 3.7 km to around 4.6 km and reducing the number of cyclists killed in road traffic by 40% [133]. Based on this motivation, a Smart Sustainable Street Light System for Cycling Security (SHEILA) was developed by the Fraunhofer Institute for Production Systems and Design Technology (IPK) in collaboration with students of the Technische Universität Berlin (TU Berlin).

### 5.2. Application of MESSIAH

SHEILA was developed using MESSIAH. In the following paragraphs, the description of the methodology application is illustrated. Before the methodology was applied, the requirements analysis was conducted using Müller's PSS checklist [111]. The requirement analysis is not described further here, since this section focuses on the validation of MESSIAH, which focuses on functional and logical engineering. The generic CE Function Process Blueprint was analysed in order to plan the CE strategy. Through the analysis, the CE strategies "long product life through improved maintenance activities", "recycling", "remanufacturing" and "reverse logistics" were chosen in order to close the cycles. In a next step, the Function Processes for the main customer-related services were created. This process triggered the need for a very early focus on the main customer groups and the value, which had to be delivered. The precise value of SHEILA had to be discussed among the design team, which led to a customer-oriented functional engineering and ultimately a customer-oriented final Smart Service. The list of requirements, which was created beforehand, had to be analysed and converted into functions. Based on this analysis, the main customer-related core processes were modelled within the Function Process modelling notation. In addition to close cooperation with the city, which also includes maintenance as part of the contract, three main services were decided as focal points of SHEILA. The first service was the analysis of non-personal data. In this way, workloads, movement patterns and potentials can be identified. This information is evaluated and stored and can be acquired by the city and included in traffic planning.

The second service is aimed at OEMs in the automotive industry who develop driver assistance systems. By providing a paid interface, turn-by-turn assistance systems can be supported and redundancy increased. The turn assist systems thus receive a signal when a cyclist approaches. In view of the obligation for new truck turn assist systems to be integrated and a possible obligation for retrofitting, many business opportunities exist within delivering this service. The third service is SHEILAs signalling service for cyclists. This service ought to minimize the time a cyclist has to stop at streetlights and regulate the cycling traffic to an ideal speed. The core customer-relevant Function Processes for these activities were modelled. This included the Function Process of the city using the data analytics service, the automobile turn assistant process and the process for the cyclist using the service. In Figure 5, the customer-relevant Function Process for the signalling service for cyclists is depicted exemplarily.

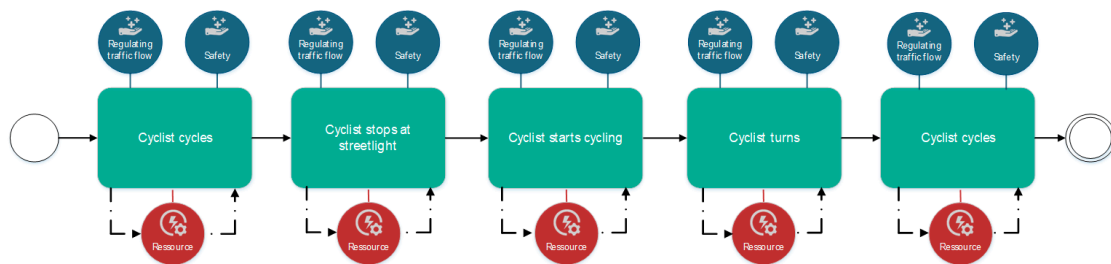


Figure 5. Function Process for the automobile turn assist system.

The model proved useful in order to better understand the activities of the client. It became apparent to the design team that SHEILA aims to provide the core customer values *regulating traffic flow* and *safety* in all of the five activity steps identified. According to the MESSIAH procedure, the resources were not specified yet, but it became apparent that specific resources were needed in all of the steps and an information flow between the resource and the customer was necessary in all steps. Consequently, the supporting business processes of SHEILA were modelled as Function Processes. Here, two Function Processes were modelled: The process for data acquisition und analytics and the process for signal provision to the bikers. In Figure 6, the business process for data analytics is shown exemplarily.

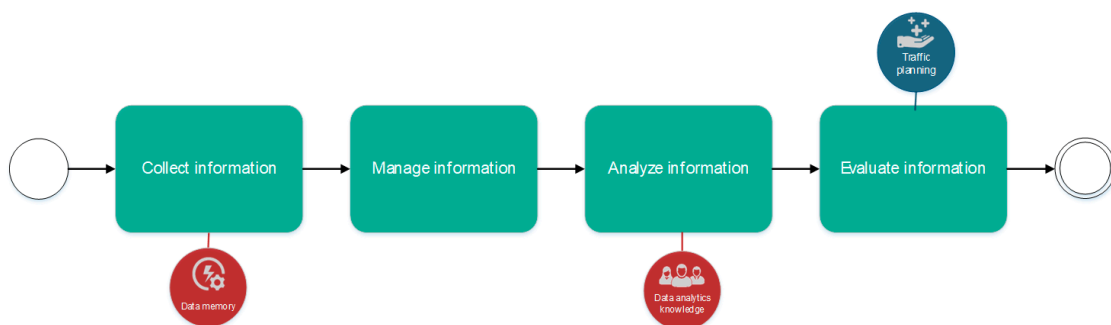
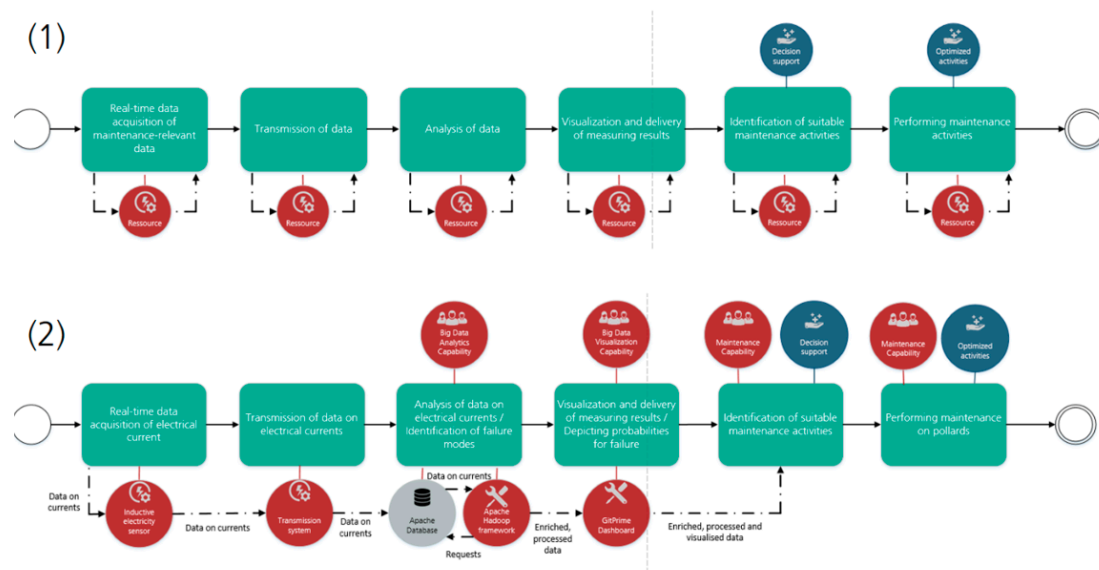


Figure 6. Supporting business Function Process for the service data analytics.

The Holistic CE Function Process was used in order to define CE strategies at an early stage. Since, the CE strategies “long product life through improved maintenance activities”, “recycling”, “remanufacturing” and “reverse logistics” had been chosen previously, respective CE Function Process Blueprints were drawn from the library, analyzed and adapted to SHEILA. In Figure 7 the Function Process for Predictive Maintenance can be seen.



**Figure 7.** CE Function Process (1) and CE Service Structure (2) for the process Predictive Maintenance for SHEILA.

In the next stage of MESSIAH, a systemic function structure was created and relationships were tracelinked to functions in the Function Processes. This way a rough outline of the system was created. Design for disassembly could be taken into account. In Section 2, creating the Logical Function Architecture, the system was analyzed in terms of deploying less material-intensive solutions. Based on the previous analysis of the system under development by creating Function Processes and structures, Service Structures were created. Here, the Function Processes were detailed and solutions were assigned. In this particular case Service Structure Blueprints were not available since the library was not built up enough. The project team created Service Structures, which model the respective services in detail. Functions were converted into concrete activities. Tools and capabilities were assigned. Service structures for the services data analytics, automotive turn assistant and the signalling service for cyclists were modelled. In Figure 7, the CE Service Structure for the process Predictive Maintenance can be seen.

Since solutions and materials were assigned, CE strategies for all components were chosen by developing respective Service Structures for all components, including “reserves logistics” processes. A product structure of the physical system was created in a next step. A software architecture was developed for the software components of SHEILA. Both domain models were created according to the necessities of the Service Structures. Services were linked to corresponding functions and components. In section 3 of MESSIAH, the system was analyzed and improved iteratively, taking into account all diagrams, functions, requirements and elements. This step proved to be useful within the development of SHEILA and contributed significantly to developing a customer-oriented, sustainable and less material-intensive solution.

### 5.3. Description of the Developed System

The SHEILA system is based on bollards, which combine lighting, guidance, safety and sustainability in road space. Dimmable LEDs integrated into the bollards illuminate the road space and the cycle path with directional light. The brightness value meets the requirements of the road traffic regulations to provide the road space with sufficient light. The cycle path is dynamically illuminated brighter in front of the cyclist, triggered by a motion sensor. In this way, frequent falls and accidents due to uneven ground and obstacles can be avoided and the visibility of cyclists can be increased. A guidance system also integrated into the bollard enables cyclists to move more efficiently and smoothly. This is achieved by visualizing the green phase on the bollard. A coloured LED in the

head of the bollard lights up green to indicate the current area of the green wave. If the cyclist moves in this area, he reaches the next green phase and can ride at a steady pace without stopping.

Safety is already increased by the physical protection of the bollards for cyclists. In addition, intersections and turning situations are made safer. The motion sensor installed on the bollards detects approaching cyclists. In order to draw the attention of turning vehicles to approaching cyclists, another bollard is placed at the intersection whose LED lights up orange when cyclists are detected at a relevant distance from the intersection.

Sustainability is promoted by three core elements. The lighting is implemented in such a way that the light emission is as low as possible. This is achieved by means of low light sources directed downwards. This is intended to reduce the impact of light emissions on birds, insects and residents. Increasing safety and driving comfort, however, also makes cycling much more attractive. According to studies, 40–50% of all car journeys in large German cities are under 5 km<sup>2</sup>. This illustrates the potential to promote more sustainable and environmentally friendly urban transport through improved cycling infrastructure. In terms of CE, End-of-Life strategies were identified for all components of the system. Furthermore, a predictive maintenance service framework was implemented, which maximizes the material intensity of the Smart Service.

## 6. Discussion

Within this article a novel methodology for the integrated development of Smart Services, MESSIAH, was presented. It encompasses a Modelling Language System, a Blueprinting Framework and a systematic procedure. For identifying requirements, two value-utility analyses were conducted. The results led the way for the development of the respective solution. The MESSIAH Modelling Language System presents a system of models with respective modelling notations.

With the help of these models, Smart Services can be described and modelled accurately. For the representation of service aspects, the authors present two novel models: The Function Process and the Service Structure. While the Function Process depicts services on a functional level, the service structure depicts the concrete services including activities, resources and roles. CE strategies can be addressed within these structures by planning advanced maintenance processes, logistics processes and assigning End-of-Life strategies. The MESSIAH Blueprinting Framework is a system for knowledge representation and access. Useful services are represented in blueprints and can be used as a basis for developing Smart Services. CE Function Process Blueprints deliver concrete mechanisms on how CE strategies can be deployed in Smart Services. The MESSIAH procedure gives Smart Service development teams a systematic procedure, which they can follow when designing Smart Services. MESSIAH CE addresses CE goals and strategies specifically in order to create sustainable Smart Services that follow the notion of CE. MESSIAH CE provides useful guidance for the development of CE strategies and the integration of CE design goals into Smart Service development. The methodology was validated through the application on SHEILA, a smart sustainable street lighting system. The design team concluded that MESSIAH assisted in the development of a robust Smart Service and in addressing CE goals and strategies.

The benefits of this study can be categorized in two main parts. One of them is the integration of systems modelling procedures and PSS/Smart Service methodologies. While MBSE methods provide sophisticated approaches for the development of technical systems, PSS/Smart Service methods address the service dimension better. The integration, which was conducted within this research and resulted in MESSIAH, can help Smart Service and PSS design teams to develop service-driven and robust Smart Services. The second benefit is the consideration of CE goals and strategies in the development of Smart Services and PSS, while still considering traditional design goals. The MESSIAH CE Add on has the potential to help in developing sustainable Smart Services, which make an impact on the transformation from linear economy to CE.

While interesting findings could be collected and the results hint to the above-mentioned benefits, this research remains limited in scope. The following points should be considered in order to assess the scientific value of the results:

(1) Shortcomings of the methodology: During the application several draw-backs of MESSIAH CE were observed. Since the methodology was not implemented in a respective tool, a lot of manual work in order to create the respective diagrams had to be deployed. Even though the application of MESSIAH CE led to a better understanding of the entire system in development, the process seemed slow due to extensive modelling activities. Especially the process of tracelinking the different diagrams was difficult and time-intensive. Since no system functionality supported the tracelinking process, the relationships had to be recorder in text processing programs and simplified diagrams. The results showed a lack of comprehensibility.

(2) Validity: MESSIAH has only been validated on one example so far. The long-term impact in terms of addressing CE goals still remains to be monitored, measured and assessed. The findings indicate that MESSIAH and the developed approaches are useful for creating Smart Services that address CE goals, but possible rebound effects might only be visible after a certain time. Furthermore, MESSIAH seemed to be useful on one certain example. In different fields of Smart Services this may not be the case. The methodology has yet to be evaluated on further examples in order to collect more severe evidence of substantial improvements. The result may be different here and hint to further improvement and adaptation. The Blueprint Library could only be filled with an initial number of blueprints. To serve as an effective system for accessing and transferring knowledge on Smart Service and CE processes, it will have to be filled with further blueprints.

(3) Suitability of MBSE practices for Smart Service and PSS engineering design: This research constitutes one of the first attempts to integrate the service perspective into MBSE practices. While the combination of practices could be applied on an industrial example, it still remains to be clarified if the elaborate approach of SE and MBSE is combinable with PSS and Smart Service engineering practices. An alignment in further steps of the SE process, such as requirements definition and detailed design should be conducted in order to investigate the further compatibility of the approaches.

(4) Connectivity to further methods and IT-solutions: MESSIAH CE is neither tool-supported nor embedded in a respective databank system. Interfaces to a number of databases could facilitate the application of the methodology in many ways and unlock synergy potentials. The time for information gathering could be reduced, resulting in sufficient time for solution generation of the engineering design team. The quality of solutions could be enhanced. Currently, the MESSIAH has not been assessed in combination with LCA, S-LCA, LCSA or MFA. Firstly, a concept for combination of these methods with MESSIAH should be developed and tested. An integrated tool with automatized access to respective databases on material and process data with ecological and societal impact could show potential.

(5) Significance in terms of a transition towards CE: Evidence still needs to be gathered on whether PSS and Smart Services contribute to a CE and a sustainable development. Certain potentials exist, but they need to outweigh the significant ecological and societal cost of the introduction of new software and electronics-based systems. This is questionable and will need to be proven through hard evidence. Only, when PSS and Smart Services can actually contribute to CE and a sustainable development, the proposed methodology is useful. Whether the proposed approach can lead to an effective transition towards CE depends on a validation on a substantial number of further industrial examples.

(6) Concluding: The developed methodology proved useful in the scope of the example of SHEILA. The research question can be considered as answered positively. This article presents an approach which successfully integrates the service perspective in MBSE systems modelling and addresses CE goals. A similar study has not been presented so far in the academic literature. A number of operational drawbacks could be identified, which will be addressed in further development of the methodology. The maturity of the methodology will still need to be improved. Through the application on further examples, more evidence on the significance of the results has to be collected.

The authors aim to conduct further research in order to increase the level of validation of MESSIAH. The methodology will have to be applied on further examples, preferably in industry. During the application, the methodology should be improved iteratively. Further blueprints should be modelled, collected and made available in respective Blueprint Libraries. Following the three pillars of MBSE, procedure, language and tool, MESSIAH should be implemented in an IT-tool in order to have proper support. It is envisioned to create a blueprint library in further research and development, which can be accessed in order to find the most suitable blueprints for different purposes and situations. This library could either be hosted internally in the database system of an organization or as online platform, where generic blueprints can be shared across organizations and industries.

**Author Contributions:** The authors contributed the following elements to this article: Conceptualization, F.A.H.; methodology, F.A.H.; validation, F.A.H.; Writing, F.A.H.; supervision, R.S. and K.L.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Steffen, W.; Richardson, K.; Rockström, J.; Cornell, S.E.; Fetzer, I.; Bennett, E.M.; Biggs, R.; Carpenter, S.R.; de Vries, W.; de Wit, C.A. Planetary boundaries: Guiding human development on a changing planet. *Science* **2015**, *347*, 1259855. [CrossRef] [PubMed]
2. Hoornweg, D.; Bhada-Tata, P.; Kennedy, C. Environment: Waste production must peak this century. *Nat. News* **2013**, *502*, 615–617. [CrossRef] [PubMed]
3. Ellen MacArthur Foundation. Towards a circular economy: Economic and business rationale for an accelerated transition. *J. Ind. Ecol.* **2013**, *2*, 23–44.
4. Höök, M.; Tang, X. Depletion of fossil fuels and anthropogenic climate change—A review. *Energy Policy* **2013**, *52*, 797–809. [CrossRef]
5. Bunse, K.; Vodicka, M.; Schönsleben, P.; Brühlhart, M.; Ernst, F.O. Integrating energy efficiency performance in production management—Gap analysis between industrial needs and scientific literature. *J. Clean. Prod.* **2011**, *19*, 667–679. [CrossRef]
6. Tang, T.; Bhamra, T. Changing energy consumption behaviour through sustainable product design. In Proceedings of the 10th International Design Conference, Dubrovnik, Croatia, 19–20 May 2008.
7. Valero Navazo, J.M.; Villalba Méndez, G.; Talens Peiró, L. Material flow analysis and energy requirements of mobile phone material recovery processes. *Int. J. Life Cycle Assess.* **2014**, *19*, 567–579. [CrossRef]
8. Boustani, A.; Sahni, S.; Graves, S.C.; Gutowski, T.G. Appliance remanufacturing and life cycle energy and economic savings. In Proceedings of the 2010 IEEE International Symposium on Sustainable Systems and Technology, Arlington, VA, USA, 19 May 2010; pp. 1–6.
9. European Parliament and Council of the European Union. Directive 2008/98/EC of the European Parliament and of the Council of the European Union: On waste and Repealing Certain Directives. Available online: <https://eur-lex.europa.eu/eli/dir/2008/98/oj> (accessed on 4 June 2019).
10. Pieroni, M.P.; McAloone, T.; Pigosso, D.A.C. Business model innovation for circular economy and sustainability: A review of approaches. *Clean. Prod.* **2019**, *215*, 198–216. [CrossRef]
11. Wijkman, A.; Skånberg, K. The circular economy and benefits for society: Jobs and Climate Clear Winners in an Economy Based on Renewable Energy and Resource Efficiency. A Study Pertaining to Finland, France, The Netherlands, Spain and Sweden. 2015. Available online: <https://www.clubofrome.org/wp-content/uploads/2016/03/The-Circular-Economy-and-Benefits-for-Society.pdf> (accessed on 24 June 2019).
12. Vandermerwe, S.; Rada, J. Servitization of business: Adding value by adding services. *Eur. Manag. J.* **1988**, *6*, 314–324. [CrossRef]
13. Rijdsdijk, S.A.; Hultink, E.J. How Today's Consumers Perceive Tomorrow's Smart Products. *J. Prod. Innov. Manag.* **2009**, *26*, 24–42. [CrossRef]
14. Tukker, A. Product services for a resource-efficient and circular economy—A review. Special Volume: Why have 'Sustainable Product-Service Systems' not been widely implemented? *J. Clean. Prod.* **2015**, *97*, 76–91. [CrossRef]

15. Tukker, A. Eight types of product–service system: Eight ways to sustainability? Experiences from SusProNet. *Bus. Strat. Environ.* **2004**, *13*, 246–260. [CrossRef]
16. Niehoff, S.; Beier, G. Industrie 4.0 and a sustainable development: A short study on the perception and expectations of experts in Germany. *Int. J. Innov. Sustain. Dev.* **2018**, *12*, 360. [CrossRef]
17. Chancerel, P.; Meskers, C.E.; HageläKen, C.; Rotter, V.S.; Hagelüken, C. Assessment of Precious Metal Flows During Preprocessing of Waste Electrical and Electronic Equipment. *J. Ind. Ecol.* **2009**, *13*, 791–810. [CrossRef]
18. Tischner, U.; Verkuijl, M.; Tukker, A. *First Draft Report of PSS Review*; SUSPRONET Report; SUSPRONET: Delft, The Netherlands, 2002.
19. Wong, M.T.N. *Implementation of Innovative Product Service Systems in the Consumer Goods Industry*; University of Cambridge: Cambridge, UK, 2004; ISBN 0000 0001 3571 575X.
20. Haberl, B.; Kopacek, K. IT on demand—towards an environmental conscious service system for Vienna (AT). In Proceedings of the Third International Symposium on Environmentally Conscious Design and Inverse Manufacturing—EcoDesign, Tokyo, Japan, 11 December 2003; pp. 799–802.
21. Baines, T.S.; Lightfoot, H.W.; Evans, S.; Neely, A.; Greenough, R.; Peppard, J.; Roy, R.; Shehab, E.; Braganza, A.; Tiwari, A.; et al. State-of-the-art in product-service systems. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* **2007**, *1543–1552*. [CrossRef]
22. Ceschin, F. *Sustainable Product-Service Systems. Between Strategic Design and Transition Studies*; Springer Science & Business Media: Berlin, Germany, 2013; ISBN 978-3-319-03794-3.
23. Tischner, U.; Ryan, C.; Vezzoli, C. *Design for Sustainability. A Step-by-Step Approach*, 1st ed.; United Nations Environment Programme: Paris, France, 2009; ISBN 92-807-2711-7.
24. Ellen MacArthur Foundation. Circular Economy—Case Studies. Available online: <https://www.ellenmacarthurfoundation.org/case-studies/business/building-blocks/new-business-models> (accessed on 6 June 2019).
25. Sauter, R.; Bode, M.; Kittelberger, D. How Industry 4.0 Is Changing How We Manage Value Creation. White Paper—Horváth & Partners. 2015. Available online: [https://www.horvath-partners.com/fileadmin/horvathpartners.com/assets/05\\_Media\\_Center/PDFs/englisch/Industry\\_4.0\\_EN\\_web-g.pdf](https://www.horvath-partners.com/fileadmin/horvathpartners.com/assets/05_Media_Center/PDFs/englisch/Industry_4.0_EN_web-g.pdf) (accessed on 24 June 2019).
26. Abramovici, M. Smart Products. In *CIRP Encyclopedia of Production Engineering*; Springer: Berlin, Germany, 2015; pp. 1–5.
27. Von Engelhardt, S.; Kudernatsch, W.; Liebchen, T.; Rifai, H.; Seidel, U.; Seifert, I.; Weiler, P.; Wischmann, S. Smart Service Welt—Innovationsbericht: Eine Studie im Rahmen der Begleitforschung zum Technologieprogramm Smart Service Welt 2017. Available online: [https://www.bmwi.de/Redaktion/DE/Publikationen/Digitale-Welt/smart-service-welt-innovationsbericht.pdf?\\_\\_blob=publicationFile&v=8](https://www.bmwi.de/Redaktion/DE/Publikationen/Digitale-Welt/smart-service-welt-innovationsbericht.pdf?__blob=publicationFile&v=8) (accessed on 24 June 2019).
28. Bishop, R.H. *The Mechatronics Handbook, -2 Volume Set*; CRC Press: Boca Raton, FL, USA, 2002; ISBN 9780849300660.
29. Chancerel, P.; Rotter, S. Recycling-oriented characterization of small waste electrical and electronic equipment. *Waste Manag.* **2009**, *29*, 2336–2352. [CrossRef] [PubMed]
30. Sievers, H.; Buijs, B.; Espinoza, L.A.T. Limits to the critical raw materials approach. *Proc. Inst. Civ. Eng. Waste Resour. Manag.* **2012**, *165*, 201–208.
31. Glöser, S.; Espinoza, L.T.; Gandenberger, C.; Faulstich, M. Raw material criticality in the context of classical risk assessment. *Resour. Policy* **2015**, *44*, 35–46. [CrossRef]
32. Buchert, M.; Manhart, A.; Bleher, D.; Pingel, D. *Recycling Critical Raw Materials from Waste Electronic Equipment*; Öko-Institut eV: Freiburg, Germany, 2012.
33. Friedenthal, S.; Griego, R.; Sampson, M. INCOSE Model Based Systems Engineering (MBSE) Initiative. Available online: [https://www.researchgate.net/profile/Mark\\_Sampson3/publication/267687693\\_INCOSE\\_Model\\_Based\\_Systems\\_Engineering\\_MBSE\\_Initiative/links/54ca7c290cf22f98631b167e.pdf](https://www.researchgate.net/profile/Mark_Sampson3/publication/267687693_INCOSE_Model_Based_Systems_Engineering_MBSE_Initiative/links/54ca7c290cf22f98631b167e.pdf) (accessed on 4 June 2019).
34. Stark, R.; Lindow, K.; Reich, D.; Vorsatz, T. Smart Engineering capabilities for new generation product offerings. In Proceedings of the 23rd CIRP Design Conference, Bochum, Germany, 11–13 March 2013.
35. Kagermann, H. Smart Service Welt 2018: Wo Stehen Wir? Wohin Gehen Wir? Available online: [https://www.acatech.de/wp-content/uploads/2018/06/SSW\\_2018.pdf](https://www.acatech.de/wp-content/uploads/2018/06/SSW_2018.pdf) (accessed on 6 June 2019).

36. Ehrlenspiel, K.; Meerkamm, H. *Integrierte Produktentwicklung. Denkabläufe, Methodeneinsatz, Zusammenarbeit*, 6th ed.; Carl Hanser Verlag: Munich, Germany, 2017; ISBN 978-3-446-44089-0.
37. Blumberg, B. *Management von Technologiekoooperationen: Partnersuche und Vertragliche Planung*; Springer: Wiesbaden, Germany, 1998; ISBN 978-3-8244-6778-5.
38. Monczka, R.M.; Handfield, R.B.; Scannell, T.V.; Ragatz, G.L.; Frayer, D.J. *New Product Development. Strategies for Supplier Integration*; ASQ—American Society für Quality: Milwaukee, WI, USA, 2000; ISBN 0-87389-468-5.
39. Pahl, G.; Beitz, W. *Engineering Design. A Systematic Approach*; Springer Science & Business Media: Berlin, Germany, 2013; ISBN 978-3-540-19917-5.
40. Eppinger, S.D.; Ulrich, K.T. *Product Design and Development*, 1th ed.; McGraw Hill: New York, NY, USA, 1995; ISBN 9870070658110.
41. Gausemeier, J.; Czaja, A.; Wiederkehr, O.; Dumitrescu, R.; Tschirner, C.; Steffen, D. *Studie: Systems Engineering in der Industriellen Praxis*; Carl Hanser Verlag: Munich, Germany, 2013; pp. 113–122.
42. Bahill, A.; Gissing, B. Re-evaluating systems engineering concepts using systems thinking. *IEEE Trans. Syst. Man Cybern. Part. C* **1998**, *28*, 516–527. [[CrossRef](#)]
43. Weilkiens, T.; Soley, R.M. *Systems Engineering mit SysML/UML. Anforderungen, Analyse, Architektur*; dpunkt.verl.: Heidelberg, Germany, 2014; ISBN 9783864900914.
44. Walden, D.D.; Roedler, G.J.; Forsberg, K.; Hamelin, R.D.; Shortell, T.M. *Systems Engineering Handbook. A Guide for System Life Cycle Processes and Activities*, 4th ed.; Wiley: Hoboken, NJ, USA, 2015; ISBN 111901512X.
45. Albers, A.; Scherer, H.; Bursac, N.; Rachenkova, G. Model Based Systems Engineering in Construction Kit Development—Two Case Studies. *Procedia CIRP* **2015**, *36*, 129–134. [[CrossRef](#)]
46. Eigner, M.; Gilz, T.; Zafirov, R. Interdisziplinäre Produktentwicklung: Modellbasiertes Systems Engineering. Available online: <https://www.plmportal.org/de/forschung-detail/interdisziplinare-produktentwicklung-modellbasiertes-systems-engineering.html> (accessed on 24 June 2019).
47. Friedenthal, S.; Moore, A.; Steiner, R. *A Practical Guide to SysML. The Systems Modeling Language*, 3rd ed.; Morgan Kaufmann: Waltham, MA, USA, 2015; ISBN 9780128002025.
48. Eigner, M.; Roubanov, D.; Zafirov, R. *Modellbasierte Virtuelle Produktentwicklung*; Springer: Berlin, Germany, 2014; ISBN 366243816X.
49. Feldhusen, J. Konstruktionslehre II: V2 Produktarchitektur/Produktstruktur. Institut für Allgemeine Konstruktionstechnik des Maschinenbaus. Available online: <https://docplayer.org/4030486-Konstruktionslehre-ii-v2-produktarchitektur-produktstruktur-univ-prof-dr-ing-joerg-feldhusen-16-april-2015.html> (accessed on 6 June 2019).
50. Lamm, J.G.; Weilkiens, T. Functional Architecture for Systems Method (FAS): The FAS Method & More. Available online: <http://fas-method.org/content/> (accessed on 8 November 2018).
51. Feldhusen, J.; Grote, K.-H. *Pahl/Beitz Konstruktionslehre: Methoden und Anwendung Erfolgreicher Produktentwicklung*; Springer: Berlin, Germany, 2013; ISBN 364229569X.
52. Tschirner, C.; Ackva, S. Systemspezifikation mit MBSE. Available online: <https://kem.industrie.de/digitalisierung/systems-engineering/systems-engineering-glossar/systemspezifikation-mit-mbse/> (accessed on 6 June 2019).
53. Two Pillars GmbH. Systems Engineering—What Is MBSE? Available online: <https://www.two-pillars.de/what-is-mbse/> (accessed on 6 June 2019).
54. IBM Knowledge Center. The Harmony Process. Available online: [https://www.ibm.com/support/knowledgecenter/SSB2MU\\_8.3.0/com.btc.tcatg.user.doc/topics/atgreqcov\\_SecSysControllerHarmony.html](https://www.ibm.com/support/knowledgecenter/SSB2MU_8.3.0/com.btc.tcatg.user.doc/topics/atgreqcov_SecSysControllerHarmony.html) (accessed on 6 June 2019).
55. Pearce, P.; Hause, M. Iso-15288, oosem and model-based submarine design. Sete Apocose. 2012. Available online: [http://www.omg.sysml.org/Pearce\\_Hause\\_ISO-15288\\_OOSEM\\_and\\_Model-Based\\_Submarine\\_Design\\_SETE\\_APCOSE\\_20121.pdf](http://www.omg.sysml.org/Pearce_Hause_ISO-15288_OOSEM_and_Model-Based_Submarine_Design_SETE_APCOSE_20121.pdf) (accessed on 6 June 2019).
56. Rational. Rational Unified Process for Systems Engineering: RUP SE 1.1. A Rational Software White Paper TP 165A, 5/02. Available online: <https://docplayer.net/14876813-Rational-unified-process-for-systems-engineering-rup-se-1-1-a-rational-software-white-paper-tp-165a-5-02.html> (accessed on 6 June 2019).
57. Vitech Corporation. STRATA: Driving Consistent Project Success: Systems Engineering Solutions. Available online: <http://www.vitechcorp.com/solutions/strata.shtml> (accessed on 6 June 2019).
58. Weilkiens, T. *SYSMOD-The Systems Modeling Toolbox. Pragmatic MBSE with SysML*, 2th ed.; Lulu.com: Hamburg, Germany, 2016; ISBN 978-3-9817875-8-0.



59. Two Pillars GmbH. Consens Methode + iQUAVIS-Software Macht Effizientes Engineering Möglich. Available online: <https://www.two-pillars.de/consens-methode-iquavis/> (accessed on 6 June 2019).
60. PolarSys. Let Yourself Be Guided with Arcadia: A Comprehensive Methodological and Tool-Supported Model-Based Engineering Guidance. Available online: <https://www.polarsys.org/capella/arcadia.html> (accessed on 4 June 2019).
61. Eigner, M.; Gilz, T.; Zafirov, R. Proposal for functional product description as part of a PLM solution in interdisciplinary product development. In Proceedings of the DESIGN 2012 12th International Design Conference, Dubrovnik, Croatia, 24 May 2012.
62. Alt, O. *Modellbasierte Systementwicklung mit SysML*; Carl Hanser Verlag: Munich, Germany, 2012; ISBN 978-3-446-43066-2.
63. Becker, J.; Beverungen, D.; Knackstedt, R. Reference Models and Modeling Languages for Product-Service Systems Status-Quo and Perspectives for Further Research. In Proceedings of the 41st Annual Hawaii International Conference on System Sciences, Waikoloa, HI, USA, 10 January 2008.
64. BPMN. *Version 2.0: OMG Specification*; BPMN: Needham, MA, USA, 2011; pp. 22–31.
65. Shostack, G.L. How to Design a Service. *Eur. J. Mark.* **1982**, *16*, 49–63. [CrossRef]
66. IEEE. IEEE Standard for Functional Modeling Language—Syntax and Semantics for IDEF0. Available online: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=749110> (accessed on 4 June 2019).
67. Object Management Group. Unified Modeling Language®UML®: Version 2.5.1. Available online: <https://www.omg.org/spec/UML/2.5.1/PDF> (accessed on 6 June 2019).
68. OMG. OMG SysML: What Is SysML? Available online: <http://www.omgsysml.org/what-is-sysml.htm> (accessed on 6 June 2019).
69. Lindow, K.; Riedelsheimer, T.; Lünemann, P.; Stark, R. Betrachtung des Entwicklungsumfeldes durch die methodische Datenflussanalyse. *ProStep iVip* **2017**, *2*, 52–56.
70. Kazemzadeh, Y.; Milton, S.K.; Johnson, L.W. A conceptual comparison of service blueprinting and business process modeling notation (BPMN). *IOSR J. Bus. Manag.* **2014**, *16*, 1–10. [CrossRef]
71. Rose, C.M.; Beiter, K.A.; Ishii, K. Determining end-of-life strategies as a part of product definition. In Proceedings of the 1999 IEEE International Symposium on Electronics and the Environment, Boston, MA, USA, 13 May 1999; pp. 219–224.
72. Stahel, W.R. *The Functional Economy: Cultural and Organizational Change*; National Academies Press: Washington, DC, USA, 1997; ISBN 0-309-52042-8.
73. Braungart, M.; McDonough, W. *Cradle to Cradle. Einfach Intelligent Produzieren*, 1st ed.; Piper Verlag: Munich, Germany, 2014; ISBN 978-3-492-96479-1.
74. Benyus, J.M. *Biomimicry: Innovation Inspired by Nature*; Morrow: New York, NY, USA, 1997; ISBN 9780060533229.
75. Lifset, R.; Graedel, T.E. Industrial Ecology: Goals and Definitions. Available online: [http://planet.uwc.ac.za/NISL/ESS/Documents/Industrial\\_Ecology\\_Overview.pdf](http://planet.uwc.ac.za/NISL/ESS/Documents/Industrial_Ecology_Overview.pdf) (accessed on 4 June 2019).
76. Hawken, P.; Lovins, A.B.; Lovins, L.H. *Natural Capitalism. The Next Industrial Revolution*; Routledge: Abingdon, UK, 2013; ISBN 9781134033065.
77. Pauli, G.A. *The Blue Economy. 10 Years, 100 Innovations, 100 Million Jobs*; Paradigm Publications: New Mexico, NM, USA, 2010; ISBN 9780912111902.
78. Bakker, C.A. Product Lifetimes and the Environment: A Product Design Framework for a Circular Economy. In Proceedings of the PLATE Conference Nottingham Trent University [Online], Nottingham, UK, 19 January 2015; pp. 365–372. Available online: [https://www.researchgate.net/profile/Giuseppe\\_Salvia/publication/303476076\\_Product\\_Lifetimes\\_And\\_The\\_Environment\\_Conference\\_Proceedings/links/57447ba808aea45ee85306ca.pdf#page=373](https://www.researchgate.net/profile/Giuseppe_Salvia/publication/303476076_Product_Lifetimes_And_The_Environment_Conference_Proceedings/links/57447ba808aea45ee85306ca.pdf#page=373) (accessed on 4 June 2019).
79. Hansen, E.; Schmitt, J. Circular Economy: Potenziale für Produkt-und Geschäftsmodellinnovation heben. UC Journal (Umwelttechnik Cluster). *Magazin für Umwelttechnik und Kooperation, Ausgabe* **2016**, *2*, 8–10. [CrossRef]
80. Gray, C.; Charter, M. *Remanufacturing and product design: Designing for the 7th generation: Designing for the 7th Generation*; South East England Development Agency: England, UK, 2007.
81. Rose, C.M. Design for environment: A method for formulating product end-of-life strategies for Electronics Industry. *J. Stanf. Univ. Stanf.* **2000**, *11*. [CrossRef]

82. Gehin, A.; Zwoliński, P.; Brissaud, D. A tool to implement sustainable end-of-life strategies in the product development phase. *J. Clean. Prod.* **2008**, *16*, 566–576. [[CrossRef](#)]
83. Ghisellini, P.; Cialani, C.; Ulgiati, S. A review on circular economy: The expected transition to a balanced interplay of environmental and economic systems. *J. Clean. Prod.* **2016**, *114*, 11–32. [[CrossRef](#)]
84. Lieder, M.; Rashid, A. Towards circular economy implementation: A comprehensive review in context of manufacturing industry. *J. Clean. Prod.* **2016**, *115*, 36–51. [[CrossRef](#)]
85. Allwood, J.M.; Ashby, M.F.; Gutowski, T.G.; Worrell, E. Material efficiency: A white paper. *Resour. Conserv. Recycl.* **2011**, *55*, 362–381. [[CrossRef](#)]
86. Guinée, J.B. Handbook on life cycle assessment operational guide to the ISO standards. *Int. J. Life Cycle Assess.* **2002**, *7*, 311–313. [[CrossRef](#)]
87. Jørgensen, A.; Le Bocq, A.; Nazarkina, L.; Hauschild, M. Methodologies for social life cycle assessment. *Int. J. Life Cycle Assess.* **2008**, *13*, 96. [[CrossRef](#)]
88. Finkbeiner, M.; Schau, E.M.; Lehmann, A.; Traverso, M. Towards Life Cycle Sustainability Assessment. *Sustainability* **2010**, *2*, 3309–3322. [[CrossRef](#)]
89. Brunner, P.H.; Rechberger, H. Practical handbook of material flow analysis. *Int. J. Life Cycle Assess.* **2004**, *9*, 337–338. [[CrossRef](#)]
90. Rose, C.M.; Ishii, K.; Stevels, A. Influencing Design to Improve Product End-of-Life Stage. *Res. Eng. Des.* **2002**, *13*, 83–93. [[CrossRef](#)]
91. Ramani, K.; Ramanujan, D.; Bernstein, W.Z.; Zhao, F.; Sutherland, J.; Handwerker, C.; Choi, J.-K.; Kim, H.; Thurston, D. Integrated Sustainable Life Cycle Design: A Review. *J. Mech. Des.* **2010**, *132*. [[CrossRef](#)]
92. Romme, G.; Endenburg, G. Construction Principles and Design Rules in the Case of Circular Design. *Organ. Sci.* **2006**, *17*, 287–297. [[CrossRef](#)]
93. Vanegas, P.; Peeters, J.R.; Cattrysse, D.; Tecchio, P.; Ardente, F.; Mathieux, F.; Dewulf, W.; Duflou, J.R. Ease of disassembly of products to support circular economy strategies. *Resour. Conserv. Recycl.* **2018**, *135*, 323–334. [[CrossRef](#)] [[PubMed](#)]
94. Knapp, J.; Allesch, A.; Müller, W.; Bockreis, A. Methods to estimate the transfer of contaminants into recycling products—A case study from Austria. *Waste Manag.* **2017**, *69*, 88–100. [[CrossRef](#)]
95. Mohamed, S.A.A.; Lou, E.; Mativenga, P.T. What should be recycled: An integrated model for product recycling desirability. *J. Clean. Prod.* **2017**, *154*, 51–60. [[CrossRef](#)]
96. De Aguiar, J.; De Oliveira, L.; Da Silva, J.O.; Bond, D.; Scalice, R.K.; Becker, D. A design tool to diagnose product recyclability during product design phase. *J. Clean. Prod.* **2017**, *141*, 219–229. [[CrossRef](#)]
97. Mangun, D.; Thurston, D. Incorporating component reuse, remanufacture, and recycle into product portfolio design. *IEEE Trans. Eng. Manag.* **2002**, *49*, 479–490. [[CrossRef](#)]
98. Ceha, A. Tackling the Circular Economy: Aiding Firms in the Design and Implementation of Circular Business Models. Unpublished Thesis, University of Twente, Enschede, The Netherlands, 2017.
99. Asif, F.M.; Lieder, M.; Rashid, A. Multi-method simulation-based tool to evaluate economic and environmental performance of circular product systems. *J. Clean. Prod.* **2016**, *139*, 1261–1281. [[CrossRef](#)]
100. Kjaer, L.L.; Pigosso, D.C.A.; Niero, M.; Bech, N.M.; McAlloone, T.C. Product/Service-Systems for a Circular Economy: The Route to Decoupling Economic Growth from Resource Consumption? *J. Ind. Ecol.* **2019**, *23*, 22–35. [[CrossRef](#)]
101. Moreno, M.; Rios, C.D.L.; Rowe, Z.; Charnley, F. A Conceptual Framework for Circular Design. *Sustainability* **2016**, *8*, 937. [[CrossRef](#)]
102. Wastling, T.; Charnley, F.; Moreno, M. Design for Circular Behaviour: Considering Users in a Circular Economy. *Sustainability* **2018**, *10*, 1743. [[CrossRef](#)]
103. Okorie, O.; Salonitis, K.; Charnley, F.; Moreno, M.; Turner, C.; Tiwari, A. Data-Driven Approaches for Circular Economy in Manufacturing for Digital Technologies: A Review of Current Research and Proposed Framework. *Preprints* **2018**. [[CrossRef](#)]
104. Bocken, N.M.P.; De Pauw, I.; Bakker, C.; Van Der Grinten, B. Product design and business model strategies for a circular economy. *J. Ind. Prod. Eng.* **2016**, *33*, 308–320. [[CrossRef](#)]
105. Bakker, C.A.; Hollander, M.C.; Hultink, E.J. Product Design in a Circular Economy: Development of a Typology of Key Concepts and Terms. *J. Ind. Ecol.* **2017**, *21*, 517–525.
106. Lewandowski, M. Designing the Business Models for Circular Economy—Towards the Conceptual Framework. *Sustainability* **2016**, *8*, 43. [[CrossRef](#)]

107. Linder, M.; Williander, M. Circular business model innovation: Inherent uncertainties. *Bus. Strat. Environ.* **2017**, *26*, 182–196. [[CrossRef](#)]
108. Meier, H.; Uhlmann, E.; Kortmann, D. Hybride Leistungsbündel–Nutzenorientiertes Produktverständnis durch interferierende Sach- und Dienstleistungen. *wt Werkstattstechnik Online* **2005**, *95*, 8.
109. Schendel, C.; Spahl, T.; Birkhofer, H. Considerations towards Systematic Engineering of Product Service Systems. In Proceedings of the NordDesign 2008 Conference, Tallinn, Estonia, 23 August 2008.
110. Muto, K.; Kimita, K.; Shimomura, Y. A Guideline for Product-Service-Systems Design Process. *Procedia CIRP* **2015**, *30*, 60–65. [[CrossRef](#)]
111. Müller, P. *Integrated Engineering of Products and Services. Layer-Based Development Methodology for Product-Service Systems*; Fraunhofer Verlag: Stuttgart, Germany, 2014; ISBN 3839605490.
112. Van Halen, C.; Vezzoli, C.; Wimmer, R. *Methodology for Product Service System Innovation. How to Develop Clean, Clever and Competitive Strategies in Companies*; Van Gorcum: Assen, The Netherlands, 2005; ISBN 90-232-4143-6.
113. Welp, E.G.; Sadek, T. Conceptual design of Industrial Product Service Systems (IPSS) based on the extended heterogeneous modelling approach. In Proceedings of the DESIGN 2008 10th International Design Conference, Dubrovnik, Croatia, 19–20 May 2008.
114. Maussang, N.; Zwolinski, P.; Brissaud, D. Product-service system design methodology: From the PSS architecture design to the products specifications. *J. Eng. Des.* **2009**, *20*, 349–366. [[CrossRef](#)]
115. Uhlmann, E.; Bochnig, H. The Philosopher’s Stone for Sustainability: PSS-CAD: An Approach to the Integrated Product and Service Development. In Proceedings of the 4th CIRP International Conference on Industrial Product-Service Systems, Tokyo, Japan, 9 November 2012; pp. 61–66.
116. Alexopoulos, K. Resource Planning for the Installation of Industrial Product Service Systems. In Proceedings of the IFIP International Conference on Advances in Production Management Systems, Hamburg, Germany, 7 July 2017; pp. 205–213.
117. Pezzotta, G.; Pirola, F.; Rondini, A.; Pinto, R.; Ouertani, M.-Z. Towards a methodology to engineer industrial product-service system—Evidence from power and automation industry. *CIRP J. Manuf. Sci. Technol.* **2016**, *15*, 19–32. [[CrossRef](#)]
118. Haber, N.; Fargnoli, M. Design for product-service systems: A procedure to enhance functional integration of product-service offerings. *Int. J. Prod. Dev.* **2017**, *22*, 135. [[CrossRef](#)]
119. Sun, J.; Chai, N.; Pi, G.; Zhang, Z.; Fan, B. Modularization of Product Service System Based on Functional Requirement. *Procedia CIRP* **2017**, *64*, 301–305. [[CrossRef](#)]
120. Joore, P.; Brezet, H. A Multilevel Design Model: The mutual relationship between product-service system development and societal change processes. *J. Clean. Prod.* **2015**, *97*, 92–105. [[CrossRef](#)]
121. Fadeyi, J.A.; Monplaisir, L.; Aguwa, C. The integration of core cleaning and product serviceability into product modularization for the creation of an improved remanufacturing-product service system. *J. Clean. Prod.* **2017**, *159*, 446–455. [[CrossRef](#)]
122. Boughnim, N.; Yannou, B. Using blueprinting method for developing product-service systems. In Proceedings of the International Conference of Engineering Design (ICED), Melbourne, Australia, 15–18 August 2005.
123. Song, W.; Sakao, T. A customization-oriented framework for design of sustainable product/service system. *J. Clean. Prod.* **2017**, *140*, 1672–1685. [[CrossRef](#)]
124. Kuijken, B.; Gemser, G.; Wijnberg, N.M. Effective product-service systems: A value-based framework. *Ind. Mark. Manag.* **2017**, *60*, 33–41. [[CrossRef](#)]
125. Sakao, T.; Shimomura, Y. Service Engineering: A novel engineering discipline for producers to increase value combining service and product. *J. Clean. Prod.* **2007**, *15*, 590–604. [[CrossRef](#)]
126. Sakao, T.; Lindahl, M. *Introduction to Product/Service-System Design*; Springer: Berlin, Germany, 2009; ISBN 978-1-84882-908-4.
127. Tomiyama, T. Service engineering to intensify service contents in product life cycles. In Proceedings of the Second International Symposium on Environmentally Conscious Design and Inverse Manufacturing, Tokyo, Japan, 15 December 2001.
128. Exner, K.; Smolka, E.; Blüher, T.; Stark, R. A method to design Smart Services based on information categorization of industrial use cases. In Proceedings of the 11th CIRP Conference on Industrial Product-Service Systems, Zhuhai and Hong Kong, China, 29–31 May 2019.

129. Halstenberg, F.A.; Stark, R. Introducing Product Service System Architectures for realizing Circular Economy. In Proceedings of the 16th Global Conference on Sustainable Manufacturing, Lexington, KY, USA, 9–11 October 2019.
130. Burghardt, M. *Projektmanagement: Leitfaden für die Planung, Überwachung und Steuerung von Projekten*; John Wiley & Sons: Hoboken, NJ, USA, 2018; ISBN 3895789593.
131. Friedrichs, W.; Ostermann, J.; Lutz, M.; Buschhorn, B.; Joepen, M. *Das Fitnessprogramm für KMU: Methoden für mehr Effizienz im Automobil-, Anlagen-und Sondermaschinenbau*; Carl Hanser Verlag: Munich, Germany, 2017; ISBN 3446453733.
132. Meyer, H. *Projektmanagement Fachmann*, 10th ed.; Wissenschaft & Praxis: Sternenfels, Germany, 2011; ISBN 978-3-89673-575-1.
133. Senatsverwaltung für Umwelt, Verkehr und Klimaschutz—Stadt Berlin. Radverkehrsstrategie. Available online: [https://www.berlin.de/senuvk/verkehr/politik\\_planung/rad/strategie/index.shtml](https://www.berlin.de/senuvk/verkehr/politik_planung/rad/strategie/index.shtml) (accessed on 6 December 2018).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).