Enterprise Modelling and Information Systems Architectures
Vol. 13, No. 17 (2018). DOI:10.18417/emisa.13.17

Special Issue on Smart Service Engineering

Modelling of a Smart Service for Consumables Replenishment

A Life Cycle Perspective

Jürgen Anke[*,a], Stefan Wellsandt[b], Klaus-Dieter Thoben[b]

[a] Hochschule für Telekommunikation Leipzig, Gustav-Freytag-Straße 43-45, 04277 Leipzig, Germany
[b] University of Bremen, Faculty of Production Engineering, Badgasteiner Straße 1, 28359 Bremen, Germany

Abstract. *Smart services are an approach for the IT-supported provision of services based on networked products. They enable new relationships between manufacturers and end users, as well as the establishment of new value-creation networks. To gain benefits from these potentials, service providers face the challenges of designing and managing smart services. This is mainly due to the complexity of the underlying cyber-physical system (CPS) as well as the individual life cycles of components and third-party services it consists of. Additionally, a number of actors and their tasks, various tangible and intangible benefits, as well as flows of material, information and money need to be considered during the planning and provisioning of the service. In this paper, we investigate the potential of modelling smart services with the Lifecycle Modeling Language (LML). To this end, we analyse the fulfilment of information need of different stakeholders based on a consumable material replenishment service for 3D printers.*

Keywords. Life Cycle Modelling • Smart Services • Internet of Things • LML • Supply Chain Management

Communicated by M. Fellmann. Received 2017-03-10. Accepted after 2 revisions on 2018-06-10.

1 Introduction

1.1 Motivation

As part of the ongoing development of the "Internet of Things" (IoT), physical products get enhanced with embedded systems and communication capabilities to turn them into intelligent and networked products. Such products provide globally usable digital functions in addition to their local physical functions (Fleisch et al. 2015). Services provided based on the data from connected products are called "Smart Services" in this paper. For example, networked bicycles provide smart services to track training data, warn at chain wear-out and to request assistance when sensors indicate a crash (Shaw 2014). Industrial products such as compressors, ventilators and elevators get upgraded with smart services for remote control, monitoring, usage-based billing and other services (Herterich et al. 2015).

The transformation of product-based value offerings towards service-based ones is called "Servitization" (Neely 2008). It enables entirely new relationships and interactions between manufacturers, operators and users of physical goods and thus provides the basis for new data-driven business models (Velamuri et al. 2013; Zolnowski et al. 2016). Especially manufacturers of technical products and devices can reshape the nature and quality of their customer relationship by offering innovative services and thus differentiate from the competition (Fischer et al. 2012; Herterich et al. 2016). Additionally, such services can be provided by third parties, which become part of new value-creation structures as service providers and intermediaries in platform ecosystems (Mikusz 2015).

---

[*] Corresponding author.
E-mail. anke@hft-leipzig.de
Parts of this research was funded by the EU within the MANUTELLIGENCE project (grant No. 636951).
Note: This work is based on Wellsandt et al. (2016).

To design smart services, providers have to combine the suitable functionality to create value for the respective customer groups and plan the steps for service provisioning, which may involve partners and third-party services. At the same time, the financial impact of the new service has to be estimated based on the necessary resources and the expected sales volume. Particularly during the transfer of demonstrators and prototypes into marketable offers, a number of steps have to be taken to provide a quality level demanded by customers. For that, the entire life cycle of development, manufacturing, usage, maintenance and disposal must be considered and flows of information, material and money taken into account.

The resulting complexity is on one hand due to the interdisciplinary design process, in which different departments of the company have to collaborate. On the other hand, smart services are complex systems, which consist of many technical and organisational subsystems that need to work together in order to provide the service. Each of the subsystems and the smart service as a whole have different life cycles (Aurich et al. 2007). As the design of complex systems can be supported with appropriate models (Becker et al. 2010), we address the following research question in this paper: *Which benefits can stakeholders gain in the development of smart services through a modelling of the life cycle with LML?*

Our main thesis is, that modelling life cycles of smart services with LML increases transparency about the processes of service delivery as well as the dependencies of components with each other. Therefore, an LML model of smart services can facilitate both the provision of information needs for various stakeholders and the early identification of risks.

1.2 Methodology

Research Goal. The goal of this article is to demonstrate and discuss the benefits of making smart services transparent by modelling their life cycles. This enables developers of smart services to better understand the effects associated with the planned service. The underlying assumption is that a better comprehension of relationships between various product-, software-, and process elements in a smart service improves the handling of complexity, e. g. identification of risks.

As result of our research, we expect insights regarding the potential benefits that a smart service life cycle model can provide for different stakeholders. Identified shortcomings may at the same time provide a basis for an extension of the modelling language.

This work focusses exclusively on the evaluation of a model of processes and structures for the provision of smart services for networked products. The modelling procedure as such is explicitly not considered and therefore out of scope.

Research Approach. We address the research question based on a case study in five steps:

1. At first, smart services for networked products are characterised as product service systems provided by a cyber-physical system. For that, we identify the main system elements and the relevance of their life cycles.
2. From these two aspects we derive information needs for different stakeholders that participate in the design and operation of smart services.
3. Afterwards, we introduce a scenario for a smart service, which provides automated replenishment of consumables for 3D printers. This scenario is modelled using LML.
4. Finally, the model is evaluated based on the objectives which were elaborated in the second step. Furthermore, we evaluate benefits of the model based on the fulfilment of information needs of different stakeholders.
5. From these results we draw conclusions about the suitability of life cycle models in the design process for our chosen case study and further research needs.

This paper is structured according to this approach.

2 Life Cycles of Smart Services

2.1 Characteristics of Smart Services

"Smart Services" denote the needs-based provision of a combination of Internet-based and physical

services (Kagermann et al. 2015). Smart services can also be considered as *product service systems* (PSS) as they are "an integrated offering of tangible products, intangible services and enabling infrastructure" (Tietze et al. 2013). PSS are means for transforming product-oriented offerings to service- and results-oriented offerings, especially for technical products (Adrodegari et al. 2015).

Networked products are also referred to as "intelligent products" (Meyer et al. 2009) or "smart objects" (Vasseur and Dunkels 2010). The terms may vary in detail, but they all share the idea that physical products are equipped with digital communication capabilities as well as IT-based functionality to acquire or even influence the state of a product and its environment. This allows products to get integrated more efficiently in the usage-context of the customer (Kees et al. 2015). The networking of the products also enables the easy integration of the product as an external factor and thus creates the basis for data-driven service offerings (Porter and Heppelmann 2014).

The provision of smart services is typically based on the following principle: A networked product records its state using sensors and supplies the data using machine-to-machine communication (M2M) over the Internet. Some devices also provide actuators and respective control operations. The communication between product and central server or cloud service is conducted via the Internet (Wortmann and Flüchter 2015). In the cloud, the product state data serves as basis for operational and analytical functions. A smart service combines different functions on the cloud, augmented with data and functionality from other Internet services. Customers can consume the service via mobile apps or Web applications.

Smart services are thus socio-technical systems, consisting of sensors, actuators, embedded systems, digital networks, Internet services as well as coordination and management processes. Systems with this set of elements are *called cyber-physical systems* (CPS) (Broy et al. 2012). Existing work on CPS deals mainly with technical aspects. However, there is also the view of CPS as a basis for service provision. This view focusses on a combination of products and services as well as their coordination via of software to create value for customers (Mikusz 2014).

In this paper, we define *smart services* as PSS, which provide services using data from technical products based on a CPS (cf. Mikusz 2015). A prerequisite for the analysis and modelling of life cycles in smart services is the understanding of its system components. To this end, we have concretised the general CPS elements for the case of a CPS facilitating the provision of smart services (see Tab. 1). These CPS elements will get mapped to LML elements later on.

2.2 Fundamentals of Life Cycles

Terms and Concepts. A life cycle denotes different, successive periods of time that mark the way of a product or a service. From this flow-based perspective (Umeda et al. 2012), the life cycle can be divided into phases. Kiritsis suggests the separation into beginning-of-life (BOL), middle-of-life (MOL) and end-of-life (EOL) phases (Kiritsis 2011). The three phases can be further subdivided into processes. For example, material-, energy-, and information flows throughout the life cycle can be mapped to the life cycle phases (Umeda et al. 2012). An example of a product life cycle model is shown in Fig. 1.

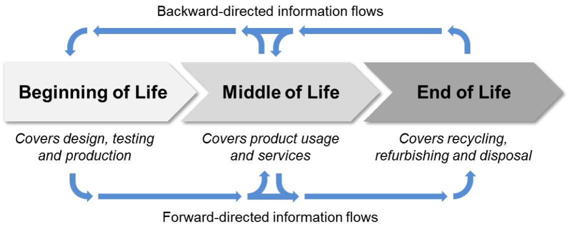


Fig. 1: Example of a product life cycle model (Wellsandt et al. 2015)

An even more specific perspective on the life cycle is the consideration of individual products (Hans et al. 2010) rather than classes of products. This so-called "item-level PLM" focusses on product information, which belong to a single, identifiable product or product component. This

General CPS item	Manifestation of the CPS element for a smart service
Physical process	Local, physical function of the product
Sensors and actuators for capturing and controlling physical processes	Various forms depending on the specific product, such as sensors for fill level, temperature, pressure and actuators for switching and control
Embedded systems	Embedded system for monitoring and control of the physical product with embedded software and communication module for data transmission
Digital networks	Various technologies for the connection of embedded system and operator cloud platform, e. g. mobile communications, Wi-Fi, or corporate networks
Utilization of globally available data and services	(cloud-based) operator software platform and any additional Internet services, e. g. electronic marketplaces
Multi-modal human machine interfaces	Various forms of user interaction on the product itself, via mobile apps or Web applications
Management processes	Management of the service (booking, configuration, billing)
Coordination processes	Provision of the service, control of service provision
Logistics processes	Delivery of physical (e. g. spare parts, consumables) or digital components (e. g. software updates)

Tab. 1: Manifestation of the CPS elements for a smart service

information, for example, is relevant for the individualisation and servicing of products (Corcelle et al. 2007).

Life cycles and smart services. The provider of a smart service is typically responsible for the entire life cycle of the underlying service system. The challenge in designing such a system is that individual components of the system are developed separately but have to work together in order to provide value to the customer. Each service system element can have an individual life cycle. For example, different versions of a component can exist over time (e. g. software platform). Changes in the life cycle of individual components may arise through new customer requirements, legal changes or availability of new technologies (Wolfenstetter et al. 2015).

In contrast to the product life cycle shown in Fig. 1, several life cycles need to be considered simultaneously. These life cycles run in parallel or side by side with a temporal offset. Furthermore, there are relations between the stages of life cycles. The following list shows examples of effects arising from changes in the life cycle of a smart service system element:

- *Customer or product reports problem:* Identification of the associated elements, classification into potential physical or digital problem fixes. Remote control, configuration or update procedures or dispatching of technicians have to be performed for affected components.
- *Defect of the product:* Replacement of the physical product requires logistics for the delivery, return of the defective product, as well as updating the service configuration to make it work on the new product.
- *Defect of embedded system:* Replacement of the affected hardware, update of the embedded software, transfer of service configuration, e. g. to preserve the product's digital identity.
- *Changes in the service management processes:* Enhancement of the operator cloud-platform and service configuration for customers, update of the embedded software, e. g. to facilitate additional pricing models, such as prepaid or subscriptions.

These examples show that modifying one element can affect others, which may lead to disruptions in the service provisioning. Being aware of these dependencies is of great importance for ensuring a high availability of the system. As we will show in the analysis of information needs for different roles, this holds true for activities in all life cycle phases. It also helps to prevent unwanted side-effects and supports collaboration in the ongoing operation and optimisation of the service system.

The types of relationships between life cycles or parts of a life cycle are currently being discussed. Westphal et al. (2015) and Wiesner et al. (2015) explain the importance of interactions between product and service life cycle management for manufacturing companies. Kwak and Kim (2013) present an approach for life cycle costing and life cycle assessment to analyse the economic and ecological impact of PSS. Lindström et al. (2014) show interactions of life cycles in “Functional Products”. This type of PSS consists of hardware, software, management operations and a service support system. Further challenges in dealing with multiple life cycles in the product and service development are investigated in current research projects, e. g. the Manutelligence Consortium (2016), the Falcon Consortium (2016) and the Psymbiosys Consortium (2016).

Classes and instances of service elements. While the aforementioned life cycle considerations refer to the management of smart service component types, the advances in information technology provide increasing possibilities to trace individual components. These components will be referred to as “instances” in this paper – the same notion is already used for physical objects (Hans et al. 2010). Each instance of a smart service component has information about its own life cycle, e. g. how and where it was produced, used and disposed. Taking an instance-level perspective on smart services increases the complexity of the system under investigation. Herein, complexity refers to the number and diversity of components, as well as their relations among each other (cf. Funke 2012, p. 683). Instead of managing types of products, software and services, a plethora of instances with own life cycle information must be handled by the smart service provider. This increase in complexity challenges the existing life cycle management approaches. On the other hand, access to instance-level information can be a valuable basis for additional smart services, e. g. predictive maintenance. Further, it enables smart service designers to improve the service system (Lützenberger et al. 2016).

2.3 Modelling of Life Cycles

From the previous sections, it can be seen that a smart service as CPS-based PSS is characterised by high complexity due to two properties:

1. *Diversity* of system elements (e. g. mechanical and electronic components, software, processes for service provisioning and management) as well as the number of relevant stakeholders.
2. *Interdependencies* caused by relationships among system elements as well as the necessary consideration of information flows, material flows, and money flows between them (cf. Sect. 2.2).

These two aspects should be addressed by life cycle modelling, to make the model useful for smart services design. This is substantiated by following objectives:

- *Illustrate the complexity and thus improve its manageability.* Complexity is characterised by the number and by the variety of system elements and by the number of relationships between them. The model should be able to represent the required elements and relationships.
- *Enable identification of risks through dependencies between system elements.* Effects caused by modifying one component on other system elements must be identifiable, e. g. the exchange or upgrade of hardware or software.

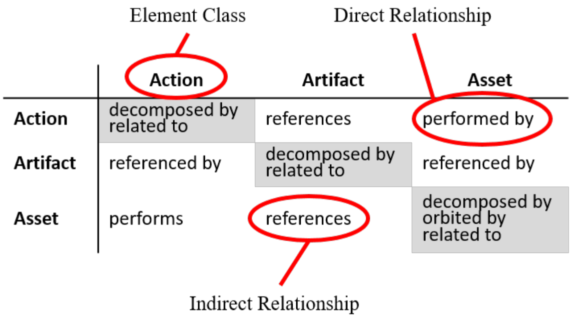
A comparison of modelling approaches for services from various disciplines showed that especially the modelling of PSS, as well as the representation of life cycles were only weakly supported (Hoffmann et al. 2009). Some basic model types that can be used in the context of service modelling, are described by Scheer et al. (2004). More specific approaches used to describe processes are for example proposed by Gronau et al. (2010), Klingner and Becker (2012), and Meis et al. (2010).

Widely acknowledged modelling languages are Business Process Model and Notation (BPMN), Unified Modeling Language (UML) and Systems Modelling Language (SysML). A comparably novel language is the Lifecycle Modelling Language (LML). A committee of systems engineering practitioners and academics introduced it in 2015. LML bases on SysML and the Department of Defense Architecture Framework, and it focuses on life cycle modelling in particular.

Very few academic papers mentioned LML since its release in 2015 (e. g. Hefnawy et al. 2016). Therefore, we had little complementary literature to evaluate it. All four modelling languages are potential tools to support life cycle modelling. An important characteristic of a smart service is the component diversity. Actors with different backgrounds are involved to plan these components and their orchestration. Examples are marketing experts, mechanical and electrical engineers, programmers, sales people, lawyers and customers. For this reason, the ease of use is an important criterion to evaluate the suitability of modelling languages. It means that a language is, for instance, comprehensible and easy to learn. Tab. 2 provides an overview about the focus, type, page count, and the ease of use of the identified languages. We derived the focus and type values from the languages' specification documents (see references). The page count refers to the pages of the specification document in PDF format – we do not differentiate preface, content and annex. We consider it as a rough indicator for the time an actor needs to learn a language. Our estimation of the ease of use grounds on the page count and the scope of a language. In this context, more pages lead to a lower ease of use.

BPMN's focus on business processes and the extensive documentation indicates the high complexity of the language. UML follows a general-purpose modelling approach and its specification is even more extensive compared to BPMN. Therefore, we consider it as difficult to use as well. SysML is an extension of UML and a general-purpose modelling language. We identified it as easier to use compared to the other two languages. The reason is that SysML aims to support the collaboration between systems engineers and software engineers. This objective reflects the involvement of heterogeneous actors in a development task. LML is a general-purpose modelling language meant for systems engineering with a dedicated focus on the product life cycle. The specification has 70 pages, which indicates a high ease of use. The preceding assessment identifies LML as the most promising candidate for a language in life cycle modelling.

One of the distinct features of LML is an ontology that stores the specified entities and their relationships. The ontology includes twelve basic entity types. Each of these entities is related to all other entities in specific ways. LML supports inheritance, for instance, the entity type "resource" is derived from the superordinate type "asset". The LML ontology is supported by different visualization methods to represent the behaviour as well as the structure of systems. Examples include activity diagrams (how the system behaves) and hierarchy diagrams (how system elements relate to each other). Fig. 2 illustrates an excerpt of the entities and their relationships.



	Action	Artifact	Asset
Action	decomposed by related to	references	performed by
Artifact	referenced by	decomposed by related to	referenced by
Asset	performs	references	decomposed by orbited by related to

Fig. 2: Selected element classes and relationships

Characteristic	BPMN 2.0	UML 2.5	SysML 1.4	LML 1.1
Focus	Business processes	Software-based systems & business processes	Systems engineering problems	Systems engineering problems along entire life cycle
Type	Process modelling	General-purpose modelling		
Spec page count	538 pages	794 pages	346 pages	70 pages
Reference	OMG (2011)	OMG (2015b)	OMG (2015a)	LML Steering Committee (2015)
Ease of use	Low	Low	Medium	High

Tab. 2: Application difficulty of systems modelling languages in life cycle modelling

The applicability of LML in the context of smart service life cycle modelling is illustrated in Tab. 3. On the left column of the table the aforementioned elements of a smart service are listed. The middle column contains examples of LML entities that can be used to represent the elements. For each entity, there are examples provided in the right column.

3 Case Study

3.1 Replenishment for 3D Printers

In this section, we present an example for a consumable replenishment service. Consumable replenishment is a commonly offered smart service. Tab. 4 summarises several examples of these services, which are currently offered.

One goal of a consumable replenishment service is to ensure that customers have access to a previously agreed service level. The service operator charges the customer based on this service level agreement. An example for a service level is the amount of printable pages in an office printer. In the case of intelligent products, embedded measurement devices monitor the service level. An office printer, for instance, monitors the remaining amount of ink/toner to calculate the amount of printable pages. Once the ink/toner supply reaches a threshold value, the printer orders additional consumables.

In this paper, we focus on a consumable replenishment service for 3D printers. We created the concept and the life cycle model for this service during the EU-funded research project Manutelligence. A producer of additive manufacturing machines collaborated with us in this process. For them, the service concept was an opportunity to estimate the complexity of a consumable replenishment service. With this information, we supported them in their decision regarding the realization of the service. Thus, the life cycle model represents a real case for a consumable replenishment service. The following descriptions and models refer to main components of the envisioned smart service for 3D printers, which serves as a representative for similar cases.

3D Printers. Additive manufacturing processes have been in use for many years for building prototypes. In consumer 3D printing, the “fused filament fabrication” (FFF) technology is mainly applied, for example in the open source 3D printer “RepRap” (RepRap Project 2014). The basis of FFF is a heated printing head that melts solid materials, like thermoplastics. The print head can be moved in all three spatial directions with the help of a moving frame structure. Once applied, the liquid material hardens and forms the body to be manufactured layer by layer. In the following, we use the term “3D printer” as a synonym for FFF-based printers.

Consumables Replenishment. Similar to conventional paper printers, a 3D printer requires a steady supply of printing material. It is typically provided in the form of a plastic wire coiled on a spool. Depending on the printer type, one or more

Tab. 3: Mapping of CPS elements to LML expressions.

General CPS Items	LML Entities	Examples
Physical process	Activity	Represents the actual process.
	Input/Output	A consumable that is used up.
	Asset	Hardware performing the process.
Sensors and actuators for capturing and interacting physical processes	Activity	Measurement or actuation.
	Input/Output	Sensor data.
	Asset	Sensors and actuators are product parts.
Embedded systems	Asset	The hardware and software.
Digital networks	Conduit	Internet, Wi-Fi, 5G network.
	Input/Output	Data shared via network.
	Asset	Software systems sharing data.
Use of globally available data and services	Activity	Service integration.
	Input/Output	Data.
	Asset	Services, service market.
Multi-modal human machine interfaces	Conduit	Human-machine interface.
	Input/Output	Data shared via the interface.
Management processes	Activity	Represents the actual process.
	Input/Output	Shared data/information.
	Asset	Business unit, role, IT-system.
Coordination processes	Activity	Represents the actual process.
	Input/Output	Shared data/information.
	Asset	Business unit, role.
Logistics processes	Activity	Represents the actual process.
	Input/Output	Parcel that is transported.
	Asset	Truck, logistics provider, customer.

spools with filament can be stored in the printer, e. g. different colours or water soluble material for support structures. A service which is relevant to owners of personal 3D printers as well as for professional service providers, is the automated deployment of filament. The provision of the service does not only involve the customer and the service provider (operator) but also other companies, e. g. the logistics provider and disposer of empty spools. The service falls in the category of condition monitoring services, which require existence of distinct product instances, relevant product state properties and target levels for these properties (Knoke and Thoben 2014). The basis for the performance of the service is a contract between the customer and provider, which defines the technical requirements, performance levels, terms and conditions. As first step, the 3D printer has to be connected to the ordering system of the operator and registered as a new instance of the product. Afterwards, the state "remaining printing material" from the printer can be queried and delivered for further monitoring to the operator's platform.

The actual service provision is conducted according to the proposal of Knoke and Thoben: At the customer site, a purchase order is automatically triggered by the 3D printer when the state of "remaining printing material" falls below a threshold value of e. g. 10%. Depending on the configuration, an approval of the order is required by the customer. At order receipt, the operator

Case	Scenario	Consumable	Source
Winterhalter	Pay-per-wash offering for industrial dishwashers	Detergents	(Winterhalter 2016)
Canon	Pay-per-page offering for industrial printers	Ink, Paper	(Canon Europe 2017)
HP	Pay-per-page offering for consumer ink cartridges	Ink	(HP Inc. 2016)

Tab. 4: Existing Cases of Smart Services for Consumables Replenishment

has to fulfil it by supplying the consumables to the customer. The provider can delegate this task to a wholesaler for plastics or to a plastic manufacturer. The purchased materials have to be delivered to the customer via a logistics service provider. The provision of consumables can additionally be bound to the obligation to take back the emptied material spools to facilitate recycling. Depending on the contractual agreement, collection points or a return shipping via a logistics service provider can be arranged for this. The billing in turn depends on the pricing model. In the example described, both individual order and subscriptions are possible. Both variants are offered with comparable services such as “Total Service Care” (Canon Europe 2017) and “Instant Ink” (HP Inc. 2016).

3.2 Information Needs of Stakeholders

Designing smart services is a interdisciplinary project. The service provider must involve different departments in order to get a comprehensive understanding of the requirements for a new service (cf. Laurischkat 2013). With regard to our research question, we need to evaluate the benefits of a smart service life cycle model for the involved stakeholders. In the following, we denote the different stakeholders as roles. To derive the information needs of each role, we first identify some of their main tasks and goals (cf. Junginger et al. 2006):

- *Marketing:* Increase customer loyalty, sell service contracts, understand customer needs
- *Development:* Design and implement smart service system technically
- *Finance:* Minimise required investment budget, ensure profitability of the service
- *Procurement:* Procurement of consumables and spare parts
- *Logistics:* Picking, packing and transporting of consumables and spare parts
- *Customer Support:* Solving customers’ problems, improving service availability

From the tasks and goals, we have identified information needs in the different life cycle phases, which are summarised in Tab. 5. For a model to fulfil these information needs, we have derived a more general set of objectives as follows:

- *O1: Support the conception of product-related services.* Modelling should support planning of smart services, as well as the preparation of further analyses, e. g. life cycle assessment, or life cycle costing.
- *O2: Allow an assessment of the service concept.* The model must be easy to comprehend, e. g. through clarity and unique labels for elements and relationships.
- *O3: Support the planning of required capacity for resources.* Required resources and capacities such as person hours can be depicted in the model.

3.3 LML Model of the Smart Service

Modelling Approach. The aim of this model is to describe the elements and their relations with respect to the features of the chosen smart service example. The formally correct modelling is not the primary goal. Instead, we focus on the

Tab. 5: Examples of information needs of different roles by life cycle phase

Role	BOL needs	MOL needs	EOL needs
Marketing	customer needs, prices	customer satisfaction, customer number	next generation products, recycling demands
Development	system requirements, solution approaches	identified problems and bugs	technical migration paths to next version
Finance	planned revenues and development cost	operating cost and actual revenues	cost for warranties and recycling
Procurement	type of items for procurement, planned lead times, potential suppliers	quantities and times for the provision of intermediate consumption	(no information need identified)
Logistics	required stock space, lead time, package sizes, quantities	Items to be delivered quantities and dates	removal of old equipment from customer sites and / or recycling
Customer Support	contact channels, availability, languages, response times	current incidents / tickets	(no information need identified)

assessment of the life cycle modelling based on the objectives and information needs listed in Sect. 3.2. The modelling of the example service is done by the derivation of specific items from the LML ontology element classes. The corresponding relationships are automatically created according to the LML specification by the modelling tool Innoslate (SPEC Innovations 2016), which we have used for our research. Activity diagrams and graphs are used to visualise the elements. The term “Graph” is used instead of “Spider Diagram” specified in LML. Spider diagrams typically have a different structure and meaning than the chart type in the LML specification.

Step 1 - Determine Stakeholders and their Relationships. As the first step, relevant stakeholders have been identified (similar to the perspectives in a service blueprint). This way, the model is given an initial frame, which can be detailed further. The number of stakeholders to be included depends on the required level of detail in the modelling. It is not our intention to show how a service can be described as comprehensively as possible. Therefore, we have only taken a small number of stakeholders taken into account. A selection of stakeholders allows a first consideration of the life cycle, i. e. stakeholder from different phases of the life cycle should be considered (but ultimately not all have to be modelled). In addition to stakeholders, first relationships were also taken into account. Relationships denote information and material exchange between stakeholders. An overview of the stakeholder and their relationships of the example service is given in Fig. 3.

Besides the customer, who is at the centre of the service, three companies are identified as external partners. The service provider communicates directly with the customer and determines current needs for consumables. The provider passes the request on to a wholesaler for plastics, which in turn informs a logistics provider. The logistics provider receives the required material from the wholesaler and delivers it to the customer. Emptied spools are returned by the customer to the logistics service provider, who transports it to the wholesaler or disposer. Returning emptied spools is not part of the model.

Step 2 - Modelling Structure and Processes. Starting from the stakeholder network shown in Fig. 3, some parts of the service were modelled

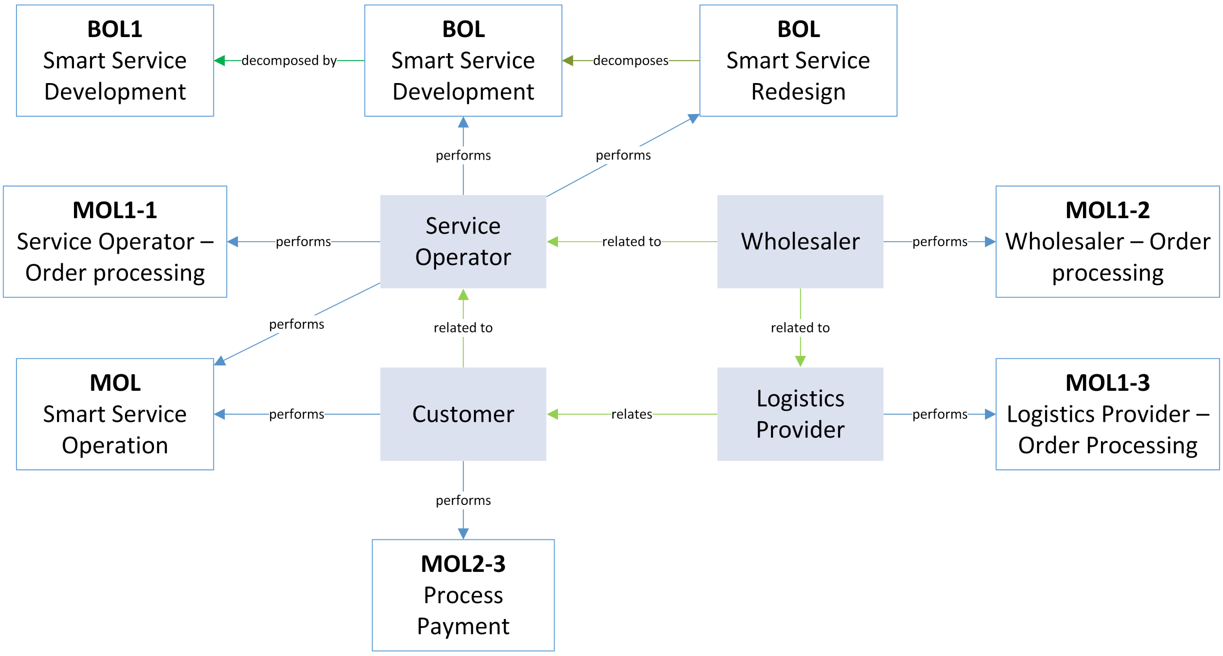


Fig. 3: Overview of stakeholder and important relationships

in more detail. Here, we differentiate static and dynamic elements:

Static elements comprise physical, software-based and abstract elements, which describe the static structure of a smart service system. In addition to physical and software-based components, also abstract elements are defined. They are used to realise different levels of detail in the form of model layers. An overview of the essential elements of the modelled smart service is provided in Tab. 6.

Dynamic elements relate to life cycle stages, processes and activities. The life cycle represents the top level of the smart service. It consists of the three phases of BOL, MOL, and EOL, as depicted in Fig. 4. The development phase of the smart service generates a 3D printer which can determine and communicate information on the material consumption. Furthermore, a service platform is developed, to facilitate parts of the service provisioning, e. g. billing. Service data is returned from the operating phase into the development phase, where it is used for further improvement of the service.

The BOL and MOL phases of the top-level model are further detailed into processes and inputs/outputs, as indicated by “decomposed”. Two model layers with details of the MOL phase are shown in Fig. 5. The naming of child activities includes the life cycle name and a sequential number (top left of each box).

Graphs in LML represent the relationships between the modelled elements. The relationships are specified by the LML ontology. If, for example, an element of the “Action” class is connected to an element of the “Input/Output” class, the relationship is “generated” or its inverse “generated by”. An example of a simple graph is shown in Fig. 6. A disadvantage of the visualization via graphs is the growing complexity if more than just the immediate neighbours of an item have to be displayed. Fig. 7 shows the same example for the input/output “Payment” with neighbouring elements of the second degree. The complete graph on the model represents all modelling levels. To make work with LML graphs meaningful, relevant parts must be filtered out.

Tab. 6: Core components of the smart service model

Description	Form	Note
3D printer	Abstract	Combines hardware and software
Hardware	Physical	Structure, mechanics and electronics
Software	Software	Data processing
Printing material	Abstract	Combines spool and plastic wire
Coil	Physical	The plastic wire carrier
Plastic wire	Physical	Consumables
Operator platform	Software	Software of the service operator

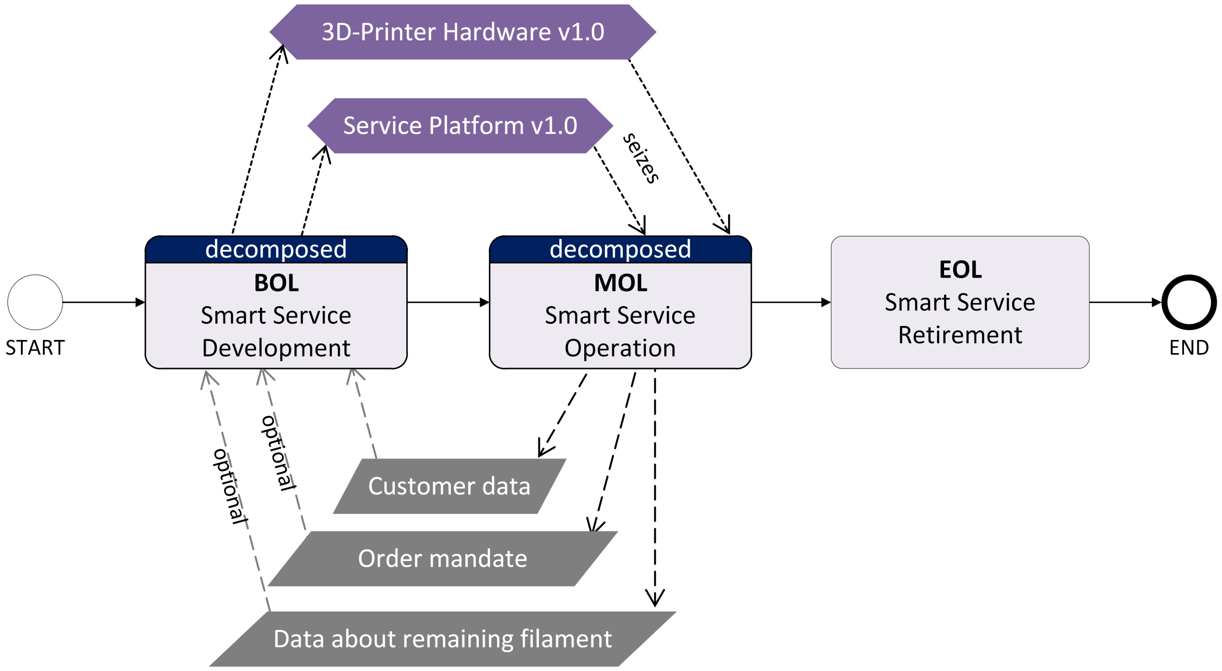


Fig. 4: Representation of the life cycle as a top-level model

Relationships across life cycle phases are depicted only for a case in our model: the material fill level of the printer is created in the operation phase of the service and sent back in the development phase, together with the customer data, and performed orders for consumables. With the obtained information the service can be improved in general or specifically for a certain customer.

An example of a possible data-based improvement is the use of the material fill level to optimise the filament stock capacity in the printer. This could be done, for example, on the basis of a parametrised CAD model of the printer. A design approach for this purpose is provided by (Klein et al. 2015). Another example is the adaptation of the smart service with regard to logistics: Additional material providers can be chosen based on customer data and order quantities. Therewith, delivery times of consumables can be reduced accordingly.

The way this improvement is depicted in the model provides the benefit that the role of the logistics provider (and optimization of related processes) is already considered in the design of the smart service. Therefore, a meaning and a value can be assigned to data from the printer in

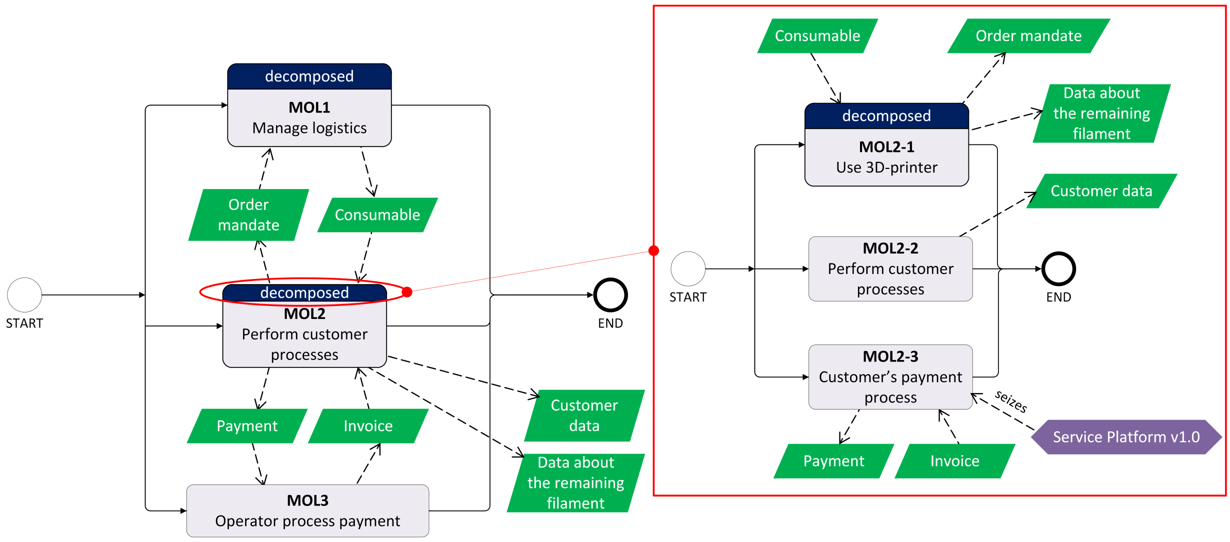


Fig. 5: Examples of MOL activities on different layers of the model

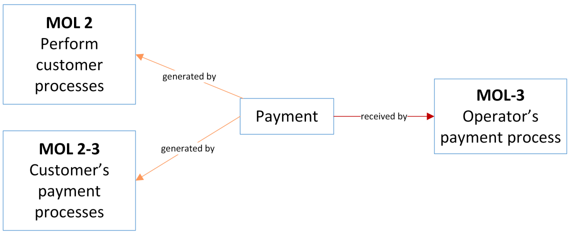


Fig. 6: Example of a graph for the input/output "Payment"

the development phase. Fig. 8 shows how the use of data for service improvements was modelled.

The complexity of the model and other characteristics are summarised in Tab. 7.

4 Discussion of the Modelling Approach

4.1 Review of Information Needs

In this section, the proposed modelling approach for the life cycles of smart services is discussed and evaluated. The hypothesis stated in the introduction will be verified. Its main assumption is that life cycle modelling makes relations among service elements transparent and thus improves the identification of risks and fulfils various information needs. For the evaluation, the information needs of different roles (see Tab. 5) are compared with the capabilities of the life cycle modelling approach.

Marketing. In the BOL phase a key concern is the identification and documentation of customer needs. From the marketing perspective, the activity diagram is not supporting this process significantly, as requirements cannot be modelled with it. However, the graph diagram may contain "requirements" entities that are connected with other service components (e. g. activities). This way a traceability can be realised to support the identification of problems introduced by changing requirements or functions. Specific risks, such as missing or erroneous requirements, may not be identified through the graph. During the MOL phase, the feedback about customer satisfaction is a key indicator to measure success of the smart service. For this purpose, the modelling approach must consider dynamic data coming from the market (e. g. a survey). Currently, the modelling approach is not capable of satisfying this information need. In the EOL phase information is needed, for instance, to understand the recycling demands of physical components. The modelling approach supports the description of activities and input/outputs related to the EOL. Their relation with other activities (e. g. redesign and legal

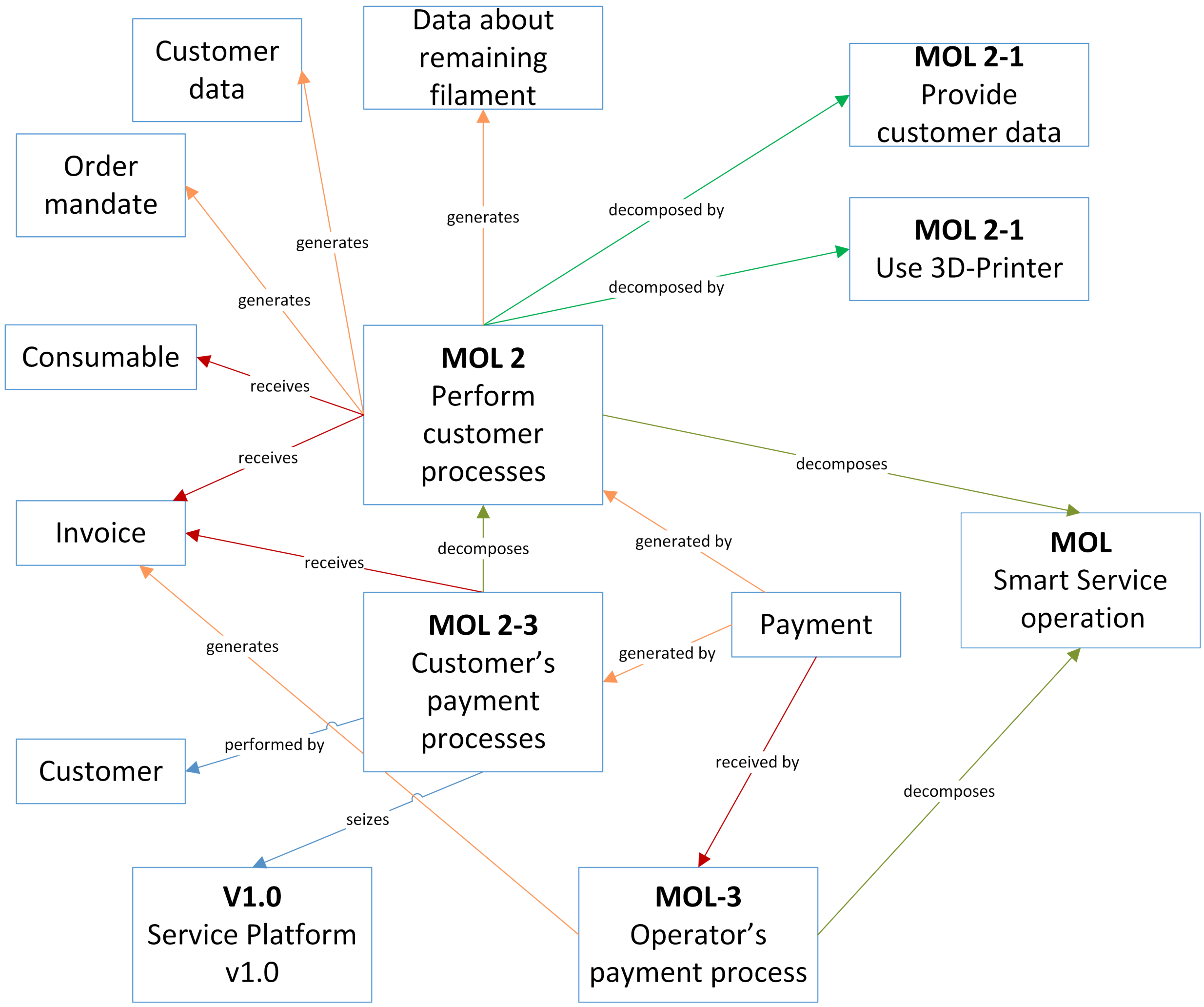


Fig. 7: Extended example of the graph for the input/output "Payment"

activities) may indicate risks, such as missing activities to manage legally mandatory take-back of electronics.

Development. During the BOL, the smart service hard- and software is designed. Activity diagrams can help to visualise the functionality and related technical requirements. In addition, data, material, energy and monetary flows can be made transparent by establishing them as inputs/outputs between activities. "Design for X" approaches, where "X" concerns, for instance logistics, maintenance, recycling and reliability, can be supported by incorporating specific activities, assets or characteristics into the life cycle model.

The MOL phase is interesting for designers, since the smart service components may be subject to failures and insufficiencies. Since the life cycle modelling approach is not supporting dynamic data (e. g. field data), feedback from the MOL phase is not represented in the life cycle model. For the EOL phase, migration paths are an information need. The developers need to plan, for instance, how non-supported service components (e. g. hardware and software) are exchanged. Possible migration paths can be described with activities and by defining related requirement entities.

Tab. 7: Complexity of the LML model for consumables replenishment

Characteristic	Manifestation in example LML model
Number of model elements	57 total: 31 actions, 7 assets, 19 inputs/outputs
Number of abstraction levels	Maximum of 5 levels, e. g. smart service operation > customer processes > printer > hardware-related functions > print object)
Design of activity diagrams	A maximum of 3 processes were used in activity diagrams, otherwise a new abstraction layer was created.
Used types of flows between processes	electronic and analogue information (e. g., sensor data, delivery note), material (e. g. consumables, empty spools), money (payment)

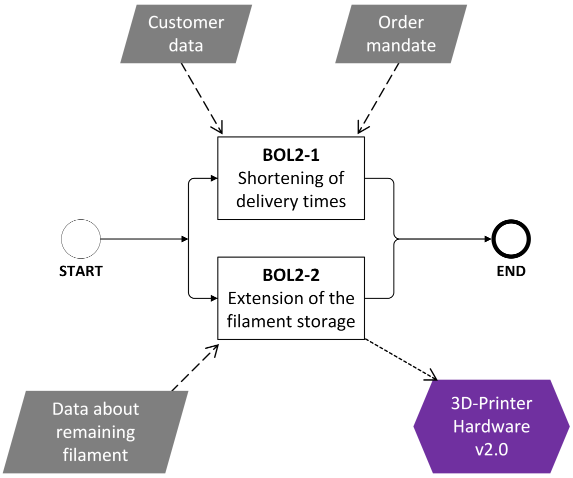


Fig. 8: Use usage information for the further development

Finance. During the BOL phase, the estimation of financial revenues and development cost can be supported by describing value streams in activity diagrams with input/output entities. The activities can be related to assets representing different value chain partners (e. g. suppliers). In addition, cost entities can be assigned to almost any other entity to clarify, for instance, the amount, currency and frequency of payments. The MOL phase and the EOL phase cannot be sufficiently supported with the current modelling approach, since dynamic data from business transactions (value streams) are not supported.

Procurement. In the BOL phase, LML may support the planning of the required types of items from different suppliers. The decomposition of assets allows to model the service components and assign them with a "purchase" activity that can be performed by different suppliers (assets). Depending on the complexity of the supply chain, the model might become quite extensive and thus difficult to manage (e. g. update in case a supplier changes). The MOL phase is not supported well, due to the lack of real time data integration.

Logistics. During the BOL phase, specific values for the types and sizes of consumables must be identified. The different types of consumables, as well as their packages, can be defined as separate assets that are related to wholesalers. The size of the package or the products can be added with measure entities for length, width and height. In addition, the supply chain stakeholders and their activities can be modelled. The MOL and EOL phases, once more require dynamic data which is not well supported in LML.

Customer Service. Important BOL-specific information, such as contact channels, languages and the definition of response time, can be described in a life cycle model with different entities. While channels can be modelled as resources, the language and response time can be defined with characteristic or measure entities. Dynamic MOL data, such as current incidents, are not well supported by LML.

The summary of evaluation results regarding information needs is shown in Tab. 8.

4.2 Review of General Objectives

The modelling approach is further evaluated according to the objectives defined in Sect. 3.2.

Tab. 8: Fulfilment of information needs of different roles by life cycle phase

Role	BOL needs	MOL needs	EOL needs
Marketing	No	No	Yes
Development	Yes	No	Yes
Finance	Yes	No	No
Procurement	Yes	No	No
Logistics	Yes	No	-
Customer Support	Yes	No	-

- *O1: Support the conception of product-related services.* The uniform description and expressions of LML, as well as the similarities with the widely-used SysML standard, provide stakeholders involved in the design an easy access to the creation and adaptation of the model. Further analyses during the conception phase may benefit from the uniformly described model elements. An example is the performance of a life cycle cost calculation grounded on the information stored in the life cycle model (this possibility is currently researched in the Manutelligence project). The specification of LML (version 1.1) provides 12 entities and their relationships among each other. In case that these original entities are not sufficient to describe the concept of an application case, new entities can be created by inheritance (i. e. the new entity inherits characteristics of its superordinate entity). In a similar way, new relationships can be created between entities. [fulfilled]
- *O2: Allow an assessment of the service concept.* The assessment of a service concept can be carried out from different perspectives (e. g. financially, technically, logistically and environmentally). Depending on the perspective, different entity types must be added to the model, for instance, cost entities in case of the financial department's perspective. A key issue of the modelling approach is that dynamic characteristics of service elements are not well supported. Each life cycle model is static, i. e. it represents the system at a specific moment. Through the integration of additional software tools, the model may be updated with real time information (a research question investigated in Manutelligence). An example is the regular update of the market price of printing filament – this could be realized by updating the associated value of a cost entity. This way, the life cycle model supports the assessment of the current system status which is an information need emerging from the MOL and EOL phases. The assessment of risks, in particular, can be realised through risk entities. Whether all risks can be modelled this way was not investigated in this paper. Some risks may be relevant at certain phases of the life cycle. In the case of a complex smart service, risks may be caused by the relationship between service elements. For instance, if a service component that acts as an information source for other components is removed, the whole service may be affected in a negative way unless the risk is addressed by appropriate measures. The time-dependency of this example is related to the fact that one component is in its EOL, while the others are still in their MOL phase. [partly fulfilled]
- *O3: Support the planning of capacity for resources.* Resource planning and the identification of resource bottlenecks can be realised with resource entities. Modelling resources with LML is difficult, since they are either created, seized or consumed. In the case of 3D-printing, the consumable could be represented by a resource entity that is created by the wholesaler and consumed by the printing process. However, the logistics process neither consumes nor seizes the consumable. For this reason, the consumable was represented by an input/output entity that moves between processes. The representation resources is not intuitive using LML. [not fulfilled]

The assessment of the objectives, given the modelled example case and the modelling environment (Innoslate), is summarised in Tab. 9. For the interpretation it must be noted that only one example was modelled without a specific organizational context. It is further worth of notice that software support for LML is still rather poor (modelling environment).

Tab. 9: Assessment of general objectives

Objective	Review
O1: Support the conception of product-related services	Fulfilled
O2: Allow an assessment of the service concept	Partly fulfilled
O3: Support the planning of capacity for resources	Not fulfilled

5 Conclusion and Outlook

The evaluation of the life cycle modelling of smart services did not lead to a definite result. The initially stated hypothesis, information needs and general objectives were assessed very differently from our perspective. The following paragraphs contain conclusions on aspects of the modelling approach. At the end of each paragraph, potential research questions and, in some cases, suggestions for literature are provided.

The collaborative design of life cycle models is well supported by the applied modelling approach. LML's ontology provides different entity types and relations that reflect stakeholder perspectives. The relations are “speaking”, i. e. named to be easily understood by stakeholders – they support the modelling process even though stakeholders might not have expert knowledge in modelling. Conflicts arising from discussions among different stakeholders were not covered in this paper. Research questions concern how a collaborative design of the life cycle model happens in practical cases, which conflicts or problems occur and how these issues could be mitigated. Previous work, for instance in Computer-aided Service Design (Laurischkat 2013), is a starting point for this research.

The static nature of the model has been identified as a weakness of the modelling approach. However, it is an open question, whether a life cycle model should be designed more static or more dynamic. The integration of dynamic data, such as product states, could satisfy several information needs of the MOL and EOL phase. An open research questions is how dynamic data could be integrated into a life cycle model on the conceptual level and on the practical level (software).

The required level of detail of the life cycle model appears difficult to estimate in advance. We assume that the model itself evolves concurrently along the life cycle, i. e. it starts simple in the conceptualization phase and becomes more complex as new perspectives need to be considered. With an increasing complexity of the model, the visualization methods proposed in the LML standard become difficult to read, especially the net diagram. An option is to reduce the scope of the net visualization; however, the reduction may limit the chances to identify potential risks and opportunities because a part of the system is no longer visible to stakeholders. Research questions concern how risks and opportunities in smart services emerge and how their identification could be supported. During the research, a classification of challenges of PSS could be useful as proposed by Kurak et al. (Di Francisco Kurak et al. 2013) as well as previous work on understanding service uncertainties of PSS cost estimation (Erkoyuncu et al. 2011).

The lack of an item-level representation of service components and the limited functions to consider dynamic system characteristics (e. g. data streams) are arguments against the use of the life cycle model for the operational management of a smart service. A research question is to determine the benefits of using a common life cycle model in the operation phase of a smart service. The question is related to the existing research on Product Lifecycle Management, i. e. the management of product-related information (e. g. Demoly et al. 2013; Kiritsis 2011).

To gain more insight into life cycle modelling, additional real-world cases should be modelled and evaluated. Action-based research appear to be a suitable methodology to determine how stakeholders develop and use the model. On this basis, best practices for modelling could be elaborated to provide orientation to new users of LML regarding structuring models and useful level of model details. Finally, different modelling languages (e. g. BPMN) should be evaluated against the requirements of different roles involved in the process of life cycle modelling. A starting point can be the information requirements identified in this paper. The result of such a comparison could clarify whether one language is sufficient to describe the life cycles of smart services or not. In the latter case, the results might indicate that different modelling languages should be used in the process.

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. In: Procedia CIRP 30, pp. 245–250

Aurich J. C., Schweitzer E., Siener M., Fuchs C., Jenne F., Kirsten U. (2007) Life Cycle Management investiver PSS Gestaltung und Realisierung investiver Produkt-Service Systeme. In: wt Werkstattstechnik online 97(7/8), pp. 579–585

Becker J., Beverungen D. F., Knackstedt R. (2010) The challenge of conceptual modeling for product–service systems Status-quo and perspectives for reference models and modeling languages. In: Information Systems and e-Business Management 8 (1), pp. 33–66

Broy M., Cengarle M. V., Geisberger E. (2012) Cyber-Physical Systems: Imminent Challenges. In: Calinescu R., Garlan D. (eds.) Large-scale complex IT systems Development, operation and management, 17th Monterey workshop 2012, Oxford, UK. LNCS 7539. Springer, Heidelberg and New York, pp. 1–28

Canon Europe (2017) Total Service Care Large Format Solutions http://www.canon-europe.com/for_work/products/large_format_printers/solutions_services/totalservicecare/ Last Access: 2017-07-10

Corcelle C., Främling K., Rabe L., Anke J., Petrow J. (2007) Assessment of item-specific information management approaches in the area of heavy load vehicles. In: Garetti M., Terzi S., Ball P. D., Han S. (eds.) Proceedings of the International Conference on Product Lifecycle Management (PLM 2007). Inderscience Publishing, pp. 773–782

Demoly F., Dutartre O., Yan X.-T., Eynard B., Kiritsis D., Gomes S. (2013) Product relationships management enabler for concurrent engineering and product lifecycle management. In: Computers in Industry 64 (7), pp. 833–848

Di Francisco Kurak C., Barquet A. P. B., Rozenfeld H. (2013) Challenges for PSS Implementation: Identification and Classification. In: Meier H. (ed.) Product-service integration for sustainable solutions Proceedings of the 5th CIRP International Conference on Industrial Product-Service Systems. Springer, Berlin, pp. 275–285

Erkoyuncu J. A., Roy R., Shehab E., Cheruvu K. (2011) Understanding service uncertainties in industrial product–service system cost estimation. In: The International Journal of Advanced Manufacturing Technology 52 (9-12), pp. 1223–1238

Falcon Consortium (2016) Feedback mechanisms across the lifecycle for customer-driven optimization of innovative product-service design. http://www.falcon-h2020.eu/ Last Access: 2016-02-18

Fischer T., Gebauer H., Fleisch E. (2012) Service Business Development Strategies for Value Creation in Manufacturing Firms. Cambridge University Press, Cambridge

Fleisch E., Weinberger M., Wortmann F. (2015) Business Models and the Internet of Things. In: Podnar Žarko I., Pripužić K., Serrano M. (eds.) Interoperability and Open-Source solutions for the Internet of Things International Workshop, FP7

OpenIoT Project. LNCS 9001 Vol. 9001. Springer, Cham, pp. 6–10

Funke J. (2012) Complex Problem Solving. In: Seel N. M. (ed.) Encyclopedia of the sciences of learning. Springer reference. Springer, [Place of publication not identified], pp. 682–685

Gronau N., Bahrs J., Hake M., Heinze P., Lembcke R., Scharff C., Vladova G. (2010) Wissensorientierte Modellierung im Lebenszyklus von Dienstleistungen. In: Thomas O., Nüttgens M. (eds.) Dienstleistungsmodellierung 2010. Physica-Verlag, Heidelberg, pp. 3–23

Hans C., Hribernik K. A., Thoben K. D. (2010) Improving reverse logistics processes using item-level product life cycle management. In: International Journal of Product Lifecycle Management 4 (4), p. 338

Hefnawy A., Bouras A., Cherifi C. (2016) IoT for Smart City Services. In: Boubiche D. E., Hidoussi F., Guezouli L., Bounceur A., Cruz H. T. (eds.) Proceedings of the International Conference on Internet of Things and Cloud Computing the International Conference. ICC '16. ACM, Cambridge, United Kingdom

Herterich M., Buehnen T., Uebernickel F., Brenner W. (2016) A Taxonomy of Industrial Service Systems Enabled by Digital Product Innovation. In: Proceedings of the 49th Hawaii International Conference on System Sciences (HICSS 2016). Koloa, HI, USA, pp. 1236–1245

Herterich M., Uebernickel F., Brenner W. (2015) The Impact of Cyber-physical Systems on Industrial Services in Manufacturing. In: Procedia CIRP 30, pp. 323–328

Hoffmann H., Fähling J., Leimeister J. M., Krcmar H. (2009) Kundenintegration in die Innovationsprozesse bei hybriden Produkten. In: Fischer S., Maehle E., Reischuk R. (eds.) INFORMATIK - Jahrestagung der GI. LNI 154. GI, Lübeck

HP Inc. (2016) Instant Ink https://instantink.hpconnected.com/ Last Access: 2016-02-18

Junginger M., Loser K.-U., Hoschke A., Winkler T., Krcmar H. (2006) Kooperationsunterstützung und Werkzeuge für die Dienstleistungsentwicklung: Die pro-services Workbench. In: Bullinger H.-J., Scheer A.-W. (eds.) Service Engineering. Springer-Verlag, Berlin/Heidelberg, pp. 593–621

Kagermann H., Riemensperger F., Hoke D., Helbig J., Stocksmeier D., Wahlster W., Scheer A.-W., Schweer D. (2015) Smart Service Welt Umsetzungsempfehlungen für das Zukunftsprojekt Internetbasierte Dienste für die Wirtschaft, Abschlussbericht Langversion

Kees A., Oberlaender A. M., Roeglinger M., Rosemann M. (2015) Understanding the Internet of Things: A Conceptualisation of Business-to-Thing (B2T) Interactions. In: Proceedings of the 23rd European Conference on Information Systems (ECIS 2015). Münster, Germany

Kiritsis D. (2011) Closed-loop PLM for intelligent products in the era of the Internet of things. In: Computer-Aided Design 43 (5), pp. 479–501

Klein P., Luetzenberger J., Thoben K.-D. (2015) A Proposal for Knowledge Formalization in Product Development Processes. In: Weber C., Husung S., Cascini G., Cantamessa M., Marjanović D. (eds.) Proceedings of the 20th International Conference on Engineering Design (ICED15)

Klingner S., Becker M. (2012) Formal Modelling of Components and Dependencies for Configuring Product-Service-Systems. In: Enterprise Modelling and Information Systems Architectures 7 (1), pp. 44–66

Knoke B., Thoben K.-D. (2014) Der Produktzustand als Basis für die Entwicklung produktnaher Dienstleistungen. In: Thomas O., Nüttgens M. (eds.) Dienstleistungsmodellierung 2014 Vom Servicemodell zum Anwendungssystem. Research. Springer Gabler, Wiesbaden, pp. 19–32

Kwak M., Kim H. (2013) Economic and Environmental Impacts of Product Service Lifetime: A Life-Cycle Perspective. In: Meier H. (ed.) Product-service integration for sustainable solutions Proceedings of the 5th CIRP International Conference on Industrial Product-Service Systems. Springer, Berlin, pp. 177–189

Laurischkat K. (2013) Computer-Aided Service Design for the Development of Product-Service Systems – Motivation and Benefits. In: Meier H. (ed.) Product-service integration for sustainable solutions Proceedings of the 5th CIRP International Conference on Industrial Product-Service Systems. Springer, Berlin, pp. 547–560

Lindström J., Dagman A., Karlberg M. (2014) Functional Products Lifecycle Governed by sustainable Win-Win Situations. In: Procedia CIRP 22, pp. 163–168

LML Steering Committee (2015) Lifecycle Modeling Language (LML) Specification Version 1.1. http://www.lifecyclemodeling.org/spec/LML_Specification_1_1.pdf

Lützenberger J., Klein P., Hribernik K., Thoben K.-D. (2016) Improving Product-Service Systems by Exploiting Information From The Usage Phase. A Case Study. In: Procedia CIRP 47, pp. 376–381

Manutelligence Consortium (2016) Product Service Design and Manufacturing Intelligence Engineering Platform http://www.manutelligence.eu/ Last Access: 2016-02-18

Meis J., Menschner P., Leimeister J. M. (2010) Modellierung von Dienstleistungen mittels Business Service Blueprinting Modeling. In: Thomas O., Nüttgens M. (eds.) Dienstleistungsmodellierung 2010. Physica-Verlag, Heidelberg, pp. 39–64

Meyer G. G., Främling K., Holmström J. (2009) Intelligent Products A survey. In: Computers in Industry 60 (3), pp. 137–148

Mikusz M. (2014) Towards an Understanding of Cyber-physical Systems as Industrial Software-Product-Service Systems. In: Procedia CIRP 16, pp. 385–389

Mikusz M. (2015) Towards a Conceptual Framework for Cyber-Physical Systems from the Service-Dominant Logic Perspective. In: Proceedings of the 21st Americas Conference on Information Systems. Puerto Rico

Neely A. (2008) Exploring the financial consequences of the servitization of manufacturing. In: Operations Management Research 1 (2), pp. 103–118

OMG (2011) Business Process Model and Notation Version 2.0 http://www.omg.org/spec/BPMN/2.0

OMG (2015a) Systems Modeling Language Version 1.4 http://www.omg.org/spec/SysML/1.4/

OMG (2015b) Unified Modeling Language Version 2.5 http://www.omg.org/spec/UML/2.5

Porter M. E., Heppelmann J. E. (2014) How Smart, Connected Products Are Transforming Competition. In: Harvard Business Review https://hbr.org/2014/11/how-smart-connected-products-are-transforming-competition

Psymbiosys Consortium (2016) Product-Service symbiotic systems http://www.psymbiosys.eu/ Last Access: 2016-02-18

RepRap Project (2014) Fused Filament Fabrication http://reprap.org/wiki/Fused_filament_fabrication Last Access: 2015-12-21

Scheer A.-W., Grieble O., Klein R. (2004) Model-based Service Engineering. In: Geberl S. (ed.) Impulse aus der Wirtschaftsinformatik 1. Aufl.. Physica-Verl., Heidelberg, pp. 17–33

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/ Last Access: 2016-11-25

SPEC Innovations (2016) Systems Engineering & Requirements Management Tools | Innoslate https://www.innoslate.com/ Last Access: 2016-02-18

Tietze F., Schiederig T., Herstatt C. (2013) Firms' transition to green product service system innovators Cases from the mobility sector. In: International Journal of Technology Management 63 (1/2), p. 51

Umeda Y., Takata S., Kimura F., Tomiyama T., Sutherland J. W., Kara S., Herrmann C., Duflou J. R. (2012) Toward integrated product and process life cycle planning—An environmental perspective. In: CIRP Annals - Manufacturing Technology 61 (2), pp. 681–702

Vasseur J.-P., Dunkels A. (2010) Interconnecting smart objects with IP The next internet. Elsevier Morgan Kaufmann, Amsterdam, The Netherlands

Velamuri V. K., Bansemir B., Neyer A.-K., Möslein K. M. (2013) Product Service Systems as a Driver for Business Model Innovation Lessons learned from the Manufacturing Industry. In: International Journal of Innovation Management 17 (1), pp. 1–25

Wellsandt S., Anke J., Thoben K.-D. (Feb. 2016) Modellierung der Lebenszyklen von Smart Services. In: Thomas O., Nüttgens M., Fellmann M. (eds.) Dienstleistungsmodellierung (DLM 2016). Springer, Berlin, Heidelberg, pp. 233–256

Wellsandt S., Hribernik K., Thoben K.-D. (2015) Sources and Characteristics of Information about Product Use. In: Procedia CIRP 36, pp. 242–247

Westphal I., Freitag M., Thoben K.-D. (2015) Visualization of Interactions Between Product and Service Lifecycle Management. In: Umeda S., Nakano M., Mizuyama H., Hibino H., Kiritsis D., von Cieminski G. (eds.) Innovative Production Management Towards Sustainable Growth. IFIP Advances in Information and Communication Technology 460. Springer, Cham, pp. 575–582

Wiesner S., Freitag M., Westphal I., Thoben K.-D. (2015) Interactions between Service and Product Lifecycle Management. In: Procedia CIRP 30, pp. 36–41

Winterhalter (2016) Pay-Per-Wash http://media.winterhalter.biz/mc/mediabase?assetId=418400&disposition=inline Last Access: 2017-07-10

Wolfenstetter T., Fuller K., Bohm M., Krcmar H., Brundl S. (2015) Towards a requirements traceability reference model for Product Service Systems. In: Framinan J. M. (ed.) The road ahead: understanding challenges and grasping opportunities in industrial and systems engineering IESM (Seville, Spain). Piscataway, NJ, pp. 1213–1220

Wortmann F., Flüchter K. (2015) Internet of Things. In: Business & Information Systems Engineering 57 (3), pp. 221–224

Zolnowski A., Christiansen T., Gudat J. (2016) Business Model Transformation Patterns of Data-Driven Innovations. In: Proceedings of the 24th European Conference on Information Systems (ECIS 2016). Istanbul, Turkey