



# Design-integrated financial assessment of smart services

Jürgen Anke<sup>1</sup>

Received: 31 March 2017 / Accepted: 14 May 2018 / Published online: 1 June 2018  
© Institute of Applied Informatics at University of Leipzig 2018

## Abstract

The emergence of data-driven services in the Internet of Things provides manufacturers of technical products the opportunity to become providers of services, which use data as primary input. The development of such smart services, however, is characterized by high complexity and uncertainty. To identify service ideas which are worth to pursue further, the potential financial impact is an important criterion. While the importance of a business case is acknowledged in service engineering, there is currently no concrete method that is integrated with the early design phases of smart services. Therefore, we propose a tool-based method for the design-integrated financial assessment of smart services. We use a design science research approach to develop a meta-model, which is implemented in a web-based tool. The experimental evaluation shows that the proposed tool provides benefits, especially in structuring the task for project teams. Therefore, it appears to be beneficial to provide interdisciplinary teams a tool-based support for the design and evaluation of smart services.

**Keywords** Smart services · Intelligent products · Service engineering · Product-service systems · Cyber-physical systems · Profitability analysis

**JEL classification** O320: Management of Technological Innovation and R&D · L86: Information and Internet Services · Computer Software. · O14: Industrialization · Manufacturing and Service Industries · Choice of Technology.

## Introduction

### Motivation

The ongoing proliferation of connected devices and assets drives the emergence of the Internet of Things (IoT), which is the enabler for a variety of innovative applications. Recent forecasts estimate the market size of IoT solutions to grow to 267 billion USD in 2020, of which 50% is to be created for applications in industries like discrete manufacturing, transportation and logistics, and utilities (Columbus 2017). Autonomous data acquisition through sensors as well as the remote control of devices through actuators using the Internet is the basis of so-called smart services (Allmendinger and Lombreglia 2005; Georgakopoulos and

Jayaraman 2016). The concept refers to physical products, which are augmented with globally usable digital functions, in addition to their local physical functions (Fleisch et al. 2015). The provision of such services is based on the recording of sensors and operational data, its transmission via digital networks, as well as its evaluation and the delivery of the analysis results, e.g. via smartphone apps. For example, networked bicycles warn of chain wear and call assistance in case of accidents (Shaw 2014). Industrial products such as compressors, ventilation systems, and elevators are also being upgraded with digital services for remote control, monitoring, usage-based billing and other services (Herterich et al. 2015). As the type of value exchange is shifted from selling products to providing services, smart services allow for a completely new relationship between manufacturers, operators, and users of physical goods and thus enable new business models (Velamuri et al. 2013; Zolnowski and Böhmann 2013).

How to turn these opportunities into useful applications and economic benefit has become an important topic in various research streams in the recent years (Wuenderlich et al. 2015). One research stream is business models in the Internet of Things (Velamuri et al. 2013; Fleisch et al. 2015), another is

---

Responsible Editor: Haluk Demirkan

✉ Jürgen Anke  
anke@hft-leipzig.de

<sup>1</sup> Hochschule für Telekommunikation Leipzig, Gustav-Freytag-Straße 43-45, 04277 Leipzig, Germany

the creation of IoT-based product-service systems. As Herterich et al. (2016) show, digitized products can enable service innovation for industrial products. The process of “creating value by adding services to products” (Baines et al. 2009) is called “servitization”. It provides opportunities for manufacturers to establish new customer relationships, increase loyalty, differentiate from competitors (Neely 2008; Fischer et al. 2012) and create new data-driven business models (Wiesner et al. 2013; Zolnowski et al. 2016). From a marketing perspective, a smart service contains both service elements and physical products. They can be therefore considered as product-service systems (PSS). PSS are well suited as a unit of analysis as they are “special case of servitization” (Baines et al. 2007) and even enable result-oriented services as a business model (Tukker 2004; Adrodegari et al. 2015). In its most advanced form, products are offered as outcome-based contracts, i.e. the customer pays for the actual performance of a product rather than the product itself (Wuenderlich et al. 2015; Visnjic et al. 2016). All these efforts contribute to the overall goal of higher customer-orientation, which improves the competitiveness of service providers (Brady and Cronin 2001).

While the potential of digital services in the IoT appears to be obvious, there is only fragmented knowledge on how to systematically develop them (Böhmman et al. 2014; Wuenderlich et al. 2015). Designing service systems for connected products is challenging as it requires the right configuration of people, technology, organization and information to create value for both the provider as well as the consumer of a service (Maglio and Spohrer 2013). Due to the complexity of PSS, it is reasonable to support this process with IT-tools for collaboration and modeling, which is however still in its nascent stage (Pezzotta et al. 2015). One approach, called Computer Aided Service Design (CASD) is proposed by Laurischkat (2013). The CASD method supports the design of both manual as well as automated service elements in PSS. It focuses particularly on the knowledge management and service information reuse in early conception phases. A comprehensive approach for the engineering of “informatics-based services” is proposed by Lim et al. (2015). Essentially, they support the design of services for vehicles, heavy equipment, and machinery by identifying needs through analysis of data, which comes from the respective products as well as other data sources. Additionally, they present a conceptual model of the informatics-based service value creation process (Lim et al. 2015). Marilungo et al. (2016) propose an integrated toolset for the ideation and design phase of PSS to support open-innovation processes. These allow integrating the customer, who is a key stakeholder in the design process (Isaksson et al. 2011).

This paper aims to contribute to the body of knowledge in service system engineering by focusing on the tool-based financial evaluation of smart services in the early stages of

conception. The development of these services is an interdisciplinary task, which sets the context of an application for the envisaged tool.

## Research goal

The development of smart services, like every service development, is a creative process that is often carried out in interdisciplinary teams and is characterized by high complexity (Barrett et al. 2015). Particularly in the early stages of such processes, often many ideas are created. However, only a limited number of them can be pursued further to allocate resources effectively. This is also highlighted within the “IoT Business Model Builder”, a toolset jointly developed by University St. Gallen and Bosch: “Besides developing a qualitative business model, it is essential to predict quantitative forecasts (i.e., a business case) to back investment decisions” (Bilgeri et al. 2015). Thus the financial case has to be a possible break-off criterion of the development process (Alam and Perry 2002). The evaluation of product-service systems (PSS) is called for in many PSS engineering methods but lacks concrete methods that are integrated with the design of the services (Lin and Hsieh 2011).

In today’s businesses, a large number of business decisions including financial evaluations are performed using spreadsheets like Microsoft Excel (Grossman et al. 2007). For example, a study shows that in agile projects, spreadsheets were the second-most used tool behind wall and paper (Azizyan et al. 2011). Spreadsheets are also used in the design of sustainable PSS (Omann 2003), for assessing IoT business models (Bilgeri et al. 2015), and the early stage evaluation of medical innovations (Craven et al. 2013). However, spreadsheets have been found to contain various types of errors, which impede their usage for business decisions (Caulkins et al. 2007; Panko and Aurigemma 2010; Reschenhofer and Matthes 2015).

Spreadsheets work well if the drivers of revenue and cost are known and well-structured for the investment object at hand. However, in early design phases of smart services, the structure of a service system is elaborated and restructured frequently, e.g. in interdisciplinary workshops (Dewit et al. 2014). To provide the financial impact of design decisions in the early stages of service conception, the structure of a spreadsheet would have to be adapted whenever elements are added, modified and removed or new information on prices, offers, cost, and quantities become available. This type of evolution of spreadsheets is an erroneous task, as the manageability of spreadsheets has found to be limited (Reschenhofer and Matthes 2015). Consequently, if spreadsheets are used, the financial evaluation will not take place before the service concept is relatively stable. Furthermore, the users must develop a spreadsheet model that contains all financially relevant information and formulae on their own. Afterwards, all information must be manually transferred from

the service concept into a spreadsheet. To mitigate these problems, our research goal is to explore the potential of tool-based support for the design-integrated and continuous financial evaluation in the early phase of smart services for connected products. More specifically, we aim to:

- Develop a flexible data structure that captures the main drivers for profitability in a smart service
- Provide a tool, which enables fast and iterative modeling of smart services elements and their attributes
- Enable immediate update of profitability on every model change to illustrate financial impact and support decision making in the conception phase

From these goals, we have derived two research questions to design and evaluate such a tool:

- RQ1: How can the profitability of a smart service be assessed at an early stage of service design?
- RQ2: What is the benefit of employing a tool for profitability assessment in the engineering process?

To provide such a tool, the structure of service, which emerges in the conception phase, must be linked with a financial case structure. In this paper, we argue that meta-models for the service and the financial case can be developed and instantiated in a web-based tool prototype, to fulfill the research goals stated above.

## Research methodology

We use the Design Science Research Process as proposed by Peffers et al. (2007) based on the design science approach by Hevner et al. (2004). In the first step, we analyze the problem domain based on a literature review, which includes identification of key concepts and terms. The structure of smart services, as well as the design process, are extracted from literature and analyzed to identify the causes of the difficulties in the assessment of smart services. These are used to derive requirements for a tool-based approach that addresses these challenges. The second step comprises the artifact design, which focuses on the development of meta-models for both the service and the financial case. In a first evaluation step, the meta-models are instantiated using real-world cases. Furthermore, a web-based tool prototype is developed to make them accessible for interactive manipulation. The third step is a lab experiment, which aims to reconstruct a service conception and evaluation context. It is evaluated through a survey.

The remainder of this paper is structured according to the outlined method, which is depicted in Fig. 1, followed by a discussion and conclusion.

## Problem analysis

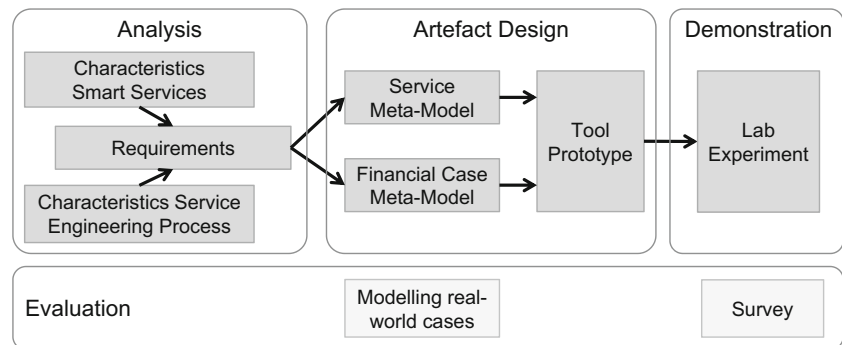
### Smart services

To understand smart services as the artifact to be designed, we reviewed the literature to identify main characteristics of smart services as an input for the requirements analysis. Allmendinger and Lombreglia (2005) characterize *smart services* as being pre-emptive in their behavior, creating the “value of removing unpleasant surprises”, and relying on machine intelligence provided by information technology. As outlined by Porter and Heppelmann (2014), such services require computing, sensors, and communication capabilities embedded into products. With these in place, data can be exchanged with the manufacturer to integrate the product as an external factor and create services for its customer (Kees et al. 2015). Smart services are therefore service systems, which enable value co-creation between service provider and beneficiary through the joint performance of service activities (Edvardsson et al. 2011).

Products with these capabilities are termed *smart things* (Püschel et al. 2016), *smart objects* (Kortuem et al. 2010) or *intelligent products* (Leitão et al. 2015). As intelligent products require communication to a central server, as well as various processes and potentially further external internet services, they can become part of a cyber-physical system (CPS) (Barbosa et al. 2016). From a technological point of view, smart services qualify therefore as CPS, which “involve a multitude of parallel and interlinked sensors, computers, and machines, which collect and interpret data to decide on this basis and control real-world physical processes” (Marilungo et al. 2017). With the ongoing proliferation of IT components in physical products, the potential of cyber-physical systems and smart things as an enabler of PSS has been recognized and conceptually substantiated (Mikusz 2014; Herterich et al. 2015; Medina-Borja 2015; Marilungo et al. 2017). The emergence of digital service systems in the IoT is accompanied by the analysis of their impact on business models (Fleisch et al. 2015; Laudien and Daxböck 2016) and innovation (Herterich et al. 2016). Finally, the concept of smart service systems emerged within service science, which highlights the adaptability of systems, e.g. with the help of big data (Maglio and Lim 2016).

Other concepts similar to smart services for connected products are Product Extension Services and Cyber-Physical Features (Scholze et al. 2016) or Extended Products (Thoben et al. 2003). As these terms are less established, we use “smart services” for our research and define them as data-driven services for technical products, which are provided as product-services systems based on cyber-physical systems. More specifically, a smart service is typically provided using the following approach: A networked device provides information about its state, which is detected, for example, by means of

**Fig. 1** Design science-oriented research approach



sensors. On some devices, also actuators (control operations) must be considered (Porter and Heppelmann 2014). Communication between the device and the central server or cloud service is done over the Internet using machine-to-machine communication (Wortmann and Flüchter 2015). Users interact with smart services through various clients such as mobile apps or web applications.

From these aspects found in literature, the following five characteristics (SC1..SC5) of smart services were identified:

- *Data transmission [SC1]*: To transfer data from products to a central server, e.g. in the cloud, connectivity is needed (Weinberger et al. 2016). Depending on the transmission technology, usage-based cost is incurred for data transmission between product and manufacturer, e.g. for mobile network usage (Luong et al. 2016).
- *External Services [SC2]*: Data is used for operational and analytical functions in the cloud, such as “determine current position” or “list of most frequent operating conditions”. Functions may also require external Internet services, such as weather information, which are provided with different pricing models (Laatikainen et al. 2013).
- *Services as a combination of functions [SC3]*: Smart services are a combination of functions provided as an offer for a specific target group (Kim et al. 2015). The functionality of the services and their prices must be aligned with the needs of the target group (Peruzzini et al. 2015).
- *Value Co-Creation [SC4]*: Customers have to integrate their resources in the service system, to have it perform its function for the creation of the desired outcome (Edvardsson et al. 2011).
- *Various types of costs and pricing models [SC5]*: To operate a smart service system, costs with different payment intervals must be considered, which are settled in different ways. For cloud services, for example, there are one-time and running costs, which can additionally be dependent on the number of users or devices (Laatikainen et al. 2013).

The complexity of smart services drives the challenges associated with their conception in the design phase, as discussed in the next section.

## Engineering of smart services and digital business models

Engineering methods for services are an important topic within service science. Currently, the engineering of service systems still lacks suitable models, methods and design knowledge to exploit the opportunities provided by such systems (Böhm et al. 2014). Within the realm of PSS, a number of design methodologies have been developed (see Cavalieri and Pezzotta 2012; Vasantha et al. 2012 for an overview). Scherer et al. (2016) propose a stage-gate-based process for complex systems and a “PSS Canvas” for simpler cases. Other variants of PSS engineering methods include the focus on PSS for consumer products with integrated intelligent data units and other IoT-technology (Yang et al. 2009; Carpanen et al. 2016), applying the design science methodology for PSS development (Niemöller et al. 2014), or improve the development of PSS by adopting ideas from the design of functional products, which put higher emphasis on IT-components (Sas and Lindström 2014).

As PSS are understood as integrated offers, their design is also intended to be integrated (cf. Marques et al. 2013). However, in this research, we focus on existing products and their servitization through additional services rather than the integrated development of product and services. To support this transformation, Pieroni et al. (2016) propose a methodology called “PSS Transition Framework”, which is depicted in Fig. 2. As the business dimensions in the lower part of the figure show, financial criteria (cost and revenue) are to be defined in the first part of the overall process, called Front End of Innovation (FEI). Hence, the economic viability is to be decided at the end of the FEI stage and could be used to determine, whether the project should proceed into the Development phase or not.

In summary, the following four characteristics (PC1..PC4) of the design process for smart services were identified from literature:

- *Interdisciplinarity and information asymmetries [PC1]*. To design successful smart services, the customer requirements, technical possibilities, and financial requirements have to be

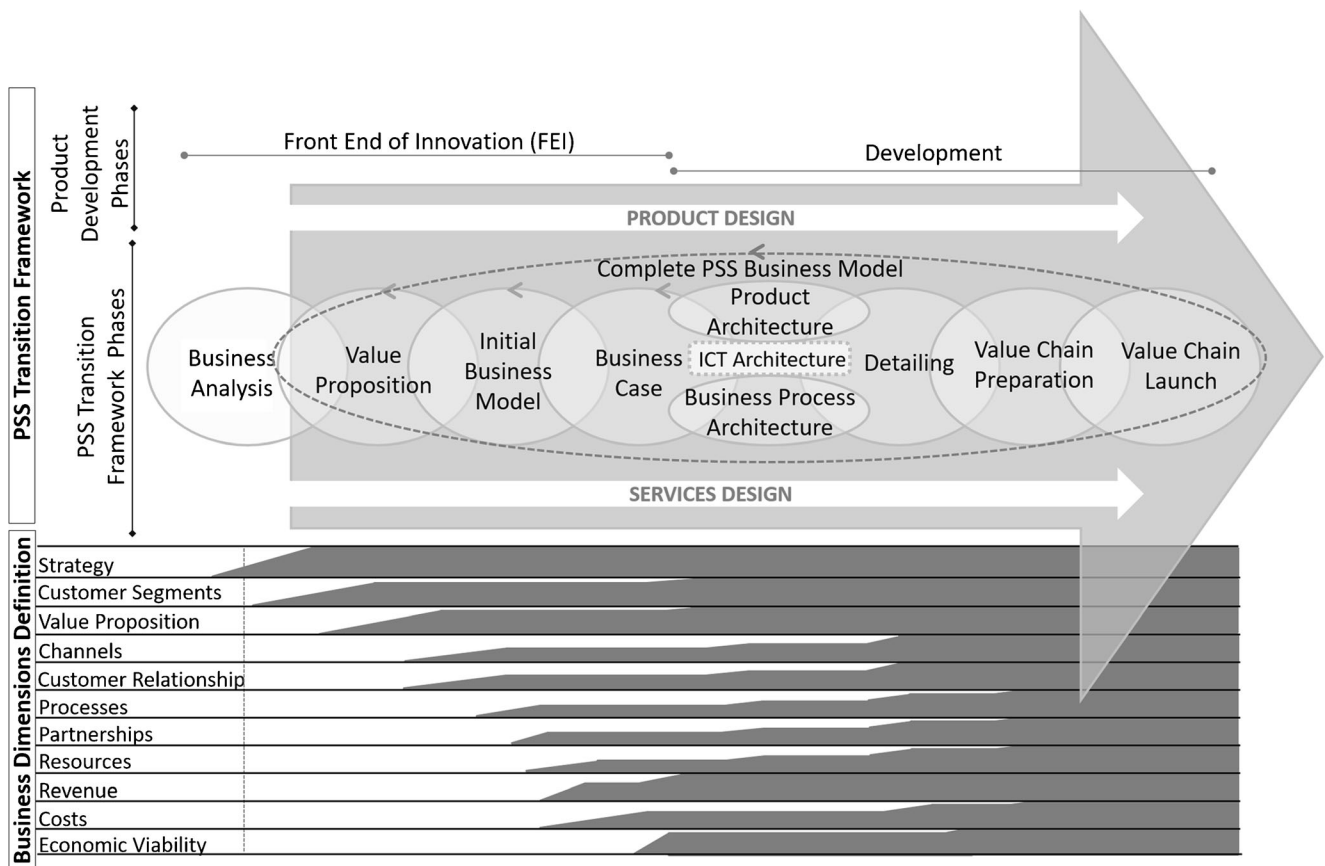


Fig. 2 PSS Transition Framework (Pieroni et al. 2016)

aligned (Maglio and Spohrer 2013). Manufacturers face a variety of decisions, which are related to technology and influence the value of the service for both the customer and the provider at the same time. Contributions and requirements from various stakeholders such as marketing, development, IT, sales, purchasing, controlling and the customer are relevant for this (Wallin 2013). As Kim and Bae (2012) point out, there are different stakeholders with conflicting goals involved in the process of PSS design. Therefore, an asymmetry of information between the parties exists, which leads to high coordination costs.

- *Creativity and interaction [PC2]*. In the early stages of the concept, the service ideas are developed iteratively with different participants. There is a high degree of creativity and interaction, whereby findings do not arise in a predictable order (Dewit et al. 2014). Ideas and interim results should be recorded promptly, and their impact be assessed.
- *Complexity and interdependencies [PC3]*. Smart services are complex systems, as they consist of many elements, which influence each other (Wiesner and Thoben 2017). For example, a more detailed data analysis may require more frequent data queries from the devices. This, in turn, leads to higher data volumes, higher transmission costs and higher energy consumption in the communication module. If the necessary data rate is higher than the planned capacity, it may even be

necessary to switch to a more powerful transmission technology, which results in more expensive communication modules with different physical dimensions. These interdependencies are not always obvious to all parties involved, which might lead to misjudgments and increased planning efforts.

- *Uncertainty in the estimation of key parameters [PC4]*. Design decisions for services are often based on uncertain information (Klein et al. 2004), e.g. customer requirements, market development and willingness to pay. For the business model, quantitative parameters such as prices, price models, and customer numbers must also be defined. While there is preparatory work on the selection of pricing models for internet services (Stiller et al. 2003), the concrete price must be determined considering own costs, customer needs, competitive environment and profitability targets. Especially regarding the pricing for new services, however, providers often have little experience (Baines et al. 2007). In addition, there is uncertainty about the customer growth rate, their usage behavior, and the behavior of the networked devices in use.

### Early stage evaluation of services

The financial evaluation is typically performed at a stage, where existing service ideas are elaborated into service

concepts. For example, in the “Feasibility Analysis” phase of the technical service design process proposed by Aurich et al. (2006) the assessment of cost and benefits is performed. In the strategy-based service engineering approach proposed by Ehrenhofer and Kreuzer (2012), a “financial reflection” activity is part of the “Service Design I” phase. While many PSS engineering methods call for a “business analysis” to determine the financial impact, there is a lack of concrete methods how this can be achieved (Lin and Hsieh 2011). As a comparison by Marques et al. (2016) show, there are only a few concrete activities related to financial analysis in PSS engineering methods.

However, some evaluation approaches were developed for specific situations and purposes. Established methods like cost-benefit analysis, total cost of ownership (TCO) or lifecycle costing (LCC) have only recently been linked to PSS engineering (Kambanou and Lindahl 2016). Becker et al. (2009) proposed a method for the calculation of economic effects for customer-specific configurations (value bundles) of PSS. Other evaluation approaches of PSS focus on the customer value, rather than the financial impact (Sakao and Lindahl 2012). An approach for using of KPIs to evaluate PSS designs in a feedback loop during the design process was proposed by Mourtzis et al. (2015). However, they do not cover financial criteria as part of their evaluation. A comprehensive list of 94 evaluation criteria for PSS from existing literature and their categorization is provided by Kim et al. (2016).

In summary, it can be observed that there is considerable previous work on concepts, engineering and evaluation methods for PSS, of which smart services are a special form. While the importance of a financial case is stated by several authors, no concrete approach for the integrated financial assessment of smart service concepts could be found in the extant literature. Furthermore, with the exception of the approach proposed by Mourtzis et al. (2015), no contribution could be found, which explicitly integrates evaluation results into the design process through a feedback loop.

## Requirements

Through argumentation and reasoning, the requirements for a design-integrated evaluation tool were derived from the characteristics of smart services, the design process, and the research goals (Johannesson and Perjons 2014). The list of requirements is summarized in Table 1.

The references to the above-mentioned characteristics SC1 to SC5 and PC1 to PC4 indicate which of them were used to derive the requirements R1 to R7, for which the reasoning is as follows:

- R1: As identified in SC1, data transmission is required for IoT devices. It must therefore be considered as part of the

model. The cost for transmission is typically incurred in a usage-dependent manner and might also contain monthly subscription fees (SC5). The demand for data transmission is caused by the invocation of functions to provide the desired service, which is an example of interdependencies (PC3).

- R2: The bundling of functions to provide services enable modular system architectures as well as reuse of functionality into different target-group specific offers (SC3). Identifying these relations is a creative process in the early conceptual phase (PC2).
- R3: Both external services (SC2) and data transmission (SC1) incur costs for providing the required functionality, while revenue is driven by customer demand for offers (SC3). These demands need to be provided in a simple and easily modifiable way.
- R4: Pricing models for external services (SC3), data transmission (SC1) and offered services (SC2) may contain multiple components (SC5), which need to be part of the modeling approach.
- R5: Early stage evaluation is challenging due to incomplete information about the metrics and values required (PC1). While these are added and refined iteratively, the evaluation result should continuously be updated to reflect the current level of detail provided (PC4). This helps to see interdependencies between the parameters (PC3) and allows deciding on which cost might be borne by customers (SC4).
- R6: As early stage elaboration of service ideas into service concepts is an interactive process (PC2), the creativity, agility and iterative design need to be supported by the tool.
- R7: The issue of diversity of backgrounds among the members of the design team (PC1), possibly also including the customer (SC4), needs to be addressed by models that are easily comprehensible.

## Meta-model design

The core of the proposed approach is the creation and iterative refinement of a service model, which is annotated with parameters to instantly calculate the financial impact on every model update. This requires a meta-model which links the main elements of the service with its financial impact. For the development of the meta-model, the top-down analysis method was used (Grässle et al. 2005). It contains the steps 1 to 3. Step 4 is performed to verify whether the created meta-model can represent typical smart service cases.

1. Identify and model classes for key concepts, e.g. customers, devices, offers, functions, revenue, cost as UML classes

**Table 1** List of derived requirements

Characteristics	Requirement
SC1, PC3, SC5	R1. Data volumes for data transmission, e.g. for cell networks, must be calculated.
SC3, PC2	R2. Model elements need to be flexibly assigned, e.g. functions to services.
SC1, SC2, SC3	R3. Demands, quantities and usage intensity must be expressible in a simple manner.
SC1, SC2, SC3, SC5	R4. Various types of costs as well as pricing and billing models must be supported.
PC1, PC3, PC4, SC4	R5. Early assessment of the service must be possible, even with incomplete information.
PC2	R6. The addition and modification of elements, properties, relationships, quantities, prices, and costs must be possible in any order.
PC1, SC4	R7. Models must be comprehensible for experts from different disciplines.

2. Identify and model associations and cardinalities between classes to describe the relations between them
3. Identify and model attributes, e.g. prices, quantities, and cost in the respective UML classes
4. Test of the model using real-world cases of smart services for connected products
5. Repeat steps 1–4 until no further changes were required

As a meta-model provides the language with which a modeler can afterwards express a concrete case in a model instance, it needs to fulfill all characteristics defined by Stachowiak (1973): a model is a mapping, a reduction, and is pragmatic, i.e. serves a dedicated purpose. Therefore, a key challenge for a meta-model design are the same as with every other model: granularity vs. comprehensibility. A fine-grained model can express many aspects of a service and thus cover a wide range of possible scenarios. Given the fact, that our model is to be used for rough assessment in very early stages of conception, a high level of detail might neither be available nor needed. Therefore, we deliberately kept the meta-model simple and the number of modeling elements at a minimum. This also serves the purpose of better comprehension for diverse groups of people that take part in a service design project.

### Smart service meta-model

Modeling digital services has been proposed in different forms, e.g. by Yoo et al. (2010), Weinberger et al. (2016) and Porter and Heppelmann (2014). They all share the concept of dividing the service into different layers, with (digital) services as the top layer, connectivity, and data processing as the middle layer, and physical products with sensors, actuators etc. as the bottom layer - see Püschel et al. (2016) for a comparison. The service model proposed in the present paper uses the integrated consideration of business, functional and technical aspects as well as the layers as guiding ideas.

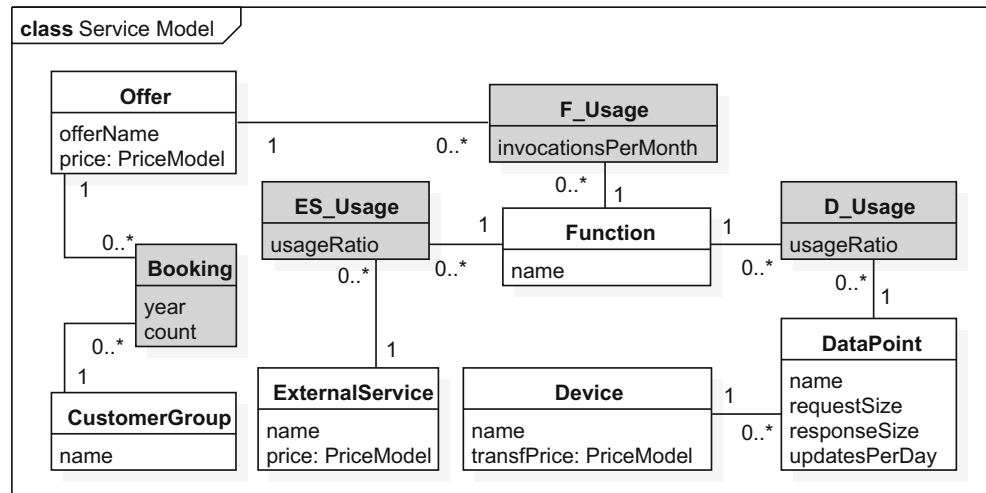
The service model (Fig. 3) describes the available model elements and their relationships. The basic elements (white) relate to each other using links (light gray) that provide attributes for quantification of the relationship. The desired easy

understandability (requirement R7) is to be achieved by a minimum number of elements and relationships. Since our research is exploratory, only the elements necessary for financial assessment (requirements R3 and R4) are included.

- *Offers* are targeted at groups of customers (*Customer Segment*) and have a *price*, which is defined using the *PriceModel*. The link between them is described using a set of *bookings*, which includes the attributes *year* and the number (*count*).
- The provision of an *Offer* requires *Functions* that represent software components with processing logic. Each function can be used by multiple offers. This results in an M:N relationship between services and functions (requirement R2), which is realized through a use of a function usage link (*F\_Usage*). The attribute *invocationsPerMonth* expresses the frequency of function calls per month by a service (requirement R3).
- Functions may be using data and operations provided by *devices* or *external services*, such as weather information, traffic information or SMS delivery. This dependency is modeled in the relationship *ES\_Usage*, where the percentage of function invocations leading to an external service call can be specified in the attribute *usageRatio*. The idea behind this is that not each call to a function requires external services, e. g. in 5% of the cases, an alarm message is sent out using SMS. To capture the cost of external services (requirement R4) information on the *price* can be stated; using the *PriceModel* explained below.
- Data and operations provided by a *device* are modeled as *data points*, which have a *requestSize* and *responseSize* to describe the transferred data in bytes during retrieval. Functions and data points are connected via *D\_Usage*, which contains the proportion of calls (*usageRatio*) that cause request/response communication. Additionally, push communication is expressed using the *updatesPerDay* attribute of a *data point* (requirement R1).

To represent various pricing schemes for services to offer and to consume (requirement R4), we propose the meta-model

**Fig. 3** Class diagram of the meta-model for the description of smart services



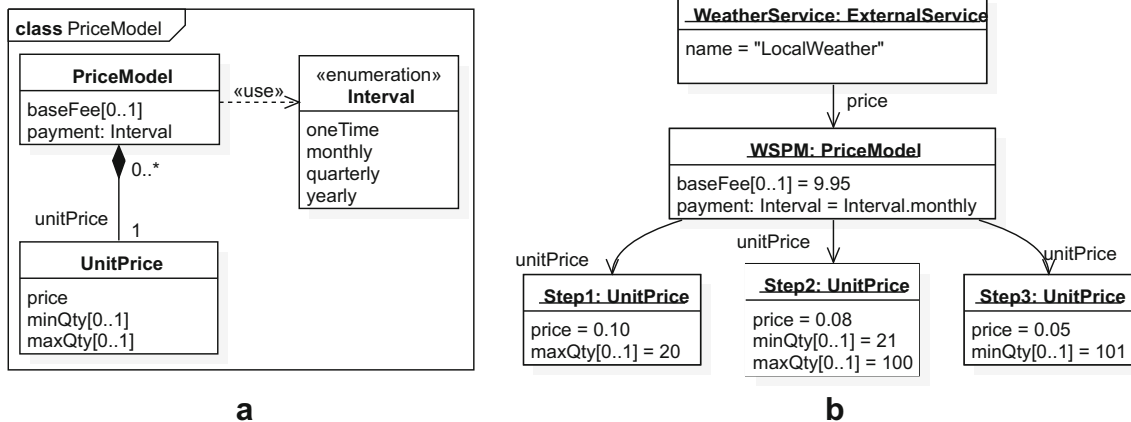
for pricing models depicted in Fig. 4a. It allows specifying an optional base fee, which is independent of the actual usage of the service, e.g. a subscription fee. The payment interval is expressed through the *Interval* enumeration. Additionally, one or more price levels can be defined by creating the required number of *UnitPrice* objects. The applicability of the price can be specified using the optional *minQty* and *maxQty* attributes, e.g. a price for a certain amount of transactions, bookings or data transmission volume. An example instantiation is a weather service with a monthly fee of 9.95 €, where the price per transaction drops from 0.10 € for 1 to 20 calls per month to 0.05 € if the service is invoked more than 100 times per month (Fig. 4b).

**Financial case meta-model**

A financial case is an instrument to investigate the profitability by providing decision makers a structured view on the returns for an investment. Common decision

criteria and methods are net present value (NPV), return on investment (ROI) or internal rate of return (IRR). Each of them requires a payment series, which consists of the difference between incoming and outgoing payments for a certain period, e.g. a year. The challenge we address in our research is to integrate the creation of this payment series in the design process of a smart service. For that, the cost of building and operating the service system as well as the revenue created through offering services to customers must be considered.

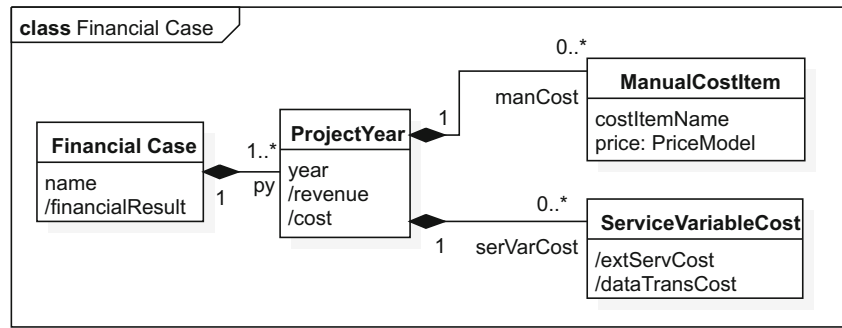
To assess the profitability, we propose a model which depicts the structure of payments for each planning year. It contains the *ServiceVariableCost*, which can be derived automatically from the service model and will be stored in the attributes *extServCost* and *dataTransCost* respectively (Fig. 5). Furthermore, it allows to specify additional costs manually, e.g. for the development and operation of the service system (*ManualCostItem*). They are specified through a *price*, which is expressed through the *PriceModel* structure introduced above (requirement A4).



**Fig. 4** a Class diagram for Price Model and (b) Example of instantiation



**Fig. 5** Class diagram for the financial case model



**Calculation of financial case**

To derive the financial case from the annotated service model, a service model and a financial case model must be instantiated. Therefore, a set of related objects of the classes defined in the meta-models for the project at hand is created. For the calculation, the following functions are defined for accessing the instantiated objects:

- $fUse(x, y)$ ..  $F\_Usage$  for Function  $x$  and Offer  $y$
- $eUse(x, y)$ ..  $ES\_Usage$  for Function  $x$  and External Service  $y$
- $dUse(x, y)$ ..  $D\_Usage$  for Function  $x$  and DataPoint  $y$
- $DP(x, y)$ ..  $DataPoint$   $x$  for Device  $y$
- $ES(x)$ .. External Service  $x$  in list of all external services
- $DE(x)$ .. Device  $x$  in list of all devices
- $BK(x,y,z)$ .. Booking for CustomerGroup  $y$  and Offer  $z$  in year  $x$

Additionally, a helper function  $getPrice(qty)$  is defined for *Offer*, *ManualCostItem*, *ExternalService*, and *Device*. It determines the price for a given quantity  $qty$ . at the reference object as defined by the *PriceModel*.

The external service cost consists of transactional cost and recurring cost for all required external services. As transactional cost are dependent on the actual usage of services, the number of invocations per external service is stored in the  $invocs()$  list. It is determined by iterating over all functions  $f$  and the number of invocations caused by the services stored in the  $invocationPerMonth$  attribute of each  $F\_Usage$  instance. For each invocation, a certain percentage specified by  $usageRatio$  leads to an external service call, which is charged with a transaction fee determined through the  $getPrice()$  function. Finally, the recurring cost of all required external services  $e$  are added. The factor  $intvl$  is used to convert payments during the year to yearly value; e.g.  $intvl$  will be 12 for monthly payments and 4 for quarterly payments.

$$invocs(ES(k)) = \sum_{i=1}^f eUse(i, k).usageRatio * \sum_{j=1}^o fUse(i, j).invocationsPerMonth * 12$$

$$extServCost = \sum_{k=0}^e invocs(ES(k)) * ES(k).getPrice(invocs(k)) + ES(k).price.baseFee * intvl$$

For the calculation of data transmission cost, push and pull communication mechanisms are considered separately. For the calculation of data volume in pull mode, the  $requestSize$  and  $responseSize$  (in byte) must be considered for every data point that is requested by a function using  $F\_Usage$  and  $D\_Usage$ . The data volume is then converted into megabyte and stored in the  $pullVol()$  list, which keeps the required data volume

for every device. For push communication, the  $updatesPerDay$  attribute of all data points  $p$  is evaluated and multiplied with the  $responseSize$  of each data point. The result is stored in the  $pushVol()$  list for each device. As described above, the push communication is independent of individual requests and takes place separately from pull communication, if a value for the  $updatesPerDay$  attribute is provided. To determine the

total transmission cost  $dC$ , the combined volume of push and pull communication is multiplied with the

price for the transmission of each device specified in a price model, which is retrieved using  $getPrice()$ .

$$pushVol(DE(j)) = 365 * \sum_{i=1}^p (DP(i, j).updatesPerDay * DP(i, j).responseSize) / 1048576$$

$$pullVol(DE(j)) = \sum_{i=1}^p ((DP(i, j).requestSize + DP(i, j).responseSize)) / 1048576 * \sum_{k=1}^f dUse(k, DP(i, j)).usageRatio * \sum_{m=1}^o fUse(k, m).invocationsPerMonth * 12$$

$$dC = \sum_{i=1}^d (pushVolume(DE(i)) + pullVol(DE(i)) * DE(i).getPrice(pushVol(DE(i)) + pullVol(DE(i))))$$

All other cost calculations are related to manual cost items. They are simply calculated by multiplying *quantity* and the *price* defined for this quantity with consideration of the payment interval through the appropriate *intvl* factor to convert into yearly cost, e. g factor 4 for quarterly occurring cost. While cost is modeled without relation to

a specific planning year, revenues can vary through the count attribute of the *Booking* class. Therefore, for the calculation of revenues, all bookings for all customer groups  $g$ , their assigned offers  $o$  in planning year  $i$  need to be considered and multiplied by the *price* for the respective offer.

$$rev(i) = \sum_{j=0}^g \sum_{k=0}^o BK(i, j, k).count * intvl * Offer(j).getPrice(BK(i, j, k).count * intvl)$$

The results of these calculations are stored in the attributes *revenue* and *cost* of a *PlanningYear* object. A set of *PlanningYear* objects within a *Project* is the payment series. For the sake of simplification, it is assumed that all revenues and cost are cash-effective in the respective period. Afterwards, established methods for capital budgeting like NPV or IRR can be applied to the payment series to determine the economic viability of the current service model (Pieroni et al. 2016).

## Application of the meta-model in a tool prototype

To facilitate easy collaboration within an interdisciplinary team as well as to allow storing results between multiple workshops, the implementation of the tool as a web-based application was devised. For the frontend, the JavaScript framework AngularJS was used, which communicates via REST APIs with backend services developed in C#. These, in turn, use a Microsoft SQL Server database to store the models as described in the previous sections. Based on the models created through user interaction in the Editor View, a

calculation component creates the financial case model and displays the result (“FC Result”) instantly in the Editor View after every modification (Fig. 6).

The tool allows the creation of projects and the configuration of their planning period. For each project, there is an

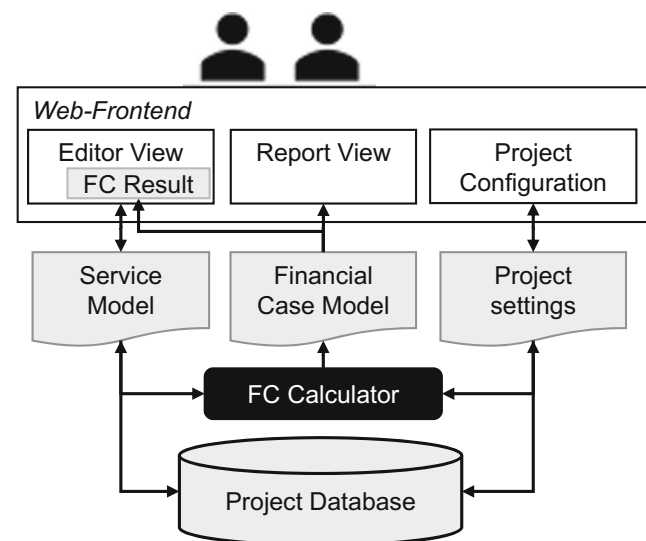
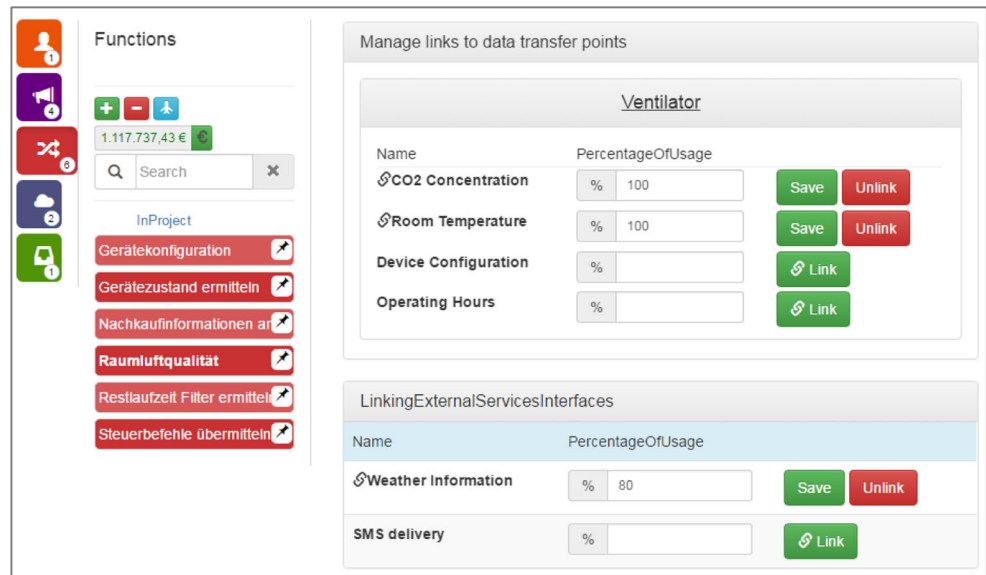


Fig. 6 Architecture of the prototype

Fig. 7 User Interface of the web-based tool prototype



overview page, from which the user can navigate to model editing, project configuration, and reporting. On the right-hand side, the model editing view provides navigation icons for customer groups, offers, functions, external services, and devices (requirement R6). Elements of the respective type can be added or deleted. On the right-hand side, there are input options for attributes of elements and manipulation of links (Fig. 7).

The application of the tool in a practical setting is depicted in Fig. 8. It allows the service structure to be built up in parallel with quantities, prices and usage behavior. The model can be iteratively manipulated and expanded as often as required. Each change leads to a recalculation of the financial case, which can be incorporated into the design process (Fig. 4). At the same time, it serves as a documentation of the development status over various workshop sessions and thus avoids the loss of important contributions. The tool prototype can also be used within a co-innovation process to include customers as important stakeholders as proposed by Marilungo et

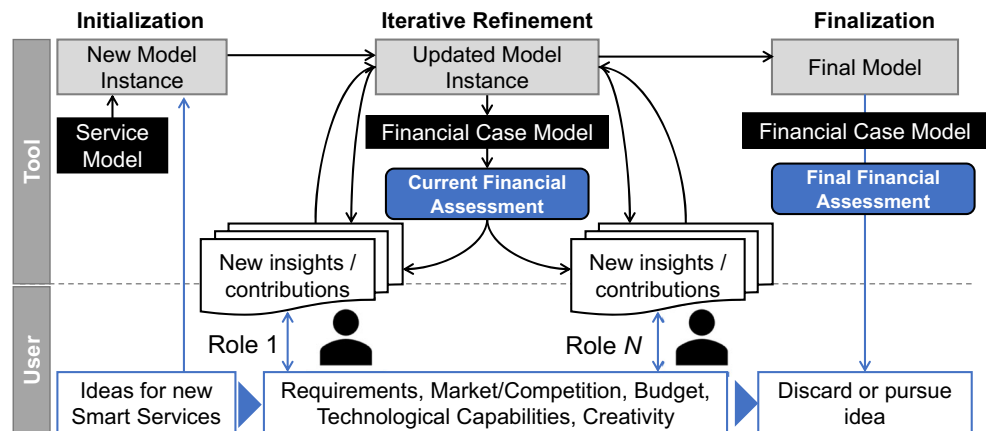
al. (2016). The revision is continued until the project team can decide on the continuation or rejection of the service idea.

## Evaluation

### Model evaluation

To evaluate models in general, the criteria *completeness*, *fidelity with real-world phenomena*, *internal consistency*, *level of detail*, and *robustness* are proposed by March and Smith (1995). *Fidelity with the real-world phenomena* refers to the external consistency, which was evaluated through the test of the model with real-world cases in the design phase. The *level of detail* and *completeness* are more difficult to evaluate as both the spectrum of potential cases as well as the information demand for the individual model users can be very different. An indirect evaluation of these two criteria is performed through the application of the model in the tool prototype as

Fig. 8 Application of the proposed tool in the design process



described below. *Internal consistency* and *robustness* of the meta-model were not evaluated. As this research aims to explore the overall approach of integrating financial evaluation in the design process, we argue that internal consistency and robustness should be considered for future research when the approach is more mature.

The specific evaluation of the meta-model is performed regarding the requirements in Table 1. Here, it can be stated that the requirements R1, R2, R3, and R4 are fulfilled to a large extent, as they relate to the structure of the model and were addressed in the design process. In terms of transferability, there will always be a debate on whether such meta-models provide too much or too little detail. Further evaluations steps are required to get a better understanding of the potential improvement on scope, expressiveness, and comprehensibility of the meta-models, especially for interdisciplinary teams.

### Design of an experiment for the tool evaluation

The tool prototype is an instantiation of the meta-model, for which potential evaluation criteria are *effectiveness*, *efficiency*, and *impact on the environment and the artifact's users* (Sonnenberg and Vom Brocke 2012). We follow the notion of Prat et al. (2014), according to which models are considered as abstract artifacts that can be indirectly evaluated through their instantiations. Therefore, we indirectly evaluate the proposed meta-model through the evaluation of the tool prototype based on it.

An experiment is used out to evaluate the tool, in which multiple teams carry out the design and evaluation of smart services. The aim of the experiment is to identify effects of the tool by comparing teams with and without the support of the tool prototype in a setting, where service ideas need to collaboratively be evolved into service concepts.

The participants were 30 information systems students of a German Cooperative State University in their final year of study. Due to their dual study model, which integrates academic studies with on-the-job training, they have a much higher level of practical experience compared to “typical” students. As preparation, all of them received a two-hour introduction to the topic “Internet of Things and Smart Services”. The basic structure of smart services and the basics of financial evaluation were explained without presenting the tool in detail.

This was followed by a brainstorming of 30 min to generate ideas for smart services, from which a total of four ideas were selected for the experiment. We deliberately chose not to use predefined service ideas or scenarios for two reasons: First, in a creative workshop setting, it is common that many ideas are generated from which only a small fraction is chosen for further elaboration. Second, the brainstorming utilizes existing knowledge of the participants, which directs the generation

of ideas to domains, they are more familiar with than with externally prepared service ideas.

Afterwards, they were randomly divided into eight teams with three to four members. Due to the randomization of the assignment between experimental and control group, a pretest is dispensable (Wilde 2008). Therefore, the “Posttest-only control group design” (Recker 2013) was chosen. Each service idea was assigned to a team of the experimental group (EG) with access to the tool prototype and a team of the control group (CG) without access to the tool prototype. Participants of the CG were however allowed to use other software. All teams in the control group decided to use spreadsheet software, in most cases MS Excel. It should be noted that there was no spreadsheet model provided. This was decided for two reasons: First, to evaluate the benefit of the proposed meta-model. Second, in a real-world situation, the spreadsheet model for financial evaluation would not be available upfront but would have to be developed by the team.

The design task given to the participants was: “(a) Elaborate and describe the assigned scenario in detail with its target customer groups, offers, functions, and data. (b) Assess the profitability based on cost and revenue with a planning horizon of three years.” Part (a) of the task specifically refers to the structure of smart services. Part (b) of the task is a general profitability question, which can be placed in a similar way for every project proposal or investment object. However, as the profitability is asked for at an early stage of service development, i.e. conceptual elaboration, it creates a setting which is specific to our research question.

Each team had up to 75 min to complete the task. Immediately afterwards, the participants were asked to complete a survey to assess their experience with the task and the utility of the tool support. The overall design of the experiment is shown in Fig. 9.

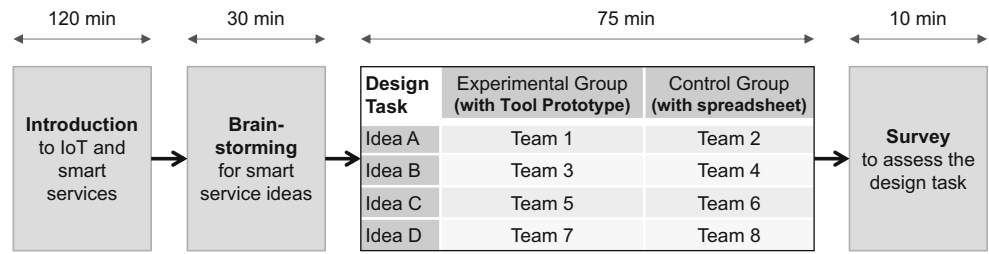
For the assessment, a set of criteria was established in a questionnaire, which relates back to the research goals as well as the requirements stated above. It allowed participants to rate their experience with the design task. Table 2 shows how the statements in the questionnaire relate to the evaluation criteria of instantiations.

All statements could be rated on a Likert scale from 5 (strongly agree), 4 (agree), 3 (neutral), 2 (disagree) to 1 (strongly disagree). As participants in the CG were unable to rate the criterion TOOL\_USEFUL, an additional option 0 („I did not use the tool“) was added to this particular statement.

### Results of the tool evaluation

All 30 participants completed the survey. *For all variables, the median was determined for the EG and CG. Since the Likert scale can also be interpreted as interval-scale, the arithmetic mean and the standard deviation is also given for better differentiation of the results (Table 3).*

**Fig. 9** Design of the experiment to evaluate the prototype



### Discussion

First, it should be noted that EG participants have considered the use of the tool to be very helpful. Furthermore, support for structuring the task was positively assessed. This can be interpreted as an indication for the basic comprehensibility (requirement R7) of the developed service model, at least for participants with an information systems background.

All other variables have nearly identical ratings in both groups. The reason for this could be that the utility of the tool increases with larger groups, longer processing time or when using it in a series of multiple workshops. Furthermore, it would have to be examined in future runs of the experiment whether an improvement in the results is achieved with a more precise introduction into the functioning of the tool. We deliberately kept the introduction of using the tool rather short, as we assumed that in practice people would not spend much time on training. Related to this is the usability of the tool prototype. The prototype is only a mean to make the meta-model usable in practice. While we tried to comply with established standards of web-based applications regarding layout, navigation and interaction, we did not have usability as an explicit design goal. It, therefore, can be assumed that improving the usability would increase the tool’s utility without changing the meta-model in any way.

Regarding the tool-support of the design process, we can state that R6 was also fulfilled as the sequence for manipulation of model elements is not restricted in any way. As the complexity of the meta-model was kept deliberately low and

the evaluation indicated a benefit for structuring the task, we can assume the R7 to be fulfilled as well.

The measurement of the variable IMPACT is particularly relevant regarding the fulfillment of requirement R5. As the evaluation showed no difference between EG and CG, further research is needed to investigate whether this requirement can be fulfilled. One strategy could be to observe the individual modeling steps in both EG and CG and compare at which points in time intermediate results were achieved.

While our results indicate that the general approach of integrating evaluation in the design processes is helpful to better manage complexity, we acknowledge the limitations of our research: First, the experiment is conducted with students. While they have a very good state of knowledge and some practical experience due to the dual study model, it is likely that results will be different for professionals with more experience in designing complex IT-based services. Second, the length of the conceptual work in the experiment was relatively short and conducted in a single session. Real-world service engineering projects are very likely to be conducted in multiple sessions over a longer period, even with changes in the team structure. Third, the training with the tool might have been too little. More complex cases or repeated usage in the real-world would justify a more intensive training to familiarize users with the concept and functionality in more detail.

Finally, we have not evaluated the internal consistency and robustness of the meta-model. As stated above, we see these

**Table 2** Questionnaire statements and their relation to evaluation criteria

Statement	Variable	Effectiveness	Efficiency	Impact on Environment and Artifact’s Users
It was easy for me to find a structure for the design task.	STRUCTURE		X	
It was easy to make decisions about the service design.	DECISION		X	
The impact of our decisions on the profit of the service was easy to estimate.	IMPACT		X	
I felt that everyone in the team had the same view on the current state of work.	PROGRESS	X		X
I am satisfied with the result of our work.	RESULT	X		X
Using the tool has helped me with the design task.	TOOL_USEFUL		X	X

**Table 3** Results of the evaluation ( $N = 30$ )

Variable	EG (with Tool Prototype)			CG (with spreadsheet)		
	Median	Mean	Std. dev.	Median	Mean	Std. dev.
STRUCTURE	4.0	3.88	0.885	3.0	3.21	1.051
DECISION	3.5	3.50	0.894	3.5	3.57	1.016
IMPACT	3.0	2.69	1.014	3.0	2.79	1.051
PROGRESS	4.0	3.79	0.998	4.0	3.94	0.975
RESULT	3.0	3.38	1.204	3.0	3.21	1.251
TOOL_USEFUL	4.0	4.21	0.696	–		

evaluations as part of future research, once the overall approach of design integrated financial evaluation is more mature and the meta-models are stable.

## Conclusions

In this article, we have presented an approach for tool-based support of the financial analysis of smart services, which is integrated into the design process. As we have discussed, most PSS engineering methods are particularly concerned with the design process and lack of explicit support for IT artifacts as part of the PSS. Furthermore, while there are design-integrated methods for evaluation of PSS (Mourtzis et al. 2015), they do not cover financial impact. Existing approaches for tools are typically too complex for an early stage evaluation of service ideas (Becker et al. 2009; Laurischkat 2013).

We addressed these deficits by providing a meta-model as the basis for a tool to allow immediate calculation of evaluation results based on the proposed meta-model. The objectives were to enable financial evaluation in early design stages and to help interdisciplinary project teams to collaborate in the design and evaluation process. Our solution is suitable for interactive use in workshops as it allows instant feedback in an iterative design process. Based on the characteristics of smart services as complex systems and their engineering process, we derived seven requirements. After designing meta-models for both the service and the financial case, we implemented a tool prototype based on these models. From our experimental evaluation, we can derive the following findings:

1. *Using the tool prototype is not obstructive:* None of the participants in the experimental group showed a lower ranking of utility criteria than the control group. This indicates that using the tool was not hindering the design process within the experiment.
2. *Tool-support is perceived as helpful:* Participants of the experimental group showed a strong option towards the general benefit of using the tool (median of 4 on a 1 to 5

scale). This implies that the participants explicitly saw not only a benefit of having a tool in general but perceived this particular tool prototype as helpful for the task at hand. This indicates that the general approach of tool-based design was accepted and appreciated within the group of participants.

3. *Structuring of the task is a major benefit:* The results of the experiment indicate that a tool with the underlying meta-model is helpful to describe a smart service. This indirectly provides an evaluation of the suitability of the model for the given task.

The theoretical contributions of this paper to the body of knowledge in model-based service engineering are as follows: First, we introduced the concept of design-integrated financial evaluation for smart services, which addresses the identified research gap of missing concrete methods for financial evaluation in PSS engineering. Secondly, we developed a data structure (meta-model), which enables the early stage modeling of smart services, and its link to a financial case model. This link is established through a calculation model provided as a set of formulae. The meta-model can only be understood as a first proposal on how to integrate design and financial evaluation. Especially the flexibility of the meta-model regarding different business models is currently rather low. A more comprehensive test with smart services in different industries and with varying level of complexity will offer insights into shortcomings of the current meta-model.

Regarding practical contributions, there is an indication that our approach can help to improve the design process of smart services through tools. For that, empirical evidence on the utility of the design-integrated evaluation of smart services was collected, which indicates that the proposed approach is beneficial. Additionally, we showed the technical feasibility of implementing a working tool from our concept. Both the meta-model and the presented architecture can serve as a foundation for the development of similar tools for similar purposes. While tool-support for complex tasks such as smart service design and evaluation appears to be rather obvious, creating actual benefits from it depends on many factors such as qualification of users, complexity of the problem, and

usability of the tool. Most of them were not explicitly investigated in this study but will be relevant in practice.

Empirically, we could find comparatively small improvements caused by the tool in the design process. Future research should focus on the elaboration of conditions under which such a tool use becomes more effective. This refers to the “Eval4 activity”, as proposed by Sonnenberg and Vom Brocke (2012), which aims to evaluate applicability and usefulness of artifacts in practice, e.g. through field experiments, case studies, and expert interviews.

Future research topics are usability improvements and a graphical notation for the model. Extension to the modeling capabilities of the meta-model should also be considered. For example, the meta-model does not provide any means to express the sequencing of function to model processes. It should be investigated, whether this is an important addition to get a more detailed analysis of cost or whether the added complexity counters intuitive use.

In general, the technical and organizational integration of a smart service modeling and evaluation tool as part of the overall service engineering process is still an open question. For that, further research needs to be conducted to better understand the acceptance of tool-based business modeling and evaluation schemes. Furthermore, the reuse of created service models in other tools for more advanced design stages of digital product-service systems (McKay and Kundu 2014) needs to be address to enable integrated tool-chains. Research regarding these topics will be highly beneficial to both research and practice, as the economic relevance of smart services will continue to increase.

## References

- Adrodegari, F., Alghisi, A., Ardolino, M., & Saccani, N. (2015). From ownership to service-oriented business models: A survey in capital goods companies and a PSS typology. *Procedia CIRP*, 30, 245–250. <https://doi.org/10.1016/j.procir.2015.02.105>.
- Alam, I., & Perry, C. (2002). A customer-oriented new service development process. *Journal of Services Marketing*, 16(6), 515–534. <https://doi.org/10.1108/08876040210443391>.
- Allmendinger, G., & Lombreglia, R. (2005). Four strategies for the age of smart services. *Harvard Business Review*, 83(10), 131.
- Aurich, J. C., Fuchs, C., & Wagenknecht, C. (2006). Life cycle oriented design of technical product-service systems. *Journal of Cleaner Production*, 14(17), 1480–1494. <https://doi.org/10.1016/j.jclepro.2006.01.019>.
- Azizyan G, Magarian MK, Kajko-Matsson M (2011) Survey of agile tool usage and needs. In: AGILE Conference. IEEE, pp 29–38.
- Baines, T. S., Lightfoot, H. W., Evans, S., Neely, A., Greenough, R., Peppard, J., . . . Wilson, H. (2007). State-of-the-art in product-service systems. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 221(10), 1543–1552. <https://doi.org/10.1243/09544054JEM858>.
- Baines, T. S., Lightfoot, H. W., Benedettini, O., & Kay, J. M. (2009). The servitization of manufacturing: A review of literature and reflection on future challenges. *Journal of Manufacturing Technology Management*, 20(5), 547–567.
- Barbosa J, Leitão P, Trentesaux D, Karnouskos S (2016) Cross Benefits from Cyber-Physical Systems and Intelligent Products for Future Smart Industries. In: Proceedings of the 14<sup>th</sup> IEEE International Conference on Industrial Informatics.
- Barrett, M., Davidson, E., Prabhu, J., & Vargo, S. L. (2015). Service innovation in the digital age: Key contributions and future directions. *MIS Quarterly*, 39(1), 135–154.
- Becker, J., Beverungen, D., Knackstedt, R., & Müller, O. (2009). Model-based decision support for the customer-specific configuration of value bundles. Advance online publication. <https://doi.org/10.18417/emisa.4.1.3>.
- Bilgeri, D., Brandt, V., Lang, M., Tesch, J., & Weinberger, M. (2015). The IoT Business Model Builder. *Bosch IoT Lab White Paper*. Retrieved from [https://www.boschsi.com/media/bosch\\_si/services/iot\\_business\\_consulting/whitepaper\\_iod-business-model-builder.pdf](https://www.boschsi.com/media/bosch_si/services/iot_business_consulting/whitepaper_iod-business-model-builder.pdf)
- Böhmman, T., Leimeister, J. M., & Möslin, K. (2014). Service systems engineering. *Business & Information Systems Engineering*, 6(2), 73–79. <https://doi.org/10.1007/s12599-014-0314-8>.
- Brady, M. K., & Cronin, J. J. (2001). Customer orientation. *Journal of Service Research*, 3(3), 241–251. <https://doi.org/10.1177/109467050133005>.
- Carpanen, P., Patricio, L., & Ribeiro, B. (2016). Designing Product Service Systems in the Context of Social Internet of Things. In T. Borangiu, M. Dragoicea, & H. Nóvoa (Eds.), *Lecture Notes in Business Information Processing: Vol. 247. Exploring Services Science: 7<sup>th</sup> International Conference, IESS 2016, Bucharest, Romania, May 25-27, 2016, Proceedings (Vol. 247, pp. 419–431)*. Cham, s.l.: Springer International Publishing.
- Caulkins, J. P., Morrison, E. L., & Weidemann, T. (2007). Spreadsheet errors and decision making. *Journal of Organizational and End User Computing*, 19(3), 1–23. <https://doi.org/10.4018/joeuc.2007070101>.
- Cavaliere, S., & Pezzotta, G. (2012). Product–Service Systems Engineering: State of the art and research challenges. *Product Service System Engineering: from Theory to Industrial Applications Product Service System Engineering: from Theory to Industrial Applications*, 63(4), 278–288. <https://doi.org/10.1016/j.compind.2012.02.006>.
- Columbus L (2017) Internet Of Things Market To Reach \$267B By 2020. <https://www.forbes.com/sites/louisacolumbus/2017/01/29/internet-of-things-market-to-reach-267b-by-2020/#c5804d609bd6>. Accessed 18 September 2017.
- Craven, M. P., Morgan, S. P., Crowe, J. A., & Lu, B. (2013). Deploying a spreadsheet tool for early economic value assessment of medical device innovations with healthcare decision makers. *Journal of Management & Marketing in Healthcare*, 2(3), 278–292. <https://doi.org/10.1179/mmh.2009.2.3.278>.
- Dewit I, Roeck D de, & Baelus C (2014) Roadmap and toolbox for the ideation stage of the development process of product Service Systems. In 16<sup>th</sup> International conference on Engineering and Product Design Education. <https://doi.org/10.13140/2.1.2375.1045>
- Edvardsson, B., Tronvoll, B., & Gruber, T. (2011). Expanding understanding of service exchange and value co-creation: A social construction approach. *Journal of the Academy of Marketing Science*, 39(2), 327–339. <https://doi.org/10.1007/s11747-010-0200-y>.
- Ehrenhofer C, Kreuzer E (2012) The role of business model Design in the Service Engineering Process: A comparative case study in the field of cloud computing to join service engineering with business model design. In: SRII Global Conference (SRII), 2012 Annual. [publisher not identified], [Place of publication not identified], pp 283–292.
- Fischer, T., Gebauer, H., & Fleisch, E. (2012). *Service business development: Strategies for Value Creation in Manufacturing Firms*. Cambridge: Cambridge University Press.

- Fleisch, E., Weinberger, M., & Wortmann, F. (2015). Business models and the internet of things. In I. Podnar Žarko, K. Pripužić, & M. Serrano (Eds.), *Interoperability and open-source solutions for the internet of things: International workshop, FP7 OpenIoT project, vol 9001* (pp. 6–10).
- Georgakopoulos, D., & Jayaraman, P. P. (2016). Internet of things: From internet scale sensing to smart services. *Computing*, 98(10), 1041–1058. <https://doi.org/10.1007/s00607-016-0510-0>.
- Grässle P, Smith C, Baumann H, Baumann P, Chakrabarti P (2005) UML 2.0 in action: A project based tutorial. Packt Pub, Birmingham, U.K.
- Grossman, T. A., Mehrotra, V., & Özlük, Ö. (2007). Lessons from mission-critical spreadsheets. *Communications of the Assoc. for Information Systems*, 20(1), 60.
- Herterich, M., Uebernickel, F., & Brenner, W. (2015). The impact of cyber-physical systems on industrial Services in Manufacturing. *Procedia CIRP*, 30, 323–328. <https://doi.org/10.1016/j.procir.2015.02.110>.
- Herterich M, Eck A, Uebernickel F (2016) Exploring how Digitized Products enable Industrial Service Innovation - An Affordance Perspective. In: Proceedings of the 24<sup>th</sup> European Conference on Information Systems.
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105.
- Isaksson, O., Larsson, T. C., & Johansson, P. (2011). Towards a framework for developing product/service systems. In J. Hesselbach & C. Herrmann (Eds.), *Functional thinking for value creation* (pp. 44–49). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Johannesson, P., & Perjons, E. (2014). Define Requirements. In P. Johannesson & E. Perjons (Eds.), *An introduction to design science* (pp. 103–116). Cham: Springer International Publishing.
- Kambanou, M. L., & Lindahl, M. (2016). A literature review of life cycle costing in the product-service system context. *Procedia CIRP*, 47, 186–191. <https://doi.org/10.1016/j.procir.2016.03.054>.
- Kees A, Oberlaender AM, Roeglinger M, Rosemann M (2015) Understanding the Internet of Things: A Conceptualisation of Business-to-Thing (B2T) Interactions. In: Proceedings of the Twenty-Third European Conference on Information Systems (ECIS 2015), Münster.
- Kim T, Bae D (2012) Stakeholder Conflict Resolving Model in PSS Design. In: Cumulus Helsinki Conference.
- Kim, S., Son, C., Yoon, B., & Park, Y. (2015). Development of an innovation model based on a service-oriented product service system (PSS). *Sustainability*, 7(11), 14427–14449. <https://doi.org/10.3390/su71114427>.
- Kim, K.-J., Lim, C.-H., Heo, J.-Y., Lee, D.-H., Hong, Y.-S., & Park, K. (2016). An evaluation scheme for product-service system models: Development of evaluation criteria and case studies. *Service Business*, 10(3), 507–530. <https://doi.org/10.1007/s11628-015-0280-3>.
- Klein, R., Herrmann, K., Scheer, A.-W., & Spath, D. (Eds.). (2004). *Computer aided service engineering*. Berlin: Springer Berlin Heidelberg.
- Kortuem, G., Kawsar, F., Fitton, D., & Sundramoorthy, V. (2010). Smart objects as building blocks for the internet of things. *IEEE Internet Computing*, 14(1), 44–51. <https://doi.org/10.1109/MIC.2009.143>.
- Laatikainen G, Ojala A, Mazhelis O (2013) Cloud Services Pricing Models. In: Herzwurm G, Margaria T (eds) Software business: From physical products to software services and solutions : 4<sup>th</sup> International Conference, ICSOB 2013, Potsdam, Germany, June 11–14, 2013 : proceedings, vol 150. Springer, Berlin, pp 117–129.
- Laudien, S. M., & Daxböck, B. (2016). The influence model of the industrial internet of things on business model design: A qualitative-empirical analysis. *International Journal of Innovation Management*, 20(08), 1640014. <https://doi.org/10.1142/S1363919616400144>.
- Laurischkat, K. (2013). Computer-aided Service Design for the Development of product-service systems – Motivation and benefits. In H. Meier (Ed.), *Product-service integration for sustainable solutions: Proceedings of the 5th CIRP international conference on industrial product-service systems* (pp. 547–560). Berlin: Springer.
- Leitão, P., Rodrigues, N., Barbosa, J., Turrin, C., & Pagani, A. (2015). Intelligent products: The grace experience. *Control Engineering Practice*, 42, 95–105. <https://doi.org/10.1016/j.conengprac.2015.05.001>.
- Lim, C.-H., Kim, M.-J., Heo, J.-Y., & Kim, K.-J. (2015). Design of informatics-based services in manufacturing industries: Case studies using large vehicle-related databases. *Journal of Intelligent Manufacturing*, 50(4), 181. <https://doi.org/10.1007/s10845-015-1123-8>.
- Lin, F.-R., & Hsieh, P.-S. (2011). A SAT view on new service development. *Service Science*, 3(2), 141–157. <https://doi.org/10.1287/serv.3.2.141>.
- Luong, N. C., Hoang, D. T., Wang, P., Niyato, D., Kim, D. I., & Han, Z. (2016). Data collection and wireless communication in internet of things (IoT) using economic analysis and pricing models: A survey. *IEEE Communication Surveys and Tutorials*, 18(4), 2546–2590. <https://doi.org/10.1109/COMST.2016.2582841>.
- Maglio PP, Lim C-H (2016) Innovation and big data in smart service systems. *Journal of Innovation Management*, 4(1), 11–21.
- Maglio, P. P., & Spohrer, J. (2013). A service science perspective on business model innovation. *Industrial Marketing Management*, 42(5), 665–670. <https://doi.org/10.1016/j.indmarman.2013.05.007>.
- March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, 15(4), 251–266. [https://doi.org/10.1016/0167-9236\(94\)00041-2](https://doi.org/10.1016/0167-9236(94)00041-2).
- Marilungo, E., Coscia, E., Quaglia, A., Peruzzini, M., & Germani, M. (2016). Open innovation for ideating and designing new product service systems. *Procedia CIRP*, 47, 305–310. <https://doi.org/10.1016/j.procir.2016.03.214>.
- Marilungo, E., Papetti, A., Germani, M., & Peruzzini, M. (2017). From PSS to CPS design: A real industrial use case toward industry 4.0. *Procedia CIRP*, 64, 357–362. <https://doi.org/10.1016/j.procir.2017.03.007>.
- Marques, P., Cunha, P. F., Valente, F., & Leitão, A. (2013). A methodology for product-service systems development. *Procedia CIRP*, 7, 371–376. <https://doi.org/10.1016/j.procir.2013.06.001>.
- Marques, C. A. N., Mendes, G. H. S., Oliveira, M. G., & Rozenfeld, H. (2016). Comparing PSS design models based on content analysis. *Procedia CIRP*, 47, 144–149. <https://doi.org/10.1016/j.procir.2016.03.068>.
- McKay, A., & Kundu, S. (2014). A representation scheme for digital product service system definitions. *Advanced Engineering Informatics*, 28(4), 479–498. <https://doi.org/10.1016/j.aei.2014.07.004>.
- Medina-Borja, A. (2015). Editorial column—Smart things as service providers: A call for convergence of disciplines to build a research agenda for the Service Systems of the Future. *Service Science*, 7(1), ii–iv. <https://doi.org/10.1287/serv.2014.0090>.
- Mikusz, M. (2014). Towards an understanding of cyber-physical systems as industrial software-product-service systems. *Procedia CIRP*, 16, 385–389. <https://doi.org/10.1016/j.procir.2014.02.025>.
- Mourtzis, D., Fotia, S., & Doukas, M. (2015). Performance indicators for the evaluation of product-service systems design: A review. In S. Umeda, M. Nakano, H. Mizuyama, H. Hibino, D. Kiritsis, & G. von Cieminski (Eds.), *Innovative production management towards sustainable growth, vol 460* (pp. 592–601). Cham: Springer.
- Neely, A. (2008). Exploring the financial consequences of the servitization of manufacturing. *Operations Management Research*, 1(2), 103–118. <https://doi.org/10.1007/s12063-009-0015-5>.
- Niemöller, C., Özcan, D., Metzger, D., & Thomas, O. (2014). Towards a Design Science-Driven Product-Service System Engineering Methodology. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, A. Kobsa, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, D. Terzopoulos, D. Tygar, G. Weikum, M. C. Tremblay, D. VanderMeer, M. Rothenberger, A. Gupta, & V. Yoon (Eds.), *Advancing the Impact of Design Science: Moving from Theory to Practice, vol 8463* (pp. 180–193). Cham: Springer International Publishing.
- Omami, I. (2003). Product service systems and their impacts on sustainable development. *Frontiers*, 2, 1.
- Panko, R. R., & Aurigemma, S. (2010). Revising the Panko-Halverson taxonomy of spreadsheet errors. *Decision Support Systems*, 49(2), 235–244.



- Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-122240302>.
- Peruzzini M, Marilungo E, Germani M (2015) Technical-business design methodology for PSS. In: ISPE CE, pp 513–522.
- Pezzotta, G., Pirola, F., Pinto, R., Akasaka, F., & Shimomura, Y. (2015). A service engineering framework to design and assess an integrated product-service. *Mechatronics*, 31, 169–179. <https://doi.org/10.1016/j.mechatronics.2015.05.010>.
- Pieroni, M., Marques, C., Campese, C., Guzzo, D., Mendes, G., Costa, J., Rosa, M., MGD, O., Macul, V., & Rozenfeld, H. (2016). Transforming a traditional product offer into PSS: A practical application. *Procedia CIRP*, 47, 412–417. <https://doi.org/10.1016/j.procir.2016.03.036>.
- Porter ME, Heppelmann JE (2014) How smart, connected products are transforming competition. Harvard business review. Retrieved from <https://hbr.org/2014/11/how-smart-connected-products-aretransforming-competition>.
- Prat N, Comyn-Wattiau I, Akoka J (2014) Artifact Evaluation in Information Systems Design Science Research ? A Holistic View. In: Proceeding of the 18<sup>th</sup> Pacific Asia Conference on Information Systems.
- Püschel L, Schlott H, Röglinger M (2016) What's in a Smart Thing? Development of a Multi-layer Taxonomy. In: Proceedings of the 37<sup>th</sup> International Conference on Information Systems (ICIS).
- Recker J (2013) Scientific research in Information Systems: A beginner's guide. Progress in IS. Springer, Berlin, Heidelberg.
- Reschenhofer, T., & Matthes, F. (2015). An Empirical Study on Spreadsheet Shortcomings from an Information Systems Perspective. In W. Abramowicz (Ed.), *18<sup>th</sup> International Conference on Business information systems, vol 208* (pp. 50–61). Cham: Springer.
- Sakao, T., & Lindahl, M. (2012). A value based evaluation method for product/service system using design information. *CIRP Annals - Manufacturing Technology*, 61(1), 51–54. <https://doi.org/10.1016/j.cirp.2012.03.108>.
- Sas, D., & Lindström, J. (2014). Advancing development of product-service systems using ideas from functional product development. *Procedia CIRP*, 21, 242–246. <https://doi.org/10.1016/j.procir.2014.03.170>.
- Scherer, J. O., Kloeckner, A. P., Ribeiro, J. L. D., Pezzotta, G., & Pirola, F. (2016). Product-service system (PSS) design: Using design thinking and business analytics to improve PSS design. *Procedia CIRP*, 47, 341–346. <https://doi.org/10.1016/j.procir.2016.03.062>.
- Scholz, S., Correia, A. T., & Stokic, D. (2016). Novel tools for product-service system engineering. *Procedia CIRP*, 47, 120–125. <https://doi.org/10.1016/j.procir.2016.03.237>.
- Shaw N (2014) Looking ahead: Is this the future? Canyon reveal new project bike. <http://enduro-mtb.com/en/looking-ahead-is-this-the-future-canyon-presents-a-new-project-bike/>. Accessed 25 November 2016.
- Sonnenberg, C., & Vom Brocke, J. (2012). Evaluation Patterns for Design Science Research Artefacts. In M. Helfert & B. Donnellan (Eds.), *Practical aspects of design science: European Design Science Symposium, EDSS 2011, Leixlip, Ireland, October 14, 2011, vol 286* (pp. 71–83). New York: Springer.
- Stachowiak, H. (1973). *Allgemeine Modelltheorie*. New York: Springer.
- Stiller, B., Barlet-Ros, P., Cushnie, J., Domingo-Pascual, J., Hutchison, D., Lopes, R., Mauthe, A., Popa, M., Roberts, J., Solé-Pareta, J., Trcek, D., & Veciana, C. (2003). Pricing and QoS. In M. Smimov, E. Biersack, C. Blondia, O. Bonaventure, O. Casals, G. Karlsson, G. Pavlou, B. Quoitin, J. Roberts, I. Stavrakakis, B. Stiller, P. Trimintzios, & P. van Mieghem (Eds.), *Quality of future Internet services: COST Action 263 final report* (pp. 263–292). Berlin: Springer.
- Thoben, K.-D., Eschenbächer, J., & Jagdev, H. S. (2003). Emerging concepts in E-business and extended products. In J. Gasós & K.-D. Thoben (Eds.), *E-business applications: Technologies for Tomorrow's solutions* (pp. 17–37). Berlin: Springer Berlin Heidelberg; imprint; Springer.
- Tukker, A. (2004). Eight types of product-service system: Eight ways to sustainability? Experiences from SusProNet. *Business Strategy and the Environment*, 13(4), 246–260. <https://doi.org/10.1002/bse.414>.
- Vasanth, G. A. V., Roy, R., Lelah, A., & Brissaud, D. (2012). A review of product-service systems design methodologies. *Journal of Engineering Design*, 23(9), 635–659. <https://doi.org/10.1080/09544828.2011.639712>.
- Velamuri, V. K., Bansemir, B., Neyer, A.-K., & Möslin, K. M. (2013). Product service systems as a driver for business model innovation: Lessons learned from the manufacturing industry. *International Journal of Innovation Management*, 17(01), 1340004. <https://doi.org/10.1142/S1363919613400045>.
- Visnjic I, Jovanovic M, Neely A, Engwall M (2016) What brings the value to outcome-based contract providers?: Value drivers in outcome business models. *International Journal of Production Economics*. Advance online publication. <https://doi.org/10.1016/j.ijpe.2016.12.008>.
- Wallin, J. (2013). Turning internal product knowledge into external service offers: Building PSS capabilities. In Y. Shimomura & K. Kimita (Eds.), *The Philosopher's stone for sustainability: Proceedings of the 4<sup>th</sup> CIRP International Conference on Industrial Product-Service Systems, Tokyo, Japan, November 8<sup>th</sup> - 9<sup>th</sup>, 2012* (pp. 327–332). Berlin: Springer.
- Weinberger M, Bilgeri D, Fleisch E (2016) IoT business models in an industrial context. *at - Automatisierungstechnik* 64(9). <https://doi.org/10.1515/auto-2016-0054>.
- Wiesner, S., & Thoben, K.-D. (2017). Cyber-physical product-service systems. In S. Biffl, A. Lüder, & D. Gerhard (Eds.), *Multi-disciplinary engineering for cyber-physical production systems*. Cham: Springer International Publishing.
- Wiesner, S., Winkler, M., Eschenbächer, J., & Thoben, K.-D. (2013). Strategies for extended product business models in manufacturing service ecosystems. In H. Meier (Ed.), *Product-service integration for sustainable solutions: Proceedings of the 5<sup>th</sup> CIRP international conference on industrial product-service systems* (pp. 239–250). Berlin: Springer.
- Wilde T (2008) Experimentelle Forschung in der Wirtschaftsinformatik: Analyse des Methodenpotenzials und Entwicklung geeigneter Experimentaldesigns. Schriftenreihe Studien zur Wirtschaftsinformatik, Bd. 27. Kovač, Hamburg.
- Wortmann, F., & Flüchter, K. (2015). Internet of things. *Business & Information Systems Engineering*, 57(3), 221–224. <https://doi.org/10.1007/s12599-015-0383-3>.
- Wuenderlich, N. V., Heinonen, K., Ostrom, A. L., Patricio, L., Sousa, R., Voss, C., & Lemmink, J. G. A. M. (2015). “Futurizing” smart service: Implications for service researchers and managers. *Journal of Services Marketing*, 29(6/7), 442–447. <https://doi.org/10.1108/JSM-01-2015-0040>.
- Yang, X., Moore, P., Pu, J.-S., & Wong, C.-B. (2009). A practical methodology for realizing product service systems for consumer products. *Computers & Industrial Engineering*, 56(1), 224–235. <https://doi.org/10.1016/j.cie.2008.05.008>.
- Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research*, 21(4), 724–735. <https://doi.org/10.1287/isre.1100.0322>.
- Zolnowski, A., & Böhm, T. (2013). Veränderungstreiber service-orientierter Geschäftsmodelle. In T. Böhm, M. Warg, & P. Weiß (Eds.), *Service-orientierte Geschäftsmodelle* (pp. 31–52). Berlin: Springer Berlin Heidelberg.
- Zolnowski A, Christiansen T, Gudat J (2016) Business Model Transformation Patterns of Data-Driven Innovations. In: Proceedings of the 24<sup>th</sup> European Conference on Information Systems.