19. SYMPOSIUM „DESIGN FOR X“
NEUKIRCHEN, 09. UND 10. OKTOBER 2008

DETECTING AND STRUCTURING REQUIREMENTS FOR THE DEVELOPMENT OF PRODUCT-SERVICE SYSTEMS

Patrick Müller, Rainer Stark

Abstract

Analysing and managing requirements effectively is important for a successful development of problem solutions independent of whether they are products, services, software or product-service systems. Approaches to detect, structure and formalize requirements are already proposed by many authors, e.g. Pahl and Beitz [1] or Ehrlenspiel [2]. Most of these guidelines are in general applicable to nearly every product in mechanical engineering, but there is still a lack concerning requirements from areas like organization, service delivery, information and communication technologies and new business models in most generic guidelines. Especially in product-service systems these areas become more important than in case of conventional product development.

Product-service systems (PSS) include product and service shares in one system. PSS are supposed to be integrated, customer- and lifecycle-oriented solutions, which are “sold” as one package. Modern types of business models are used to operate such systems more efficiently regarding technical, economical and ecological aspects. Instead of purchase (product-oriented) the customer pays for system functionality or for a defined result. The payment can be arranged in different ways such as pay-per-unit, pay-per-use or flat rates.

Supporting the generation, that is detection, structuring and formalization, of relevant requirements in PSS development is the purpose of this paper. A method to sustain this task will be introduced. In a first step it is used to “slices” a system into a set of pre-defined and optional “layers” containing physical and non-physical system elements to enable different views upon the PSS. Specially defined and clustered checklists are used to retrieve and formalize requirements in a second and third step.

1 Introduction and Motivation

1.1 Product-Service Systems

Product-service systems (PSS) integrate products and services into one development and delivery scope. They are supposed to be customer- and lifecycle-oriented problem solutions satisfying customer needs. The product and service shares (or modules) are packaged and offered as one integrated problem solution, cf. [3] and [4], embedded in modern types of business models. Especially in case of industrial product-service systems (IPS^2) the final aim is to operate such systems more efficiently regarding technical and economical aspects [5]. IPS^2 is a subset of PSS which represents PSS business-to-business solutions. The customer in fact pays for a system’s availability or a defined result instead of buying a product and additional after-sales services. The payment can be arranged in different ways such as pay-per-unit, pay-per-use or flat rates. The provider assures at least a minimum of function availability.

Figure 1 illustrates a simplified architecture of PSS core elements. Next to core products (a) and services (b), stakeholders (c) and contracts (d) are important. PSS are type of long-term commitments regulated by a contract. The contract provides tight linkages between stakeholders and defines how risk, responsibilities and costs, concerning the integrated delivery and operation of product and service shares are distributed among them, cp. [6]. Simplified the stakeholders are a group of one provider, multiple suppliers and one or more customer(s). They are typically organized in a locally distributed network with partly integrated organizational/business processes. An important aim is a value co-creation among the stakeholders during the integrated delivery (e). Supplemental systems and tools (f) have to be taken into account to enable the delivery of products and services and the exchange of information.

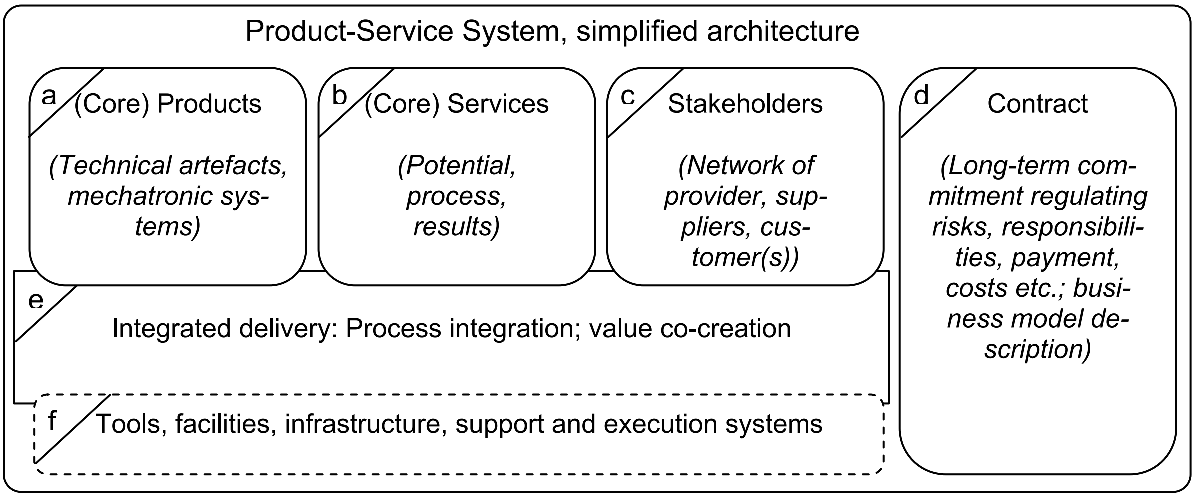

Fig. 1: A simplified architecture of PSS core elements (continuous lines) and complementary elements (broken lines)

B2C example from daily life: "Solutions for mobile communication"

- The main function of "connecting people by voice transmission over long distances" is for instance realized by a combination of mobile phones, which connect to cellular nets under contractual fixed conditions (price etc.). The mobile phone is often offered as an add-on to a contract which binds a customer and a provider of cellular nets for a longer period. The net provision can be considered as a service. The main stakeholders are here the net provider, a customer "*A"* calling another customer "*B"* (who is not necessarily contracted with the same provider) and the mobile phone manufacturer. The value for the customers is created in a process (activity chain) where a phone call is executed via the cellular net. The call process and the provision process lead to a value co-creation by and among multiple stakeholders. The main function here is finally implemented by the combination of a product (a, mobile phone), service provision (b, net provision) with multiple stakeholders (c) and a contract (d). Each component on its own is, only considering the main function, useless.

- An example with a deeper integration of products, services and added-value is for instance the BlackBerry concept, cp. https://de.blackberry.com (August 2008): The provider offers a full integrated information services as e. g. an e-mail service. BlackBerry mobile phones have keyboards which accelerate text typing (one letter or two letters on one key; enhanced functions) and thus they differs from most common mobile phones implementing T9 text editing (three letters on one key).

1.2 Requirements in a PSS context

Analysing and managing requirements effectively is important for a successful development of problem solutions independent of whether they are products, services, software solutions or product-service systems. The planning and concept design for new product-service systems concerns decisions on the amount and allocation of products and services within a PSS to implement required functionality. Furthermore the product and service shares within a PSS can change over its lifecycle, to adapt the PSS to changing conditions und customers needs under the conditions of the contract, cp. [6]. Questions like where, when and by which network partner a service has to be provided to the customer, which degree of process integration is needed, how responsibilities are distributed among all stakeholders end up in organizational requirements. Questions on lifecycle costs and contracts have to be answered. Such issues appear latently in known requirements generation approaches but they are not “made visible” explicitly for integrated combinations of products and services.

1.3 Problem statement

Next to economical and technical requirements those addressing organization structures, process organization and business models have to be captured to support a holistic system investigation. In general, this is expressed in various approaches in literature but predominantly only for products or services and not explicitly for integrated solutions. There are no checklists that focus on aspects which help to search for relevant requirements concerning the connections between products and service processes or activities and the hybrid value creation. Furthermore, there is no formal description how and where hybrid value creation takes places in a PSS. Thus, in early development phases, it is difficult to compare PSS ideas, to derive solution independent requirements, which not only concern product or service characteristics, and to compare PSS concepts. The lack of a kind of meta-model for product-service systems makes a clear discussion on the systems requirements even more difficult. In addition interdisciplinary cooperation, e. g. between engineers and economists, leads to different views upon the same system, and thus to different sets of requirements.

1.4 Objectives / Intensions

The purpose is to support the early phases in PSS development with an appropriate method to lead a structured discussion on PSS ideas and to support an integrated requirements generation. A basic condition is a common understanding of PSS (and IPS^2) main “dimensions” among all who apply this method. Developers (engineers), marketing people or involved customers are supposed to be possible users. The objective is to come up with an integrated meta-model and a requirements retrieval checklist for PSS/IPS^2.

Research questions, driving the study are as follows:

- How does a PSS/IPS^2 meta-model that provides the idea and requirements generation in a simple but structured way look like?

- What combines products and services and affects the hybrid/integrated value creation?

- How can the meta-model be used to reason on PSS requirements and which points have to be checked to derive and formalize relevant requirements?

Answering these questions in details is not the purpose of this paper, although they drive the setup of the new method. The intention is to give an overview on all elements of the new method and to position the method in the landscape of research on PSS, IPS^2 and requirements.

1.5 Research Approach

The method is generated from an engineering point of view. It is based on a background in mechanical engineering and it was designed after a literature study on PSS, service engineering and requirements analysis and management. It has been tested for the first time in a workshop with PSS and engineering design researchers. It was applied to the example of a "micro-manufacturing system with PSS business model", which is the demonstration example in a German IPS^2 research project called Transregio 29 (www.tr29.de). The feedback of the participants was used to redesign the method. Furthermore the method is enhanced by application and test on other PSS/IPS^2 examples. Its transferability to other technical systems seems to be promising and the implementation in a computer supported tool possible.

2 Requirements generation approaches – a small sample

Pahl and Beitz [1] propose a well known guideline to retrieve requirements for products. This guideline refers to customers, designers, lifecycle phases, cost and time as sources for requirements. It considers many areas which are mirrored by design for X guidelines (e. g. design for assembly). The mentioned areas are broad and the designer has to search for required system properties on his own. The model of Ehrlenspiel [2] is hierarchical and has two main branches, (i) technical-economical requirements and (ii) organizational requirements. To search for requirements, these are detailed in a tree structure. Below the leaves of the tree (which are pure technical requirements, technical periphery, law, human, society, cost, time, staff and tools) are "question terms" listed to retrieve requirements from various system life phases. These question terms are very heterogeneous, considering the abstraction level; examples: "Manufacturability?", "Transport problems?", "Maintenance duration?" or "Training?". Ahrens [7] compares various approaches to retrieve and manage requirements, but she doesn't compile a new checklist to retrieve requirements in product development. Jaschinski [8] elaborates on the process of a quality-oriented redesign of services but he gives no guideline or checklist which helps to retrieve requirements on service properties in detail. In his thesis van Husen [9] comes up with checklists to discover stakeholders and influencing factors (strategic, economic, legislative and social factors and boundary conditions) to analyse requirements for product-related services. Steinbach [10] delivers a comprehensive list of service characteristics and properties collated from many business approaches and sources; examples of service properties mentioned: friendliness, responsiveness, patience, duration of delivery, reliability etc.

3 A PSS ideas and requirements generation approach

In the proposed method general ideas and concrete system properties from all approaches and sources mentioned in section 2 have been taken into account. Both types of requirements, the ones directly retrieved from customers as well as those from the PSS context and network, shall be investigated by the same method. The next sections lead through the three steps of this method.

3.1 A process in three steps

The application of this method is divided into three main steps which may occur iteratively during the planning and concept phase in PSS development. Two general ways of application may happen:

1. The starting point can be an already existing product which the designers want to include in or transform into a PSS business model. In such a case they can use the proposed method to run a systematic variation, cp. [1], of the entire system to detect new requirements from the PSS context. *(Bottom-up use)*

2. Starting with customer needs, which differ significantly from (technical) requirements, cf. [11]. *(Top-down use)*

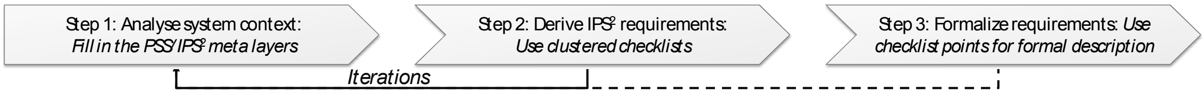

Fig. 2: Iterative steps of the proposed PSS requirements generation method

In the first step the (future) system is "sliced" into a set of pre-defined and optional "layers" containing physical and non-physical system elements. Specially defined and clustered "requirements search lists" (type of checklists) are used to retrieve and formalize requirements in a second and third step. The steps of Figure 2 are described in detail in the following subsections.

3.2 PSS specific meta-layers (Step 1)

Different views on the system are graphically layered to simplify the representation, Figure 3. Each view has to be filled by simple sketches, models or text. If a technical artefact is chosen as a starting point, its product context (vertical dimension) is analyzed over lifecycle perspectives (horizontal dimension) in detail. If a customer need is chosen as starting point the, development engineer can define totally new system elements on each layer. System elements which have direct connections are mapped vertically one above the other. Already detected or determined system elements can help to identify other system characteristics which have to be detailed in the form of infrastructure or resources or which have to be ensured in the contract. The layers can be used to elaborate on single phases/episodes of the PSS lifecycle, if not on the whole lifecycle.

The order of the layers is not important. The designer may reorder them or even add new optional layers to bring in new points of interest. "Sustainability" for instance is often mentioned as driver for PSS [4]. If a designer wants to make a system more sustainable, he may add a sustainability layer to record ideas on how and where to improve a system or future PSS. The most important layer is the "lifecycle activities" layer. Activities are kind of connectors of primary system elements, viz. services, products and stakeholders. Activity chains (processes) lead to a result/use/benefit which is the final aim of the PSS, cp. [12]. The layers "needs" and "value (benefits)" represent the customers view upon the PSS. All other layers are kind of design layers representing the solution space for the designers. There may be an iteration from filling the "needs" layer over filling the design layers to filling the "value/benefits" layer and going back to the "needs" layer. Such iterations can be used to commit with a customer on a PSS/IPS^2 which satisfies his needs best.

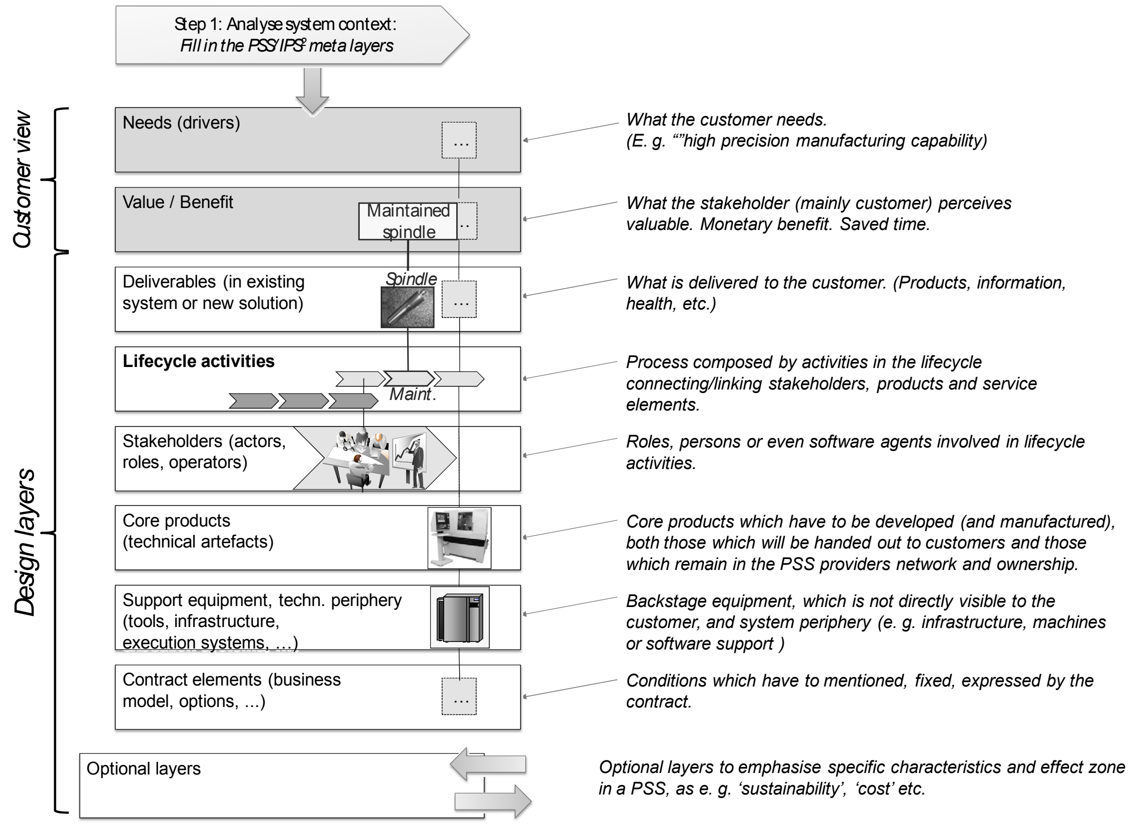

Fig. 3: Step 1 – Filling in the PSS specific layers (e. g. for the transformation of a conventional micro-manufacturing system into an IPS2). Simple sketches and text can be used to model.

3.3 Clustered PSS requirements checklists (Step 2)

After having determined certain system elements it is possible to derive structured requirements for the entire system. To be able to search and tackle individual requirements coarse categories (or clusters) for requirements have been defined, which contain expandable listings of (PSS specific) system properties (e.g. frequencies of activities). These categories correlate with the layered views from step 1 and they are extended by technological and organizational categories, according to the explanations from section 1.1. The listed system properties are used like checklists to retrieve relevant requirements in a structured way, cp. Figure 4.

In total about 100 checklist points (system properties) have been summed up and clustered. A great number has been collated from references discussed in section 2. Many additional points have been added by the authors based on a “activity-oriented thinking”, according to section 3.2. The tree structure in Figure 4 was used to compress the contents, it has no hierarchical meaning. The pairs relate to each other more than to other categories, but relations exist among all categories. The left column is more “object oriented” (static elements) and the right column more “process oriented”.

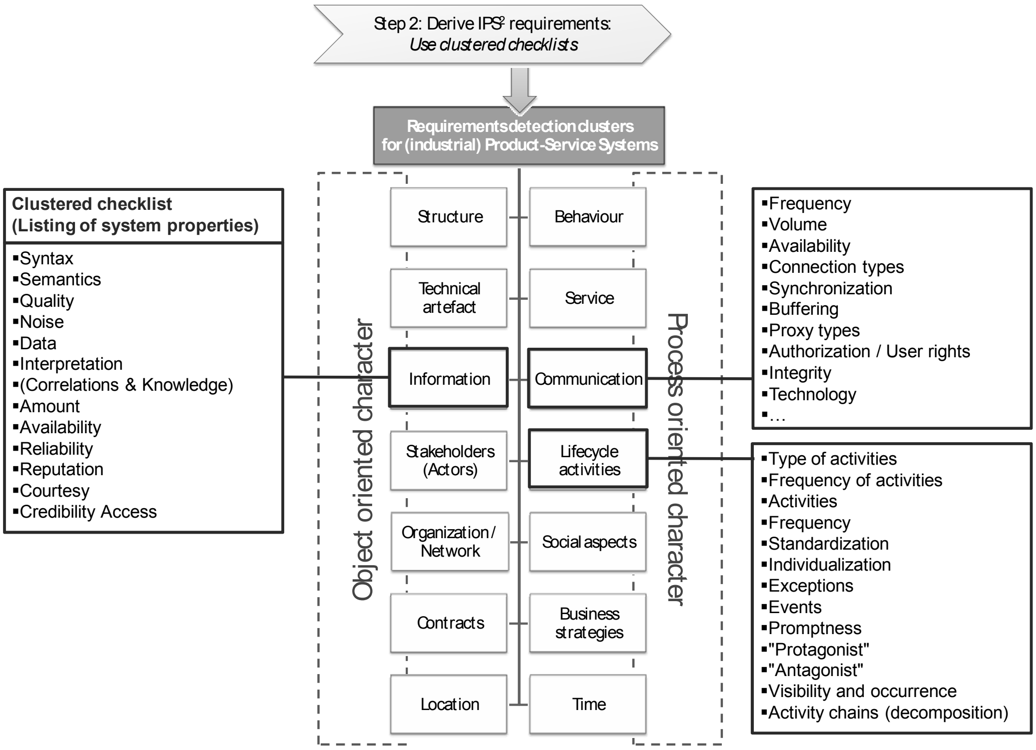

Fig. 4: Step 2 – Applying clustered checklists to search for requirements (three categories exemplary highlighted)

3.4 Requirements formulization (Step 3)

All detected, relevant requirements are formalized according to the requirement categories and checklist points (system properties), cp. Figure 5.

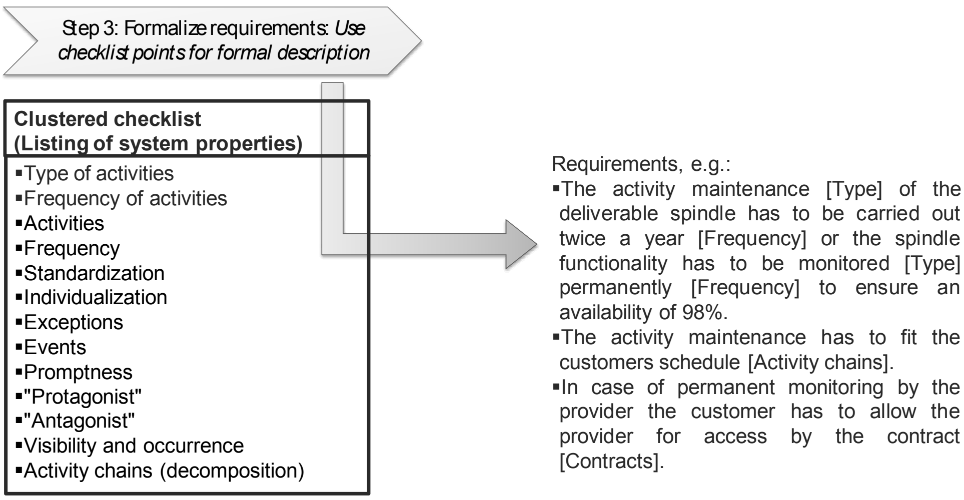

Fig. 5: Step 3 – Formalizing detected requirements

3.5 Composing all elements of the approach – "micro-manufacturing" example

Figure 6 composes all elements introduced before. A manufacturing system was chosen as example. The need of the customer is for instance the capability to manufacture a minimal number of parts with high precision in a given amount of time. What the PSS provider may offer is a spindle and the assurance that the spindle is available in 98% of the required run-time. If the spindle breaks, the provider is responsible for an adequate compensation. He may exchange the spindle or even manufacture parts for the customer during the time which is necessary to maintain the manufacturing system or spindle. The way which is chosen to deliver the required availability of 98% has an effect on the requirements of the system elements (spindle or even manufactured parts; infrastructure; operators). The provider may assure the availability by scheduled system inspections or by permanent monitoring and state-oriented maintenance. The first variant requires free slots in the manufacturing processes of the customer. The second variant requires permanent authorization for access via IT systems. (Figure 6 is limited to some known elements to reduce the illustration's complexity.)

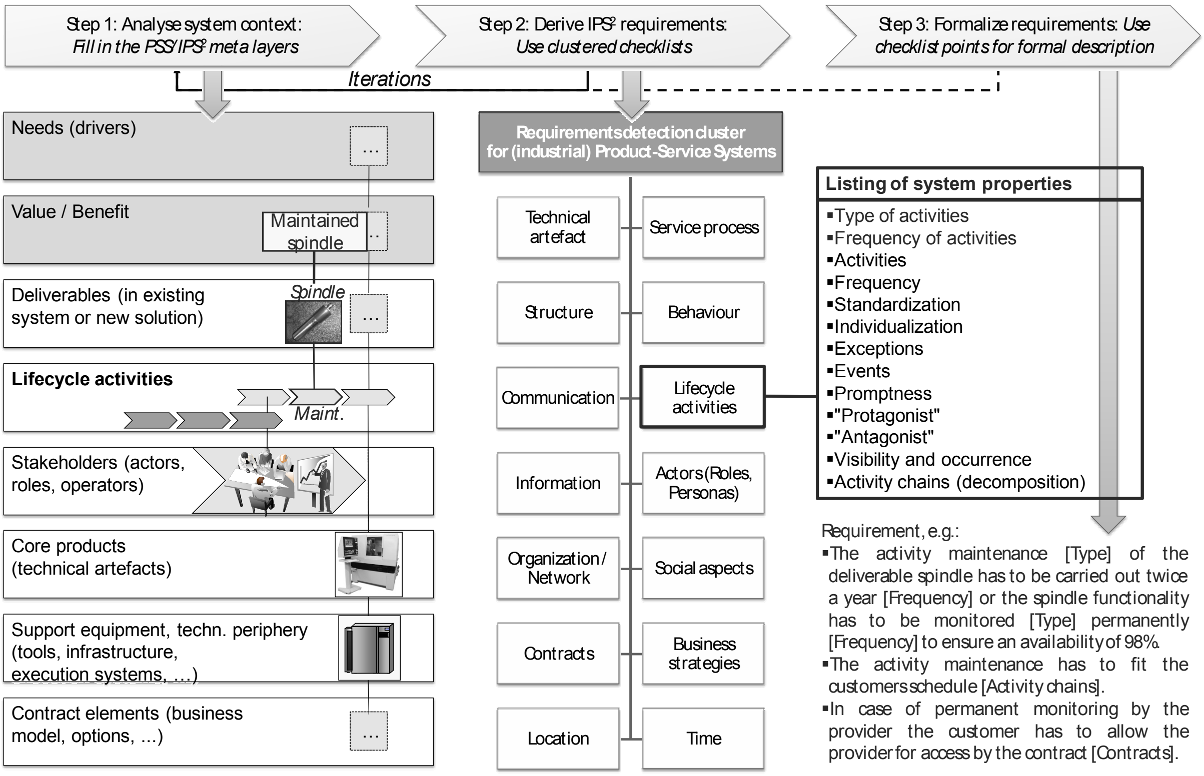

Fig. 6: Composed PSS idea and requirements generation in three steps

4 Discussion and Conclusions

The main purpose was the support of the early and creative phases in PSS development, i. e. the PSS planning and conceptualization. The state described in this paper is type of a first evolution step of a method for the generation of PSS ideas and requirements; thus the paper has the characteristic of a "working paper".

The method provides a layered PSS meta-model, which can be used to document and compare PSS ideas for the communication among stakeholders of different background. The clustered checklists for searching requirements are not at a first glance PSS specific, but a closer look shows that many checklist points (not mentioned in detail in this paper due to place restrictions) relate to service quality criteria and to activity related system properties; both important for a discussion and design of PSS and IPS^2.

The claim that this method is especially appropriate for PSS development is based on its activity-orientation, on the introduction of a simple PSS meta-model and the clustered checklists which consider many activity-oriented and thus PSS-relevant properties. First experiences gained from the method's application during a workshop with PSS researchers are promising and resulted in first enhancements. The feed-back of research partners within the research project Transregio 29 “Engineering of Industrial Product-Service Systems” (www.tr29.de) also showed that there is a potential and relevance for the further development of the method. Its applicability on “real life cases” will be investigated soon. So far few experiences exist on the effort which is necessary to run the method through all three steps in every single detail.

Filling up the layers in step 1 is an iterative process for its own, iterating between the customer need(s), possible PSS solutions and the value for the customer, cp. section 3.2. As input for step 1 it is possible to use the Activity Modelling Cycle (AMC) published by Matzen and McAloone [12]. The AMC shows which stakeholders are interacting in which lifecycle activity. This information is helpful to fill the layers in the proposed method. (The Service Design Strategies, presented by Tan, Andreasen and Matzen in 2008, can be used to vary the (future) PSS, modelled in step 1, systematically to discover potential for more added value in the future PSS. In principle the method is also compatible with the Japanese Service Development Methodology by Arai and Shimomura which is focusing on attitudes and characteristics of individual actors interacting in activity chains.)

To implement more robustness the descriptions of all layers have to be refined and a clear mapping between the layers in step 1 and the clustered checklists in step 2 have to be defined. The way of formalizing detected requirements in step 3 (syntax and semantics) will be defined according to up-to-date standards. Precise requirements for this will be worked out for the next evolution step of the method.

5 Outlook

Methods such as the House of Quality, cp. [13], UML use-case diagrams, role concepts, or the soft systems approach have not been compared with the new method so far. It has to be figured out if there are worthy benefits in these methods which can be transferred or combined with the new one. Approaches to retrieve and manage requirements in software development, for instance included in the ITIL (IT Infrastructure Library), will be investigated. Literature from the discipline requirements engineering will also be considered in more detail.

A deeper evaluation of the method is planned in three ways: (i) Application of the method’s next evolution step during the exemplary development of the PSS micro-manufacturing system in the TR29. (ii) Application on the example of “decentralized energy supply with solar home systems (SHS)”. Such systems are used in low developed regions, coupled with micro-financing strategies, and combining B2B and B2C in one context. These SHS additionally have a need for certification. Finally they are a wonderful PSS example. (iii) Application within workshops with researchers form the PSS community and industrial partners.

Future research steps will address the mapping of the complete list of PSS requirements to appropriate realization options which constitute a mixture of traditional technical solution modules (products) with service activities. The support by a computer tool is an option which has to be assessed considering relevance, benefits, and effort after the method has been evaluated.

6 References

[1] Pahl G., Beitz W., Feldhusen J., Grote K. H.: Engineering Design – A Systematic Approach. Third Edition, Springer-Verlag, London, 2007.

[2] Ehrlenspiel, K.: Integrierte Produktentwicklung – Methoden für Prozessorganisation, Produkterstellung und Konstruktion. Carl Hanser Verlag, München, Wien, 1995.

[3] Matzen, D., Tan, A. R., Andreasen, M. M.: Product/service-systems: Proposal for models and terminology. Beiträge zum 16. Symposium Design for X, Lehrstuhl für Konstruktionstechnik, Technische Universität Erlangen, 2005, pp. 27-38.

[4] McAloone, T. C., Andreasen, M. M.: Design for Utility, sustainability and societal virtues: Developing Product Service Systems, Proceedings of the International Design Conference, 2004.

[5] Meier, H., Uhlmann, E., Kortmann, D., 2005, Hybride Leistungsbündel, Nutzenorientiertes Produktverständnis durch interferierende Sach- und Dienstleistungen. wt Werkstatttechnik online Jahrgang 95 H. 7/8, S. 528-532.

[6] Steven, M., Welp, E. G., Richter, A., Sadek, T.: Verfügbarkeitsorientierte Geschäftsmodelle und Flexibilität industrieller Produkt-Service Systeme. wt Wertstattstechnik online, year 98, journal 7/8, 2008.

[7] Ahrens, G.: Das Erfassen und Handhaben von Produktanforderungen – Methodische Voraussetzungen und Anwendung in der Praxis. Maschinenbau und Produktionstechnik, TU Berlin, 2000.

[8] Jaschinski, C.: Qualitätsorientiertes Redesign von Dienstleistungen. FIR, RWTH Aachen, 1998.

[9] van Husen, C.: Anforderungsanalyse für produktbegleitende Dienstleistungen. IAT Universität Stuttgart, Fraunhofer IAO, Jost-Jetter Verlag, 2007.

[10] Steinbach, M., Bley, H. (edt.), Weber, C. (edt.): Systematische Gestaltung von Product-Service Systems – Integrierte Entwicklung von Product-Service Systems auf Basis der Lehre von Merkmalen und Eigenschaften. Lehrstuhl für Konstruktionstechnik/CAD, Saarbrücken, 2005.

[11] Ericson, Å., Müller, P., Larsson, T., Stark, R.: Product-Service Systems – From Customer Needs to Requirements in Early Development Phases. CIRP IPS^2 Conference, Cranfield, 2009. *(Paper handed in for review.)*

[12] Matzen, D., McAloone, T.: A tool for conceptualising in PSS development. Beiträge zum 17. Symposium Design for X, Lehrstuhl für Konstruktionstechnik, Technische Universität Erlangen, 2006, pp. 131-140.

[13] Danner, S.: Ganzheitliches Anforderungsmanagement für marktorientierte Entwicklungsprozesse. Lehrstuhl für Konstruktion im Maschinenbau, TU München, 1996.

Patrick Müller, Dipl.-Ing.
Industrial Information Technology
Technische Universität Berlin
Pascalstraße 8-9, D-10587 Berlin
Tel.: +49 (0)30 / 3900 6294
Fax: +49 (0)30 / 3930 246
Email: patrick.mueller@ipk.fraunhofer.de
URL: http://www.iit.tu-berlin.de