# Requirements Engineering for Product Service Systems

## A State of the Art Analysis

As product service systems (PSS) have specific characteristics, they impose special requirements on requirements engineering (RE) that are initially explained by means of nine criteria. Based on that, main approaches of RE in product, software, and service engineering as well as integrated approaches are evaluated in terms of their suitability for PSS. The result shows different maturity levels of RE in all domains, being highest in product and software engineering. However, we must note a major deficit in the overall cooperation due to the individual concepts of the respective domains. A direct transfer to other areas, as needed for PSS, is not possible.

### The Authors

**Dipl.-Inf. Marina Berkovich**
**Prof. Dr. Helmut Krcmar**
Lehrstuhl für Wirtschaftsinformatik
Technische Universität München
Boltzmannstr. 3
85748 Garching bei München
Germany
berkovic@in.tum.de
krcmar@in.tum.de

**Prof. Dr. Jan Marco Leimeister (✉)**
Fachgebiet Wirtschaftsinformatik
Universität Kassel
Nora-Platiel-Str. 4
34127 Kassel
Germany
leimeister@uni-kassel.de

**Electronic Supplementary Material**
The online version of this article (doi: 10.1007/s12599-011-0192-2) contains supplementary material, which is available to authorized users.

## 1 Introduction

Recently, manufacturing companies and service providers are faced with new challenges in terms of increasingly competitive pressure and more complex customer requirements. Offering goods or services is often not sufficient for companies to differentiate themselves from competitors (Leimeister and Glauner 2008). Further, customers do not want products or services per se, but they demand individual solutions of their problems. As a consequence, many vendors are changing their strategy from "product-centric" to "customer-centric" (Galbraith 2002), and becoming solution providers (Davies 2004). By supplying a customized and integrated bundle of hardware, software, and service elements, the customer problem is solved completely. These bundles are known as product service systems (PSS) or hybrid products (Becker et al. 2008).

A well-known example for PSS represents photocopiers in the context of the "performance-based pricing" model in the commercial sector (the user only pays for printed pages; he does not buy a photocopier and does not maintain it). In the IT industry, software as a service describes a widespread PSS (Böhmann et al. 2008). This business model includes standard software offered as a service, accessible via the Internet, and adapted to customer's wishes (Böhmann and Krcmar 2007). Additionally, the hardware component, optimized in the computer center of the provider and utilized for high-performance data storage, also contributes to the market success of the solution (Grohmann 2007).

*Motivation.* The development of PSS is mainly affected by the coordination of the involved domains, such as product, software, and service engineering (PE, SE, SER, respectively), by the different life cycles of the components and their interdependencies, as well as by a high degree of customer interaction (Sturm and Bading 2008). In order to handle the resulting complexity, a complete understanding of the overall solution and its features is essential. In this connection, requirements engineering (RE) plays a key factor (Cheng and Atlee 2007). Its main

tasks comprise the extensive identification of the solving problem in form of requirements and constraints, their management, traceability, and description in an adequate level of detail throughout all development stages (Hull et al. 2004, pp. 6–8). In literature, none of the RE approaches takes the requirements to PSS into full account. Moreover, a structured collection and analysis of the criteria for a successful RE concerning PSS do not exist.

*Objective and contribution.* The objective of this article is to point out all requirements that have to be fulfilled by the RE of PSS, and to determine the suitability of existing RE approaches for PSS. Thereby, the selection criteria are based on the main characteristics and RE tasks area in the lifecycle of PSS. They serve for the identification of research deficits in RE for PSS. The results of this work are a basis for future research in the area of RE for PSS, with the goal of creating an integrated requirements management for PSS and a development approach for PSS relying thereupon. To achieve this, first, a conceptual and logical connection between the requirements models of PE, SE, and DE needs to be achieved.

*Preliminaries.* The presented literature analysis is based on other publications, which either contain domain-specific analyses of RE approaches in product engineering (Berkovich et al. 2009c) as well as preliminaries of the analytical framework (Berkovich et al. 2009a), or refer to empirical studies of RE methods in practice and their challenges in the development of PSS (Berkovich et al. 2009b). Additionally, process models for the development of PSS were analyzed with regard to their coverage of the various life cycle phases and of the involved domains (Langer et al. 2010). Furthermore, a reference framework for an integrated RE model concerning PSS was developed (Berkovich et al. 2010). Picking up on this, the following article extends the present scheme of analysis significantly, by deriving it from a life cycle assessment of PSS and using it for the subsequent literature analysis. Even the criteria, as well as the referenced sources, were refined substantially and extended, respectively.

*Structure.* The structure of the rest of this paper is orientated towards the implemented steps.

Section 2 contains a methodology for the literature search and analysis. Section 3 introduces related works, after which Sect. 4 illustrates the different tasks of RE in the life cycle of PSS. Section 5 describes the derivation and explanation of the analysis criteria, with Sect. 5.1 describing the framework used to introduce the evaluation criteria of the analyzed literature in Sect. 5.2. Section 6 then summarizes the identified approaches in order to give an overview of the existing literature. Subsequently, the results of the literature analysis are presented in accordance with the criteria of Sect. 5. Section 7 then discusses the analysis and evaluates it. Section 8 concludes the article with a summary and outlook for future research.

## 2 Methodology

As the number of academic studies increases, systemic and exhaustive literature analyses become more and more important (Webster and Watson 2002). A literature analysis enables the collecting of existing experience and knowledge of one particular aspect derived from both science and practice, thus identifying gaps in research (Budgen and Brereton 2006).

To ensure the scientific value of a literature analysis, it is necessary to pay special attention to an intersubjectively verifiable study design and procedure (Torraco 2005). Accordingly, the analysis in this paper is based on the approach of Webster and Watson (2002), which is characterized by the assignment of the concepts present to the respective authors. The procedure and structure of the paper, however, have been extracted from the approaches of Brereton et al. (2007) and Kitchenham et al. (2009). The individual steps of our approach are shown in the Appendix 1 (available online). Thus, the central question of the literature analysis is: "To what extent are the selected domain-specific approaches of RE suited for PSS?" and "Which of these can be used in the development process of PSS?"

Due to the different terms and understanding of RE in product, software, and service engineering, an individual literature search was performed for each domain. For searching the relevant literature, the portal "Google Scholar" was chosen, which covers a major part of the journals and electronic publications in informatics as well as in engineering. Meier and Conkling (2008) have shown that 90% of the publications in engineering, published after 1990, are available in "Google Scholar." In the field of computer science, important publishing companies (e.g., Elsevier) and non-profit organizations (e.g., ACM or IEEE) provide different articles in this section (Meier and Conkling 2008; Jacsó 2008).

All hits of the search portal were sorted by the frequency of their citations. Afterwards, the 100 most cited articles of each domain were examined and sorted into the following categories:
(a) Articles that constitute a generic approach/process model: the generality and spread of the respective article comprised the selection criteria for this category (see Goeken and Patas 2010).
(b) Articles describing a specific topic of RE: this category includes articles addressing specific topics of RE, such as different activities and techniques. Each article in this category is mapped to an RE activity proposed by the generic approaches.
(c) Articles that do not consider RE: these papers are not considered here.
In **Tab. 1** the sources of the literature describing generic approaches are listed. The specific sources are referenced in the Appendix 1 (available online).

## 3 Related Work

Literature related to the topic of RE for PSS can be divided into two parts:
(a) analyses and literature reviews of RE issues in the different domains that do not consider PSS, and
(b) research and analysis of process models for the development of PSS.
The first category includes articles about analyses of existing literature on RE. Byrd et al. (1992) have analyzed techniques for requirements elicitation and knowledge acquisition regarding their similarities and differences. Wieringa et al. (2006) have developed a classification scheme for contributions to research on RE. The goal of their classification scheme is to detect the different types of scientific articles, but not to identify different topics within RE. Hickey and Davis (2003) have collected several criteria by means of interviews that can be used by requirements analysts to select from a wide range of techniques for requirements elicitation. Thus, they focus on the selection criteria and not on comprehensively listing RE techniques.

The second category includes articles about literature analyses of process models for PSS. For instance, Gräßle and

**Table 1** Analyzed approaches

| Selected product engineering approaches | Selected software engineering approaches | Selected service engineering approaches | Selected approaches for integrated development of PSS |
|---|---|---|---|
| • Vorgehenszyklus für die Lösungssuche (Ehrlenspiel 2002) | • Requirements Engineering Framework (Pohl 1993, 2007) | • Design and Management Service Processes (Ramaswamy 1996) | • Integrated Product and Service Engineering versus Design for Environment (Lindahl et al. 2007) |
| • Engineering Design (Pahl et al. 2006) | • Requirements Engineering Process (Kotonya and Sommerville 1998) | • Key Concepts for New Service Development (Edvardsson and Olsson 1996) | • Vorgehensmodell der hybriden Produktentwicklung (Spath and Demuß 2003) |
| • Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte (VDI 2221 1993) | • Requirements Engineering Process (Lamsweerde 2009) | • Ein Rahmenkonzept für die systematische Entwicklung von Dienstleistungen (Bullinger and Schreiner 2003) | • Life Cycle Management of Industrial Product-Service Systems (Aurich et al. 2007) |
| • Product Design and Development (Ulrich and Eppinger 2003) | • A Generic Process for Requirements Engineering (Hull et al. 2004) | • Dienstleistungsproduktion (Frietzsche and Maleri 2003) | • Rahmenkonzept zur Entwicklung von Product-Service Systems (Botta 2007) |
| • Erfassen und Handhaben von Produktanforderungen (Ahrens 2000) | • Engineering and Managing Software Requirements (Aurum and Wohlin 2005) | • Anforderungsanalyse für produktbegleitende Dienstleistungen (Husen 2007) | • Systematische Überfuhrüng von kundenspezifischen IT-Lösungen in integrierte Produkt-Dienstleistungsbausteine mit der SCORE-Methode (Böhmann et al. 2008) |

Dollmann (2010) describe process models for the development of PSS, whereby they analyze the composition and structure of process models. Thus, they do not describe RE in detail, but only mention it as one important life cycle activity within the development process. It is thus not possible to derive criteria from this article for the structuring of task areas for RE for PSS.

The articles of the first section focus on software engineering; however, they cover only certain RE activities and primarily present criteria for the analysis of the RE techniques that are offered for the RE activities. However, none these articles includes a comprehensive analysis of the RE task areas. The publications of the second section, as well as our previous work (Langer et al. 2010), deal with the issue of process models that consider RE only rudimentarily. Single task areas of RE are not analyzed in these works.

## 4 Requirements Engineering in the Life Cycle of PSS

In order to clarify the role of RE in the life cycle of PSS, we contrast the task areas of RE and the life cycle phases of RE, bearing in mind the characteristics of PSS. RE has the goal of systematically determining the requirements to a product correctly and completely (Byrd et al. 1992). In this way, RE creates the basis for all following development steps (Spath and

Demuß 2003). As errors in the requirements specifications often cause project failures and cost-intensive changes at a later point in time (Pohl 2007), RE is seen as a crucial factor in the development process. According to Pohl (2007), RE should therefore elicit all requirements, analyze and represent them in a necessary degree of detail, reach a sufficient agreement on the known requirements, and document them in accordance with defined directions. If any changes in the requirements occur, RE should also offer corresponding measures to handle the changes.

According to Tuli et al. (2007), the life cycle of PSS comprises four main phases, namely, customer requirements definition, customization and integration of goods and/or services, deployment, and postdeployment customer support. These phases and their relation to RE are described in the following section.

*Customer requirements definition*: In this phase, the requirements to the PSS are elicited and analyzed (Spath and Demuß 2003). Particularly, the customer needs are in the foreground of requirements elicitation, since they serve as a basis for the subsequent implementation (Sawhney 2006). By fulfilling this type of requirement as best as possible, PSS create high added value in the form of an individual *problem solution for the customer* that cannot be realized by consuming standardized products or services. It is thus important to interact with

the customer in the entire development process, from requirements elicitation to requirements validation (Tukker 2004). The elicited requirements must be validated and verified in cooperation with the customer (Tukker 2004).

Requirements emanating from the value creation processes of the customer must also be elicited because the PSS is *integrated into the value chain processes* of the customers after its deployment (Böhmann and Krcmar 2007; Zellner 2008). The PSS has to be developed in a way that it can be integrated into the customer's existing system environment.

Furthermore, the perspectives on the different domains have to be considered in requirements elicitation of PSS. This is attributable to the *bundling of material and immaterial components* that are not recognizable as individual parts in the final solution, but they are formative for the characteristics of the PSS (Leimeister and Glauner 2008).

Having collected all relevant requirements, they are documented in a way that provides a common understanding for all domains participating in the development process. Based on the documented requirements, the modules that are understood as components of the PSS for the solution are designed. Additionally, the existing modules are evaluated for their reusability. In this context, modularization means that the PSS consists of different modules or components or

partial services (Böhmann et al. 2008; Beverungen et al. 2008).

Although *modularization* does not constitute a defining characteristic of PSS, it is essential for the reusability of the existing components as well as for a flexible adaptation of the PSS based on existing modules to changing customer wishes (Beverungen et al. 2009). Consequently, the modules can be standardized and recombined for creating various products. Therefore, RE must identify existing components and services, and combine them into modules in order to prevent their redevelopment.

In the next step, a product structure consisting of material and immaterial components is created to illustrate the complete PSS. To achieve this, RE has to concretize the solution-independent target properties (initial requirements) written in the language of the stakeholders. Concretization means the requirements are supplemented with detailed qualitative and quantitative information, and they are translated into solution-oriented design requirements that are divided according to hardware, software, and service elements and that are composed in the language of the developers (Husen 2007). If conflicts between the requirements arise, they have to be identified and resolved. After the concretization, the requirements are presented to the stakeholders in order to explore whether or not they meet their wishes.

*Customization and integration of goods and/or services*: According to Tuli et al. (2007), this phase comprises the coordinated and integrated development of the different components by the respective domains, and they are combined in an overall solution.

The requirements can change during the implementation. Therefore, the RE has to gather all changes in the specification, identify their effects on other requirements and components of the PSS, and provide measures for handling the changes. In this context, it is important to trace the life cycle of each requirement through every phase of the development in order to identify the interdependences between all requirements and components of the PSS and to keep them updated.

*Deployment & postdeployment customer support*: In the phase of deployment, the PSS is integrated into the value chain process of the customer. The postdeployment phase focuses on supporting, maintaining, and disposing of the PSS after the

end of the product life cycle (Tuli et al. 2007). For a successful implementation of both phases, the provider of the solution must know the prerequisites given by the customer. Due to these facts, an integrated development of all PSS components is essential (Spath and Demuß 2003). If subsequent changes in the requirements occur, the task of RE in that case is to update the current specifications and to keep them up-to-date.

The following paragraph shows an example for a PSS in the medical sector that illustrates the complexity of its development and the role of RE.

A former pure production enterprise that focuses on medical systems has recently provided an integrated solution for the sterilization of surgical instruments for hospitals, whereby the customer pays depending on the product usage. The main advantage of the integrated solution for the consumer is the decrease in fixed costs. The customer receives clean instruments for each operation coordinated with the schedule, which can be transmitted to the provider's system through an interface. Consequently, the PSS should include an adequate software program integrated into the hospital information system, a service organization targeted towards the on-demand-needs of the customer needs, as well as sterilizing plants offered by the supplier and applicable for each customer solution. All these components describe an integrated bundle and can hardly be used separately.

A major challenge for RE during the development of this PSS was the concretization of, and coordination between, the requirements on goods and services. Based on the prerequisites given by the hospitals, the requirements concerning the center for sterilization of surgical instruments were derived, and used for the definition of the service requirements (e.g., transport of the surgical instruments or consideration of all operating schedules in the planning of the provider). During requirements elicitation, different aspects comprising the size and sterilizing conditions of the transport boxes for the surgical instruments, as well the calculation of routes, cars, and drivers were taken into consideration. These temporal and localized requirements on the transport had to be adjusted with respect to the requirements on the integration of the provider's software into the hospital information systems. As the numerous stakeholders had heterogeneous expectations, the resolution of conflicts

in the specification was complicated, too. Particularly, the cross-domain links between the requirements represented a difficult task that could not be handled by applying traditional RE methods.

## 5 Derivation of the Analysis Criteria

In the following sections, a framework is developed that is needed for the derivation of the criteria. The goal is to illustrate the subject area of general approaches in RE for PSS in order to structure older and future research (Goeken and Patas 2010). The framework (Sect. 5.1) describes the life cycle and characteristics of PSS, as well as their relationship to the RE task areas. Subsequently, the criteria for the literature analysis are identified (Sect. 5.2).

### 5.1 Framework for the Subject Matter of Requirements Engineering for PSS

In order to get the criteria for the analysis of the RE approaches in product, software, and service engineering, as well as in the integrated development of PSS, a framework is created. The task is not only to illustrate the subject area of RE for PSS, but also to make the background knowledge and assumptions of the researchers explicit (see Goeken and Patas 2010). This is particularly important in RE, since there is no possibility of referring to theories for supporting research work.

The development of the framework is based on the dimensions of an architectural concept in the IT sector proposed by Vogel et al. (2009) in order to structure the knowledge and experiences of one topic and to provide orientation. Generally, a framework is derived from existing contents and research results, whereby relevant components are extracted from current work and related to each other in an appropriate way. A prerequisite for this procedure is that there is an explicit agreement on the present subject in research, which can be explicitly captured through the framework. However, the framework does not guarantee completeness because there are the possibilities for further extensions and changes caused by future research results.

Respecting the requirements to a framework for RE by Goeken and Patas (2010), we derived the following requirements to our framework:
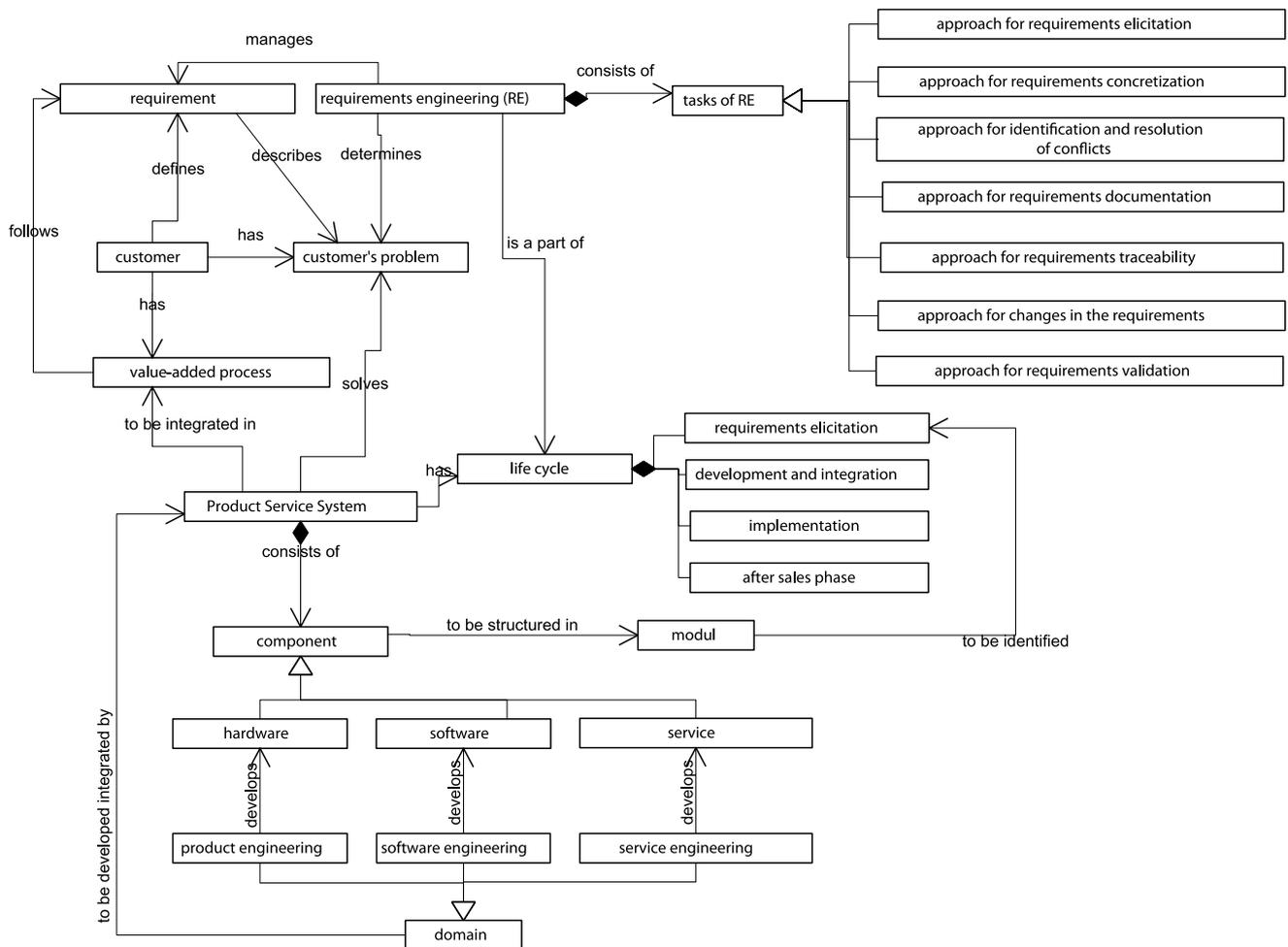
**Fig. 1** Framework for RE

(1) The framework should depict essential components of RE for PSS.
(2) The framework should reflect all relevant relations between the components.
(3) The relations between the components should be scientifically substantiated.

By meeting these requirements, the structuring of existing knowledge is facilitated. To gain this advantage, an appropriate – sufficiently high – level of abstraction is necessary that enables the representation of different research results using various terminologies and methods.

In this paper, the framework shows the relationship between the characteristics of PSS and the RE tasks in the product life cycle of PSS (see Sect. 4). As a result, it should be evident how the characteristics of PSS influence the RE. Therefore, the framework provides the basis for the establishment of the criteria considering the analysis of RE approaches. **Figure 1** illustrates this complex subject by rep-

resenting the framework for RE for PSS with UML modeling.

In the next section, the most important concepts of the framework are described. The requirements elicited and managed by the RE constitute the customer problem. As the PSS are integrated into the value creation process of the customer, corresponding requirements arise. PSS consist of different components that are developed by the appropriate domains. Moreover, the life cycle of PSS involves four stages, whereby requirements elicitation represents the first phase. The RE tasks extracted from Berkovich et al. (2010), addressed in Sect. 4, are also part of the framework.

## 5.2 Criteria for the Literature Analysis and Evaluation

The criteria for the analysis and literature evaluation described in this section are derived from the RE task areas in the de-

velopment of PSS (Sect. 4) and from the framework described in Sect. 5.1.

C1. *Provision of procedures for requirements elicitation.* In the case of PSS, the customer expresses wishes for the entire solution. Therefore, the requirements elicitation procedures must be able to identify the relevant requirements and their sources as completely as possible. As the service requirements are often not given directly by the customer, they must be defined as well (Ramaswamy 1996, pp. 14–18). Additionally, the requirements arising from the customer's value chain processes must also be elicited.

C2. *Provision of procedures for requirements concretization.* Based on the first step, the requirements are concretized and translated into the "language of the developers." They result in domain-specific design requirements to the components of the PSS that are developed by the do-

mains. The requirements must be concretized in a way that the design requirements can be allocated to the single domains. This means that the initial requirements must be concretized into design requirements for services that are expressed in three dimensions: potential, process, and result (Husen 2007) for hardware and software.

C3. *Provision of procedures for identification and resolution of conflicts between the requirements.* Frequently, the requirements are not consistent. In order to identify and resolve the conflicts between them, a procedure must be provided. This procedure must be able to identify conflicts within one domain and across different domains.

C4. *Provision of procedures for requirements documentation.* Procedures for documenting requirements as complete, consistent, and unique as possible must be provided. Also, the handling of requirements' changes is part of the documentation procedure. As the requirements documentation is the basis for the communication between all domains, the characteristics of each discipline should be considered. Therefore, the sources of requirements, the requirements derived from a requirement by concretization and the relations from a requirement to requirements of other domains must be documented. If, for instance, a requirement is concretized and assigned to the service component, the related dimension (potential, process, or result) must be recorded as well.

C5. *Provision of procedures for requirements traceability.* The progressing development or changes of the customer needs are the main reasons for changing requirements. Hence, the requirements have to be controlled and traced to get the current specification at any time in the life cycle of PSS. This includes a procedure to trace the requirements from their origin to their transmission into the discipline-specific design. The interdependencies between the requirements of one domain, as well as of different domains, should be identified and documented.

C6. *Provision of procedures for changes in the requirements.* Procedures for analyzing the impacts of changes in the

specification on the initial requirements on the domain-specific design requirements, as well as on the development results of the components, must be available. Especially important is the consideration of changes during the usage phase of PSS.

C7. *Provision of procedures for requirements validation.* Procedures must be provided for enabling the validation of the design requirements regarding their compliance with the initial requirements. For PSS, it is especially important to test the design requirements of each domain with the initial stakeholder requirements.

C8. *Integration of customers and other stakeholders into RE.* Customers and other stakeholders are the most important sources of requirements, and significantly influence the product success by their decisions. For the complete fulfillment of their claims, the customers and stakeholders are integrated into the RE process.

C9. *Support of modularization by RE.* The design and reuse of modules constitute an important characteristic of PSS. To identify reusable modules in the early stage of the development process, RE must prepare the requirements in an appropriate way.

# 6 Analysis of Approaches in Product, Software, and Service Engineering as Well as of Integrated Approaches for the Development of PSS

## 6.1 Presentation of the Approaches

To gain a better understanding of the subsequent analysis of the selected RE approaches regarding their satisfaction of the criteria defined in Sect. 5.2, they are briefly described here, structured according to the different tasks of RE. A detailed explanation is given in the Appendix 2.

### 6.1.1 Requirements Engineering in Product Engineering

As a first step in the development process, the approaches of Ulrich and Eppinger (2003), Ehrlenspiel (2002), VDI-Richtlinien 2221 (1993), as well as that of Pahl et al. (2006), state the analysis of the future development environment in order to identify possible influencing factors and to establish the overall objective of the development. Based on that, all

stakeholders and their requirements on the solution are determined.

Since the requirements of the stakeholders are often qualitative and imprecise (Tseng and Jiao 1997), they must be concretized and translated into the language of the developers, by supplementing them with detailed and quantitative information used for product development (Ahrens 2000; Pahl et al. 2006). Thereby, the requirements of the stakeholders are becoming design requirements that are implemented by the development. Although requirements concretization is one of the key elements in RE, it is not mentioned explicitly in the analyzed approaches.

Furthermore, conflicts between the requirements should be identified and resolved as soon as possible (Ehrlenspiel 2002; Ulrich and Eppinger 2003). The resolution of conflicts often leads to changes in the requirements and in realized product components (Peterson et al. 2007). For the evaluation of the impacts of changes, it is necessary to record the interdependencies of the requirements by using e.g., a DSM-matrix (Danilovic and Sandkull 2005).

Afterwards, the concretized requirements have to be validated by using for example the simulation methodology. This method enables statements about characteristics of the product in early stages of the development process (Schäppi et al. 2005).

In product engineering, modularization is presented in various forms. Its basic principle is to define modules and their interfaces so that they can be reused for different products (Ehrlenspiel 2002; Schäppi et al. 2005).

Many of the analyzed approaches comprise classic procedures following the waterfall model, such as that in Pahl et al. (2006). However in practice, iterative process models are frequently used. All activities of RE are done iteratively and should be integrated into the development process. For further information about RE approaches in product engineering, see Berkovich et al. (2009c).

### 6.1.2 Requirements Engineering in Software Engineering

In software development, RE constitutes a separate discipline. It is defined as a process of defining the relevant requirements, by identifying all stakeholders and their needs, and by documenting the requirements in the form of a specification

that can be used for communication, further analyses, and implementation (Sommerville 2004, pp. 143–144).

According to the analyzed approaches, RE includes requirements elicitation, prioritization, concretization, documentation, validation, negotiation, traceability, and change management. These tasks are seen as structural elements of RE in the framework of Pohl (1993, 39 ff).

During requirements elicitation, the requirements and their sources are identified (Lamsweerde 2009, p. 62). By communicating intensively and targeting stakeholders, a better understanding of the requirements is achieved (Coughlan and Macredie 2002).

In the next step, in requirements concretization, a bridge between the initial stakeholder requirements and the detailed design requirements is created (Kotonya and Sommerville 1998, pp. 146–149). If inconsistencies between the requirements arise, they must be discovered and resolved by finding some compromise (Cheng and Atlee 2007). In subsequent documentation, essential information about the requirements (e.g., their description, changes in requirements, or responsibilities) should be specified (Pohl 2007, pp. 547–549).

During requirements validation, the requirements are checked for ambiguity and falsity (Jönnson and Lindvall 2005; Kotonya and Sommerville 1998, pp. 87–90). To do so, prototypes are frequently used to illustrate the requirements, to express new requirements, as well as to validate the requirements particularly with regard to the fulfillment concerning the expectations of all stakeholders (Lamsweerde 2009, pp. 70–72).

The requirements can change during the entire product lifetime from their identification to product use (Cox et al. 2009). In this context, the task of change management is to capture all changes, to check them for their feasibility by determining their costs and impacts on other requirements, to prepare them for further stages of development, as well as to ensure appropriate documentation (Kotonya and Sommerville 1998, pp. 143–146).

In software engineering, the problem of creating modules and reusing them is known as software product lines. To enable a directed reuse, a domain-specific basis of applications is necessary (Käkölä and López 2006).

In addition to the RE task areas presented in this section, various process models are available that refer to certain problems, indicating in which order existing RE steps should be applied and what particular aspects should be considered.

### 6.1.3 Requirements Engineering in Service Engineering

The objective of service engineering is to enable a systematic development and design of services by providing various methods, process models and tools (Bullinger and Schreiner 2003).

Initially, essential information about service ideas, key clients, and sources of requirements, namely, the customer and the supplier, are identified (Bullinger and Schreiner 2003). Afterwards, the goals, chances, and risks of the developing services should be determined (Frietzsche and Maleri 2003).

The initial requirements are concretized by assigning them to quantifiable attributes related to the implementation of the services (Husen 2007; Ramaswamy 1996). By classifying the concretized requirements according to the three dimensions of development (potential, process, and result), the result of service provision, as well as the necessary activities and resources, can be received (Bullinger and Schreiner 2003).

The approaches of service engineering do not focus on the identification and resolution of conflicts between the requirements, change management, and requirements traceability. They mention these activities only briefly, and refer to the procedures in software engineering. For further information about RE approaches in service engineering, see Berkovich et al. (2009a).

In service development, the benefits of modularization and reuse are recognized. The reuse of undifferentiating service components leads to an increase in profitability (Böhmann et al. 2008). Böhmann and Krcmar (2003) propose an approach to developing modular services, which also includes the RE phases, albeit superficially.

### 6.1.4 Requirements Engineering of Integrated Approaches for PSS Development

The topic of RE in integrated development approaches for PSS is abstractly discussed in the literature without going into detail on the activities of RE. In the development process of PSS, the

hardware, software, and service components are developed in parallel, and coordinated by the involved disciplines (Spath and Demuß 2003). Aurich et al. (2007) propose a process model for the life cycle management of PSS. While in its first phase (organizational structuring) the organizational conditions are created in order to enable an integrated development of services and hardware/software, and the requirements are elicited afterwards (PSS planning). Then, the requirements are analyzed, categorized according to services and hardware/software, and finally realized (PSS implementation). Another model considering the entire lifetime of PSS is described in Lindahl et al. (2007). As an essential step in the development process, the task of RE is to identify customer needs in terms of goods and services; however, concrete techniques for its realization are not mentioned.

According to Spath and Demuß's (2003) "hybrid product development" approach, the development of PSS exclusively focuses on the single elements of the solution. Thereby, the requirements are elicited, analyzed, and used for the derivation of the product structures regarding material and immaterial components in the form of a requirements model that has to be updated during the entire development.

As a basis for the development process, the framework concept of Botta (2007) considers the requirements as required product properties of the PSS. Based on the required product properties, its feature structures are derived, which describe the structure of the PSS.

The approach of Böhmann et al. (2008) regarding the modularization of standardized solutions is particularly tailored to IT services, consisting of hardware, software, or classic services.

### 6.2 Criteria Based Analysis of RE Approaches

Based on the findings obtained from the literature search and the criteria previously defined, the selected RE approaches are analyzed regarding the suitability for PSS. **Table 2** summarizes the analysis results for the domain-specific approaches and the integrated approaches for the development of PSS. They are depicted according to the criteria and explained in detail subsequently.

C1: *Provision of procedures for requirements elicitation.* The elicitation of

**Table 2** Analysis of RE approaches for their suitability for PSS

| Criteria for the literature analysis | Approaches of Product engineering | Approaches of Software engineering | Approaches of Service engineering | Approaches of integrated development of PSS |
|---|:---:|:---:|:---:|:---:|
| C1: Provision of procedures for requirements elicitation | ◐ | ◐ | ◐ | ◐ |
| C2: Provision of procedures for requirements concretization | ○ | ○ | ◐ | ○ |
| C3: Provision of procedures for identification and resolution of conflicts between the requirements | ◐ | ◐ | ◐ | ○ |
| C4: Provision of procedures for requirements documentation | ● | ● | ● | ◐ |
| C5: Provision of procedures for requirements traceability | ◐ | ◐ | ◐ | ◐ |
| C6: Provision of procedures for changes in the requirements | ○ | ◐ | ○ | ◐ |
| C7: Provision of procedures for requirements validation | ◐ | ● | ◐ | ◐ |
| C8: Integration of customers and other stakeholders into RE | ○ | ◐ | ◐ | ◐ |
| C9: Support of modularization by RE | ◐ | ◐ | ◐ | ◐ |

● Completely fulfilled.   ◐ Partly fulfilled.   ○ Not fulfilled.

the requirements is addressed in all disciplines (product, software, and service engineering) and supported by different techniques such as interviews, workshops, or analyses of existing documents. However, each domain strongly focuses on the respective domain. In product engineering, for instance, there are no procedures for requirements elicitation of the product-related services that are often only expressed implicitly. On the contrary, the integrated approaches do not include concrete procedures for elicitation, but recognize their necessity. Weaknesses have been identified in the derivation of the requirements from the value-added process of the customer in the approaches of all domains. Some approaches propose an environment analysis for re-

solving this weakness, but do not describe it in detail. However, detailed instructions on the determination of relevant value-added processes, as well as on the elicitation of requirements, are missing. Further, the need of cross-domain knowledge for an integrated development is not considered. This aspect becomes particularly important for asking the customer the right questions.

C2: *Provision of procedures for requirements concretization*. The concretization of stakeholders to design requirements is a central element of the analyzed RE approaches in product and software engineering. In this connection, procedures are offered that are solely applicable in the respective domain. In contrary, the integrated approaches emphasize the

translation of the initially defined requirements, but do not provide concrete procedures. The method of QFD (Quality Function Deployment) is applied by all domains. As this method includes a comparison with existing products, its usage for the development of new products is difficult. QFD was also adapted to PSS. However, it cannot be employed for new development or for the derivation of design requirements from customer requirements. It is only applicable for evaluating possible solutions according to their fulfillment of the customer requirements (Möhrle and Spilgies 2005). Nonetheless, the translation of initial requirements to the PSS to design requirements for the single domains remains widely unsolved.

None of the procedures described in product and software engineering is suitable for the structuring of services according to the dimensions of development. Moreover, these approaches focus on the quantification of the requirements and are thus not applicable for service engineering (Husen 2007). Although the concept of requirements concretization is addressed in the approaches of service engineering, concrete procedures are missing. Considering product-related services, the identification of the primary product is also only incompletely examined.

C3: *Provision of procedures for identification and resolution of conflicts between the requirements*. The identification and resolution of conflicts are hardly discussed in the approaches of service engineering and of integrated procedure models. Only Husen (2007) proposes procedures for these activities that are based on those of product and software engineering. Procedures for the identification of conflicts known from software engineering and product engineering, such as influence matrices used for controlling interdependencies between the requirements, concentrate on the respective domain. This can lead to undiscovered conflicts between requirements of different domains. To resolve such conflicts, many approaches suggest negotiations with stakeholders as well as with the developers.

C4: *Provision of procedures for requirements documentation*. The documentation of requirements in natural language is described in all approaches. In service engineering as well as in the integrated approaches, no detailed information about creating a requirements specification is given. The advantages of documents written in natural language are the resulting simplicity and comprehensibility. Disadvantages include, in particular, ambiguities that should be considered in a cross-domain development. Since the documentation of requirements constitutes a complex process, its maintenance over the entire product lifetime is highly cost-intensive. Creating a model based requirements documentation that is widespread in software engineering is difficult to use for PSS. The reason for this is that there are no procedures and models

for the representation of the requirements on services, nor for the relationship between the requirements of different domains.

C5: *Provision of procedures for requirements traceability*. The interdependencies between the requirements and the product components, as well as the traceability of one requirement to its origin, are captured by influence or link matrices in product and software engineering. In service engineering as well as in the integrated approaches, no concrete details for guaranteeing traceability are provided. The procedures for the identification of interdependencies between the requirements do not consider the participation of different domains. Therefore, the influence and link matrices have to include all domains of the PSS as well as the three dimensions of service development in a multidimensional way. All approaches indicate that the specification should contain corresponding references to: dependent requirements and components, sources of the requirements, as well as changes in the requirements.

C6: *Provision of procedures for changes in the requirements*. The issue of changes in the requirements is neither discussed extensively in product engineering nor in service engineering. The approaches of service engineering, as well as the integrated approaches, point out that the requirements can change, but do not provide further information. A reference is made solely to the procedures of software engineering. None of the domains indicates the importance of change management during the utilization of the product. To estimate the impacts of changes on further requirements and components, the information gained from requirements traceability is used. All analyzed approaches emphasize that changes can only be realized if they are accepted by all people participating in the development process, and if they do not cause any conflicts.

C7: *Provision of procedures for requirements validation*. Most approaches of service engineering describe the validation of the requirements as a comparison between the service concept based on the specification and the initial requirements. Thereby, no specific steps towards requirements

validation are given. Also, in product engineering and in the integrated approaches, the validation is not discussed in detail. Techniques, such as inspections and reviews, are merely suggested in software engineering, where they are used for detecting errors and inconsistencies in the specification. These techniques are also suitable for PSS and product-related services (Husen 2007). In this context, it should be noted that all involved domains have to be integrated into requirements validation. Moreover, the three dimensions of service development, as well as the distinction between redundant and related requirements, must be considered.

C8: *Integration of customers and other stakeholders into RE*. All domains emphasize the role of the customer who decides on the success of a product. In service engineering, the customer constitutes the recipient of the solution and should therefore be integrated into the development process. In most approaches the integration is restricted to the phases of requirements elicitation and agreement. However, requirements prioritization is also mainly determined by the customer and should therefore be taken into account. As the customer frequently has a different understanding of the product to be developed, the communication and interpretation of his wishes and expectations are difficult, particularly for developers.

C9: *Support of modularization by RE*. The advantages of the modularization and reuse of modules are recognized in all disciplines. Additionally, preliminary works towards this topic have been discovered in several integrated approaches. With regard to RE, the theme of modularization is not presented in detail, since it is basically dealt with in the subsequent stages of the development process.

## 7 Discussion of the Results

In Sect. 6, the different approaches were reviewed concerning their suitability for PSS. **Table 3** summarizes the evaluation and provides an overview of the gaps in the literature concerning RE for PSS. For each criterion, the mean value of the four domains was calculated and assigned to

**Table 3** Results of the analysis of the RE approaches on the basis of the defined criteria

| Criterion | Fulfillment | Description |
|---|---|---|
| C1: Provision of procedures for requirements elicitation | ◑ | The elicitation of the requirements is well elaborated in all domains and in the integrated approaches. However, the single techniques can solely be used for the respective domain. Requirements elicitation across all domains, which also includes the implicit requirements on the service component, is discussed insufficiently. In this area, special procedures for PSS are needed. |
| C2: Provision of procedures for requirements concretization | ○ | Despite the provision of numerous approaches and methods for requirements concretization, integrated procedures suitable for PSS are missing. For this reason, it can be stated that there is no support of the translation of the initially defined requirements into the domain-specific design requirements. Furthermore, the interplay between the requirements of different domains is not addressed. |
| C3: Provision of procedures for identification and resolution of conflicts between the requirements | ◑ | The existing approaches only imply the resolution of conflicts by negotiations with the customers and developers. Conflicts between the different domains are not considered. |
| C4: Provision of procedures for requirements documentation | ● | The approaches examined in this paper mainly propose to document the requirements in natural language. This kind of documentation can also be used for PSS. |
| C5: Provision of procedures for requirements traceability | ◑ | In product and software engineering, procedures for requirements traceability are described. However, these procedures do not play any important role in service engineering and in the integrated approaches. The traceability of requirements across all domains, as needed for PSS, is not discussed. |
| C6: Provision of procedures for changes in the requirements | ○ | In all approaches, changes in the requirements are not sufficiently taken into account. Therefore, none of them is suitable for PSS. |
| C7: Provision of procedures for requirements validation | ◑ | For requirements validation, sophisticated procedures are introduced in software engineering. By adjusting them accordingly to the conditions for PSS, they can be reused. |
| C8: Integration of customers and other stakeholders into RE | ◑ | The essential role of the customer in the development process is recognized by all disciplines and by the integrated approaches. In the majority of cases, however, the integration is unstructured and only takes place in the first phases. Consequently, new procedures for PSS are necessary. |
| C9: Support of modularization by RE | ◑ | The issue of modularization is discussed in all approaches. Considering RE, no detailed information exists yet and has to be elaborated for PSS. |

● Completely fulfilled.  ◑ Partly fulfilled.  ○ Not fulfilled.

one degree of fulfillment (completely fulfilled, partly fulfilled, or not fulfilled). If an unambiguous assignment was not possible, one of the three values was chosen and justified by the description of the analysis.

## 8 Summary and Conclusion

PSS represent a new trend on the market for hardware, software, or service providers. As they consist of various components that are produced by different domains, their development constitutes a major challenge. For successful development, RE becomes a key factor, and its role for PSS is thus investigated in this paper. Based on the characteristics and the different life cycle stages of PSS, the criteria for an effective RE are derived. Within the literature review, 15 leading RE process models and approaches of product, software, and service engineering, as well as five integrated approaches of PSS development, are analyzed for their compliance with the predetermined criteria. Thus, deficits in the existing approaches can be evaluated and additional research needs can be identified.

In this paper, the criteria for a successful RE are derived from the analysis of the characteristics and life cycle of PSS. In this context, a key feature of PSS constitutes the orientation towards the customer/individualization of solutions. Therefore, the requirements arising from the customer's value creation process must be analyzed. Importantly, the stakeholders of the various domains have to be considered. Thus, the communication between the domains, as well as the

consideration of service engineering, are essential parts of the entire RE process.

The analysis of the existing RE approaches for PSS shows that the necessity of a systematical RE is realized in all domains. However, the maturity level of the RE approaches varies widely in the single domains. In service engineering and in the integrated approaches, approaches for requirements management are hardly suggested or not tested. The concept of RE is developed furthest in product and software engineering, but only focuses on single domains. Approaches of cross-domain cooperation and communication are not mentioned explicitly in the analyzed approaches that imply approaches tailored to the respective fields. This impedes the cooperation between the domains, i.e., the use of the results of one domain in another. The different terminologies even obscure the view on possible commonalities. A further deficit constitutes the cross-linking of the RE approaches of the different domains. Despite conceptual and methodical analogies, the various terminologies complicate the cooperation.

In order to meet these challenges, sophisticated development techniques in the single domains can be adapted to PSS. This imposes a first step to cross-domain integration. Future research could target a comprehensive approach of RE that incorporates different perspectives on, and characteristics of, the domains as well as the attributes of the PSS. Similarities of the existing approaches have to be identified and summarized. The core concepts of such an approach could include a cross-domain cooperation, the integration of RE in the development process, and the handling of the different life cycles of the components. Consequently, further scientific research should focus on the creation of an artifact model and defining a structure for the documented requirements. On the basis of a common artifact model for all domains, the coordination of single approaches is possible.

## References

Ahrens G (2000) Das Erfassen und Handhaben von Produktanforderungen – methodische Voraussetzungen und Anwendung in der Praxis. Dissertation, Technische Universität Berlin

Aurich JC, Schweitzer E, Fuchs C (2007) Life cycle management of industrial product-service systems. In: Takata S, Umeda Y (eds) Advances in life cycle engineering for sustainable manufacturing businesses. Springer, London, pp 171–176

Aurum A, Wohlin C (2005) Requirements engineering: setting the context. In: Aurum A, Wohlin C (eds) Engineering and managing software requirements. Springer, Berlin, pp 1–15

Becker J, Beverungen D, Knackstedt R (2008) Reference Models and modeling languages for product-service systems – status-quo and perspectives for further research. In: Proc 41st annual Hawaii international conference on system sciences, Hawaii

Berkovich M, Esch S, Leimeister JM, Krcmar H (2009a) Requirements engineering for hybrid products as bundles of hardware, software and service elements – a literature review. In: Internationale Tagung Wirtschaftsinformatik (WI), Wien

Berkovich M, Leimeister JM, Krcmar H (2009b) An empirical exploration of requirements engineering for hybrid products. In: Proc XVIIth European conference on information systems (ECIS), Verona

Berkovich M, Leimeister JM, Krcmar H (2009c) Suitability of product development methods for hybrid products as bundles of classic products, software and service elements. In: Proc ASME 2009 international design engineering technical conferences & computers and information in engineering conference IDETC/CIE, San Diego

Berkovich M, Leimeister JM, Krcmar H (2010) Ein Bezugsrahmen für Requirements Engineering hybrider Produkte. In: Multikonferenz Wirtschaftsinformatik (MKWI, 2010), Göttingen

Beverungen D, Knackstedt R, Müller O (2008) Entwicklung Serviceorientierter Architekturen zur Integration von Produktion und Dienstleistung – Eine Konzeptionsmethode und ihre Anwendung am Beispiel des Recyclings elektronischer Geräte. WIRTSCHAFTSINFORMATIK 50(3):220–234

Beverungen D, Knackstedt R, Hatfield S, Biege S, Bollhöfer E, Krug C, Wienhold D, Müller P, Stelzer C, Köbler F, Blinn N (2009) Hybride Wertschöpfung – Integration von Produktion und Dienstleistung. In: Deutsches Institut für Normung e. V. (ed) PAS 1094. Beuth, Berlin

Böhmann T, Krcmar H (2003) Modulare Servicearchitekturen. In: Bullinger H-J, Scheer A-W (eds) Service Engineering – Entwicklung und Gestaltung innovativer Dienstleistungen. Springer, Berlin, pp 376–401

Böhmann T, Krcmar H (2007) Hybride Produkte: Merkmale und Herausforderungen. In: Bruhn M, Stauss B (eds) Wertschöpfungsprozesse bei Dienstleistungen: Forum Dienstleistungsmanagement. Gabler, Wiesbaden, pp 240–255

Böhmann T, Langer P, Schermann M (2008) Systematische Überführung von kundenspezifischen IT-Lösungen in integrierte Produkt-Dienstleistungsbausteine mit der SCORE-Methode. WIRTSCHAFTSINFORMATIK 50(3):196–207

Botta C (2007) Rahmenkonzept zur Entwicklung von Product-Service Systems. Eul

Brereton P, Kitchenham BA, Budgen D, Turner M, Khahil M (2007) Lessons from applying the systematic literature review process within the software engineering do-

**Abstract**

**Marina Berkovich, Jan Marco Leimeister, Helmut Krcmar**

## Requirements Engineering for Product Service Systems

### A State of the Art Analysis

In recent years, manufacturing companies and service providers have moved towards offering customer-specific problem solutions. These integrated bundles usually consist of hardware, software, and service components and are called product service systems (PSS) or hybrid products. Since the success of the resulting solution depends on the understanding of all requirements, requirements engineering (RE) has become a key factor. The article analyzes the state of the art of RE for PSS based on an extensive literature review in the domains of product-, software-, and service engineering. For this, criteria are derived from the characteristics of PSS and from the task area of RE in the life cycle of PSS. Based on these criteria we analyze the most established RE approaches for their suitability for PSS. An important finding is that integrated/interdisciplinary approaches for RE are missing. Moreover, the maturity of RE approaches in the three domains varies significantly. All analyzed approaches heavily rely on concepts and solution characteristics of their own domain so that a transfer to other domains is hardly possible. This literature review lays the foundation for successful RE for PSS and especially for future research aiming at combining and integrating RE approaches and models of product-, software-, and service engineering. Such requirement models could connect concepts of single domains and enable an integrated and seamless RE for PSS.

**Keywords:** Requirements engineering, Product service system, Hybrid product, State of the art, Product engineering, Software engineering, Service engineering

main. Journal of Systems and Software 4(80):571–583

Budgen D, Brereton P (2006) Performing systematic literature reviews in software engineering. In: Proc 28th international conference on software engineering, Shanghai, pp 1051–1052

Bullinger H-J, Schreiner P (2003) Service Engineering: Ein Rahmenkonzept für die systematische Entwicklung von Dienstleistungen. In: Bullinger H-J, Scheer A-W (eds) Service Engineering – Entwicklung und Gestaltung innovativer Dienstleistungen. Springer, Berlin, pp 53–84

Byrd TA, Cossick KL, Zmud RW (1992) A synthesis of research on requirements analysis and knowledge acquisition techniques. Management Information Systems Quarterly 16(1):117–138

Cheng BHC, Atlee JM (2007) Research directions in requirements engineering. In: Proc future of software engineering (FOSE'07), Minneapolis

Coughlan J, Macredie RD (2002) Effective communication in requirements elicitation: a comparison of methodologies. Requirements Engineering 7(2):47–60

Cox K, Niazi M, Verner J (2009) Empirical study of Sommerville and Sawyer's requirements engineering practices. IET Software 3(5):339–355

Danilovic M, Sandkull B (2005) The use of dependence structure matrix and domain mapping matrix in managing uncertainty in multiple project situations. International Journal of Project Management 23(3):193–203

Davies A (2004) Moving base into high-value integrated solutions: a value stream approach. Industrial and Corporate Change 13(5):727–756

Edvardsson B, Olsson J (1996) Key concepts for new service development. Service Industries Journal 16(2):140–164

Ehrlenspiel K (2002) Integrierte Produktentwicklung. Hanser

Frietzsche U, Maleri R (2003) Dienstleistungsproduktion. In: Bullinger H-J, Scheer A-W (eds) Service Engineering – Entwicklung und Gestaltung innovativer Dienstleistungen. Springer, Berlin, pp 195–225

Galbraith JR (2002) Organizing to deliver solutions. Organizational Dynamics 31(2):194–207

Goeken M, Patas J (2010) Evidence-based structuring and evaluation of empirical research in requirements engineering. Business & Information Systems Engineering 2(3):175–185

Gräßle M, Dollmann T (2010) Vorgehensmodelle des Product-Service Systems Engineering. In: Thomas O, Loos P, Nüttgens M (eds) Hybride Wertschöpfung. Springer, Berlin, pp 82–129

Grohmann W (2007) Von der Software zum Service. ASP – Software on Demand –

Software-as-a-Service – Neue Formen der Software-Nutzung. H. K. P. Consulting

Hickey AM, Davis AM (2003) Elicitation technique selection: how do experts do it. In: Proc 11th IEEE international requirements engineering conference, Monterey Bay

Hull E, Jackson K, Dick J (2004) Requirements engineering. Springer, London

Husen Cv (2007) Anforderungsanalyse für produktbegleitende Dienstleistungen. Universität Stuttgart, Stuttgart

Jacsó P (2008) Google Scholar revisited. Online Information Review 32(1):102–114

Jönsson P, Lindvall M (2005) Impact analysis. In: Aurum A, Wohlin C (eds) Engineering and managing software requirements. Springer, Berlin, pp 121–146

Käkölä T, López JCD (2006) Software product lines. Research issues in engineering and management. Springer, Berlin

Kitchenham BA, Brereton OP, Budgen D, Turner M, Bailey J, Linkman S (2009) Systematic literature reviews in software engineering – a systematic literature review. Information and Software Technology 51(1):7–15

Kotonya G, Sommerville I (1998) Requirements engineering: processes and techniques. Wiley, New York

Lamsweerde Av (2009) Requirements engineering: from system goals to UML models to software specifications. Wiley, New York

Langer P, Köbler F, Berkovich M, Weyde F, Leimeister JM, Krcmar H (2010) Vorgehensmodelle für die Entwicklung hybrider Produkte – eine Vergleichsanalyse. In: Multikonferenz Wirtschaftsinformatik (MKWI 2010), Göttingen

Leimeister JM, Glauner C (2008) Hybride Produkte – Einordnung und Herausforderungen für die Wirtschaftsinformatik. WIRTSCHAFTSINFORMATIK 50(3):248–251

Lindahl M, Sundin E, Sakao T, Shimomura Y (2007) Integrated product and service engineering versus design for environment – a comparison and evaluation of advantages and disadvantages. In: Takata S, Umeda Y (eds) Advances in life cycle engineering for sustainable manufacturing businesses. Springer, London, pp 137–142

Meier JJ, Conkling TW (2008) Google Scholar's coverage of the engineering literature: an empirical study. The Journal of Academic Librarianship 34(3):196–201

Möhrle MG, Spilgies W-D (2005) QFD für product service systems. Industrie Management 21(3):9–12

Pahl G, Beitz W, Feldhusen J, Grote K-H (2006) Engineering design: a systematic approach. Springer, Berlin

Peterson C, Paasch RK, Ge P, Dietterich TG (2007) Product innovation for interdisciplinary design under changing requirements. In: Proc international conference on engineering design (ICED'07), Paris

Pohl K (1993) The three dimensions of requirements engineering. In: Proc 5th inter-

national conference on advanced information systems engineering (CAiSE'93), Paris

Pohl K (2007) Requirements Engineering. Grundlagen, Prinzipien, Techniken. Dpunkt

Ramaswamy R (1996) Design and management service processes: keeping customers for life. Addison-Wesley, Reading

Sawhney M (2006) Going beyond the product: defining, designing and devlivering customer solutions. In: Lusch RF, Vargo SL (eds) The service-dominant logic of marketing. Sharpe, New York, pp 365–380

Schäppi B, Andreasen MM, Kirchgeorg M, Radermacher F-J (2005) Handbuch Produktentwicklung. Hanser

Sommerville I (2004) Software engineering. Pearson, Boston

Spath D, Demuß L (2003) Entwicklung hybrider Produkte – Gestaltung materieller und immaterieller Leistungsbündel. In: Bullinger H-J, Scheer A-W (eds) Service Engineering – Entwicklung und Gestaltung innovativer Dienstleistungen. Springer, Berlin, pp 463–502

Sturm F, Bading A (2008) Investitionsgüterhersteller als Anbieter industrieller Lösungen – Bestandsaufnahme des Wandels anhand einer Umfrage. WIRTSCHAFTSINFORMATIK 50(3):174–186

Torraco RJ (2005) Writing integrative literature reviews: guidelines and examples. Human Resource Development Review 4(3):356–367

Tseng MM, Jiao J (1997) A variant approach to product definition by recognizing functional requirement patterns. Computers & Industrial Engineering 33(3-4):629–633

Tukker A (2004) Eight types of product-service system: eight ways to sustainability? Experiences from suspronet. Business Strategy and the Environment 13:246–260

Tuli R, Kohli A, Bharadwaj S (2007) Rethinking customer solutions: from product bundles to relational processes. Journal of Marketing 71(3):1–17

Ulrich K, Eppinger S (2003) Product design and development. McGraw-Hill, New York

VDI-Richtlinien (1993) VDI 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte

Vogel O, Arnold I, Chughtai A, Ihler E, Kehrer T, Mehlig U, Zdun U (2009) Software-Architektur, 2nd edn. Spektrum

Webster J, Watson RT (2002) Analyzing the past to prepare for the future: writing a literature review. Management Information Systems Quarterly 26(2):xiii–xxiii

Wieringa R, Maiden N, Mead N, Rolland C (2006) Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. Requirements Engineering 11(1):102–107

Zellner G (2008) Gestaltung hybrider Wertschöpfung mittels Architekturen – Analyse am Beispiel des Business Engineering. WIRTSCHAFTSINFORMATIK 50(3):187–195

# Requirements Engineering for Product Service Systems

## A State of the Art Analysis

**Marina Berkovich, Jan Marco Leimeister, Helmut Krcmar**

## Appendix 1 – Research Methodology

This section completes Sect. 2, by explaining the research-methodical proceeding in detail.

### 1.1  Definition of the research question

Initially, the research questions were defined that needed to be answered in the literature analysis. The central question is: "To what extent are the selected domain-specific approaches of requirements engineering suited for PSS, and which of these can be used in the development process of PSS?"

### 1.2  Definition and validation of the analysis protocol

The analysis protocol comprised the criteria for evaluating the literature. In the literature analysis the criteria facilitated the answering of the research questions by describing the relevant characteristics for the suitability of an RE approach for PSS in detail, and by providing a basis for evaluating the adequacy of the approaches.

For the derivation of the criteria, a framework was developed, the aim of which was to describe RE concepts in a general way so that older and future research could be structured (Goeken and Patas 2010). The framework (Sect. 5.1) connected the life cycle and characteristics of PSS with the main tasks of RE. Based on the gained results, the criteria for the literature analysis were derived (Sect. 5.2).

In order to validate the analysis protocol, a test run with the "procedure model of Munich" (Lindemann 2006) known from earlier research (see Berkovich et al. 2009a) was performed. The criteria were adjusted accordingly, by focusing on the PSS-specific aspects (e.g., consideration of the requirements and their affiliation to different domains in the search for conflicts). Subsequently, the analysis protocol was reviewed for correctness and completeness.

### 1.3  Identification and selection of relevant scientific sources

In this step, all sources relevant for the literature analysis were identified. Due to the different understanding of RE in product, software, and service engineering, the literature search was performed for each domain.

*Literature search.* For searching the relevant literature, the portal "Google Scholar" was chosen, covering a major part of the journals and electronic publications in computer science as well as in engineering. Meier and Conkling (2008) have shown that 90% of the publications in engineering, published after 1990, are announced in "Google Scholar." In the field of computer science, important publishing companies (e.g., Elsevier) and non-profit organizations (e.g., ACM or IEEE) are indexed in Google Scholar (Meier and Conkling 2008; Jacsó 2008). The keywords that were used are introduced in the following paragraph.

- In product engineering, we searched for "product engineering" and "product development." Then, the keywords "requirements engineering" and "management" were added in order to receive specific results of RE.

- Since in software engineering RE is regarded as a discipline, the sole keywords for this domain were "requirements engineering," "requirements management," and "Anforderungsmanagement."

- The results of the search in service engineering, completed with the terms "service engineering" and "service development," were subsequently combined with "requirements engineering" and "requirements management".

- In addition to individual concepts, other approaches covering the issue of an integrated development of PSS were also considered. To get relevant literature in this area, the search terms "hybrid products" and "product service system" were entered in "Google Scholar".

*Selection of relevant literature.* All hits of the search portal were sorted by the frequency of their citations. For each keyword, the 100 most cited articles were examined and assigned to one of the following categories.

(a) Articles that constituted a generic approach/procedure model:

The generality and spread of the respective article comprised the selection criteria for the first category (see Goeken and Patas 2010). Further, only those sources were considered that made a major contribution to RE, that is, they dedicated at least one subject area (e.g., a book chapter or a section of an extensive journal article) to RE. By selecting generic approaches, a superordinate structure of knowledge and a meta-level were established for the classification of detailed approaches appearing in conferences and journals (Goeken and Patas 2010). The assignment of the specific publications to generic approaches (mostly in the form of books) thus gave an overview of the RE in product, software, and service engineering.

For each domain, five generic approaches were presented that were based on expert opinions as well as on appraisals of the authors. In order to provide a high degree of validity, the selection of the articles was coordinated by all authors. Since the generic approaches showed certain similarities, it was sufficient to analyze five of them for each domain (see Table 1).

(b) Articles that implied a specific topic of RE:

This category included journals and conference articles, addressing specific themes of RE, such as activities and techniques also proposed in the generic approaches. Due to the resulting large number, a detailed presentation could not be included in

this paper'. However, some articles are cited in section 6.1 and section 10 in the description of the RE approaches.

The publications found in the conferences and journals were assigned to the different domains: 24 referred to product engineering, 47 to software engineering, 14 to service engineering, and 14 to an integrated development of PSS.

(c) Articles that did not consider RE:

Papers that did not primarily deal with the theme of RE were depicted in this category. Due to the full text search of the portal, articles that solely mentioned RE in the text (e.g., in the introduction) were used.

### 1.4 Data collection and analysis

The data needed for answering the research questions were extracted and analyzed according to the criteria defined in the protocol. In order to ensure the quality of the literature analysis, the following questions obtained from Kitchenham et al. (2009) were answered: Do the sources cover all relevant aspects of the research questions Is the quality of the analyzed articles sufficient? Is the information required for the literature analysis described adequately in the article?

**Appendix 2 – Description of approaches in product, software, and service engineering as well as of integrated approaches for the development of PSS**

To gain a better understanding of the subsequent analysis, the selected RE approaches were examined on the basis of the criteria defined in Sect. 5.2 and divided into the different tasks of RE.

**2.1 Requirements engineering in product engineering**

According to Ehrlenspiel (2002, p. 663), a requirement is qualitative and/or quantitative information about the attributes of, or conditions for, a product.

*Requirements Elicitation.* As a first step in the development process, the approaches of Ulrich and Eppinger (2003), Ehrlenspiel (2002), VDI-Richtlinie 2221 (1993), as well as Pahl et al. (2006) analyze the future development environment in order to identify possible influencing factors in order to establish the overall objective of the development. On this basis, all stakeholders and their requirements to the solution are determined. Subsequently, the resulting requirements have to be estimated (e.g., according to their relevance for the customer) and concretized. In his case study, Hobday (1998) points out that during the development of PSS, requirements elicitation is especially challenging, as customers cannot consciously express their wishes regarding the product.

*Requirements Concretization and documentation.* Although requirements concretization is one of the most important tasks of RE, it is not described explicitly and consistently in the analyzed approaches. Since the requirements of the stakeholders are often qualitative and imprecise (Tseng and Jiao 1997), they must be translated into the "language of the developers" by receiving detailed and quantitative information used for product development (Ahrens 2000; Pahl et al. 2006). To connect customer and design requirements (Chahadi et al. 2007), the method of "Quality Function Deployment" (QFD) is frequently quoted (Hauser and Clausing 1988). Its main goal is to evaluate the different possible solutions according to the satisfaction of customer requirements, but not to focus on the systematic derivation of the design requirements from the customer requirements (Ahrens 2000; Pahl et al. 2006). For this reason, the initial and concretized requirements should be listed in a document and adapted to changes. In practice, large and unstructured documents, containing requirements to different subsystems and dependent systems represent a serious problem (Almefelt et al. 2006).

*Identification and resolution of conflicts between the requirements.* If the requirements are not consistent, conflicts arise (Almefelt et al. 2006) that should be identified and resolved as soon as possible (Ehrlenspiel 2002; Ulrich and Eppinger 2003).

*Requirements traceability and change management.* Changes in one requirement can lead to further changes in other requirements and in the corresponding components of the product

(Peterson et al. 2007). For this reason, it is necessary to record their interdependencies by using, e.g., a DSM-matrix (Danilovic and Sandkull 2005) that allows a fast evaluation of impacts of changes. Moreover, the specification has to be updated in case of any changes during the entire development process to facilitate requirements traceability. Jiao and Chen (2006) conclude their literature analysis with the statement that a lot of researchers have lately investigated this topic. In practice, however, continuous requirements engineering is still a great challenge.

*Requirements validation.* Modeling and simulation become more and more important in the development process (Schäppi et al. 2005). The simulation enables early statements of the product features and their fulfillment. Additionally, new knowledge is gained by comparing the different alternatives (Pahl et al. 2006). Consequently, gaps in the requirements can be elicited or impacts of certain requirements can be illustrated for stakeholders.

In product engineering, *modularization* is presented in various forms. Its basic principle is to define modules and their interfaces so that they can be reused for different products (Ehrlenspiel 2002; Schäppi et al. 2005). Thus, new products are based on existing guidelines, such as standard designs, reference architectures or product families that have to be discovered in requirements elicitation. Schäppi et al. (2005, p. 328) propose a method for describing products of a product family. The product family proposes a set of variable characteristics that can be combined individually for each product. At this point, the connection to the RE is visible, but the authors only indicate it.

Many of the analyzed approaches comprise classic procedures following the waterfall model such as, e.g., Pahl et al. (2006). However in practice, iterative procedure models are frequently used. Peterson et al. (2007) purport that iterative procedure models are indispensable, as the requirements in innovative projects change permanently. Nevertheless, the waterfall models depict typical phases in product development that imply the same tasks, regardless of the concrete process design.

All activities of RE are passed iteratively and should be integrated into the development process. For further information about RE approaches in product engineering, see Berkovich et al. (2009c).

## 2.2 Requirements engineering in software engineering

In software development, requirements engineering constitute a separate discipline. It is described as a process of defining the relevant requirements, by identifying all stakeholders and their needs, and by documenting the requirements in form of a specification that can be used for communication, further analyses, and implementation (Sommerville 2004, pp. 143-144). In this regard, requirements management can be seen as a sub-area of RE (Pohl 2007, p. 493).

According to the analyzed approaches, RE includes requirements elicitation, requirements prioritization, requirements concretization, requirements documentation, requirements validation, requirements negotiation, change management, and requirements traceability. These tasks are seen as essential elements of RE in the framework of Pohl (1993, pp. 39 ff).

*Requirements elicitation.* During requirements elicitation, the requirements and their sources (Lamsweerde 2009, p. 62; Aurum and Wohlin 2005), such as customers, stakeholders, existing documents, laws and competitors, are identified (Robertson and Robertson 2007, pp. 45-54). By communicating intensively and targeted at stakeholders, a better understanding of the requirements is achieved (Coughlan and Macredie 2002; Al-Rawas and Easterbrook 1996).

*Requirements concretization.* In this step, a bridge between the initial stakeholder requirements and the detailed design requirements is created (Kotonya and Sommerville 1998, pp. 146-149). By checking the consistency of the requirements, a full description comprising qualitative and quantitative aspects is written afterwards (Hull et al. 2004). According to Jarke (1998), various models serve as a conceptual foundation for requirements concretization. In this context, the identification of the relations between the requirements and their interdependencies should be mapped (Nuseibeh and Easterbrook 2000). Thus, it is important to know the priorities of the requirements influenced by the customer needs, the costs, and the risk level. It is then possible to determinate the importance of the requirements in the implementation phase (Hull et al. 2004, p. 81), as well as to estimate the customer satisfaction (Berander and Andrews 2005).

*Requirements documentation.* In requirements documentation, essential information about the requirements (e.g., their description, changes in requirements or responsibilities) should be specified (Pohl 2007, pp. 547-549). The most common form of documentation is the natural language, which has the disadvantage of being ambiguous. According to IEEE (1998), the specification should be: correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable, and traceable.

*Identification and resolution of conflicts between the requirements.* The aim of this activity is to discover and resolve all conflicts between the requirements by finding some compromise (Lamsweerde 2009, p. 14; Cheng and Atlee 2007).

*Requirements validation.* In this phase, the requirements are checked for inconsistencies, ambiguity and falsity (Jönnson and Lindvall 2005; Kotonya and Sommerville 1998, pp. 87-90). Furthermore, the design requirements are validated for their compliance with the initial requirements by involving all stakeholders (Al-rawas and Easterbrook 1996). In this connection, prototypes are frequently used to simulate and demonstrate different aspects of the developing system (Lamsweerde 2009, pp. 70-72). Prototypes show the effects of the require-

ments and thus contribute to the extraction of new stakeholder requirements as well as to the validation of requirements, particularly with regard to their fulfillment concerning the expectations of the stakeholders (Pohl 2007, pp. 457-462).

*Requirements traceability and change management.* The task of change management is to identify changes in the requirements during the entire product life cycle (Cox et al. 2009), to check them for their feasibility by determining their costs and impacts on other requirements, as well as to modify the specification (Kotonya and Sommerville 1998, pp. 143-146). A major precondition for this constitutes requirements traceability that ensures the mapping of all relationships between the requirements, from the initial requirements to the final system properties (Kotonya and Sommerville 1998, p. 160). Therefore, it supports the detection of existing conflicts between requirements (Lucia et al. 2007).

In software engineering, the reuse of modules is known as software product lines. On the basis of a common platform, various domain-specific applications can be created (Käkölä and López 2006). In their scientific work, Käkölä and López (2006, pp. 130 ff) provide an overview of the methods relating to requirements engineering for product lines. Additionally, they propose their own approach that is rested on an artifact model of RE specification results, which gives a description and definition of the steps in the process of RE.

In addition to the RE tasks presented in this section, various procedure models are available that refer to certain problems, indicating in what order existing RE steps should be applied, and what particular aspects should be considered.

## 2.3 Requirements engineering in service engineering

Services are defined as intangible assets produced for satisfying customer needs by using external production factors (Frietzsche and Maleri 2003, p. 197). The objective of service engineering is to enable a systematic development and design of services by providing various methods, procedure models, and tools (Bullinger and Schreiner 2003). In the literature, several approaches of this discipline, also known as "service development," can be found.

*Requirements elicitation.* Initially, essential information about service ideas, key clients, and sources of requirements, namely, the customer and the supplier, is identified (Bullinger and Schreiner 2003). For product-related services, the primary product is determined and extended by different services (Husen 2007). Afterwards, the goals, chances and risks of the developing services should be recognized (Husen 2007; Frietzsche and Maleri 2003). Further, the customer requirements on the services are elicited with the help of interviews, observations, market analyses, and investigations (Husen 2007; Ramaswamy 1996), and are differentiated between wishes and needs (Edvardsson and Olsson 1996).

*Requirements concretization and documentation.* The initial requirements are prioritized (Ramaswamy 1996) and checked for interdependencies (Husen 2007). Based on the result, the requirements are concretized by assigning them to quantifiable attributes related to the implementation of the services (Husen 2007; Ramaswamy 1996). For this process, the method of QFD is frequently applied (Husen 2007). By classifying the concretized requirements according to the three dimensions of development (potential, process, and result), the result of service provision, as well as the necessary activities and resources, can be identified (Frietzsche and Maleri 2003; Bullinger and Schreiner 2003). This classification underlies the constructional characteristics of services that are defined as the achievement potential of the provider, the immateriality of the service, and the customer integration into the development process. Having completed the task of concretization, the requirements are documented afterwards (Bullinger and Schreiner 2003).

The approaches of service engineering do not focus on the identification and resolution of conflicts between the requirements, change management, and requirements traceability. They mention these activities only briefly and refer to the procedures in software engineering. For further information about RE approaches in service engineering, see Berkovich et al. (2009a).

In service development, the benefits of modularization are recognized. The reuse of undifferentiated service components leads to an increase in profitability (Böhmann et al. 2008). Böhmann and Krcmar (2003) propose an approach to develop modular services, which also includes the RE phases, but only superficially.

## 2.4  Requirements engineering of integrated approaches for PSS development

The theme of RE with regard to an integrated development of PSS is abstractly discussed in several approaches without going into detail on the activities of RE. Therefore, the following section is not structured according to the different RE tasks.

In the development process of PSS, the hardware, software, and service components are produced in parallel and coordinated by the involved disciplines (Korell and Ganz 2000; Spath and Demuß 2003).

Aurich et al. (2007) propose a procedure model for the life cycle management of PSS. While in its first phase (organizational structures) the organizational conditions are created in order to enable an integrated development, and the requirements are elicited afterwards (PSS planning). Based on that, the ideas for the PSS are generated, analyzed, categorized, and finally realized (PSS implementation). Another model considering the entire lifetime of PSS is described in Lindahl et al. (2007). As an essential step in the development process, the task of RE is to identify the customer needs in terms of goods and services. However, concrete techniques for its realization are not mentioned.

9

The "hyprid product development" of Spath and Demuß (2003) focuses mainly on the single elements of the whole solution. Initially, the requirements are elicited, analyzed, and used for the derivation of the product structures regarding material and immaterial components. Following this, the product structures are assigned to certain functions and principles that are subsequently implemented by the relating domain. During the entire development, the requirements model has to be updated.

As a basis for the development process, the "framework for the development of PSS" of Botta (2007) considers the requirements with respect to the required product properties. To describe the PSS accurately, its feature structures are derived.

Another concept is delineated in the development process of Thomas et al. (2008) that underlies the methods of product engineering. The first step includes the definition of the requirements to the PSS in order to determine its desired features and characteristics by using QFD. The article concludes that the methods of product engineering are only slightly suited for PSS.

The approach of Böhmann et al. (2008) to the modularization of standardized solutions is particularly tailored to IT services, consisting of hardware, software, or classic services. The requirements to the final product are classified based on four different views. Following this, the standardized partial solutions are identified.

**References**

Al-Rawas A, Easterbrook S (1996) Communication problems in requirements engineering: a field study. In: First Westminster conference on professional awareness in software engineering. London

Almefelt L, Berglund F, Nilsson P, Malmqvist J (2006) Requirements management in practice: findings from an empirical study in the automotive industry. Research in Engineering Design 17(3):113-134

Berander P, Andrews A (2005) Requirements prioritization. In: Aurum A, Wohlin C (eds) Engineering and managing software requirements. Springer, Berlin, pp 69-94

Chahadi Y, Wäldele M, Birkhofer H (2007) QFD – a link between customer requirements and product properties. In: International conference on engineering design, ICED'07. Paris

Hobday M (1998) Product complexity, innovation and industrial organisation. Research Policy 26(6):689-710

Hauser JR, Clausing D (1988) The House of quality. Harvard Business Review 66(3):63-73

IEEE (1998) IEEE recommended practice for software requirements specifications

Jarke M (1998) Anforderungsmodellierung: Können wir die Brücke zwischen Anwendung und Entwicklung stabilisieren? In: Modellierung 98 GI

Jiao JR, Chen C-H (2006) Customer requirement management in product development: a review of research issues. Concurrent Engineering 14(3):173-185

Korell M, Ganz W (2000) Design hybrider Produkte. Der Weg vom Produkthersteller zum Problemlöser. In: Bullinger H-J, Hermann S (eds) Wettbewerbsfaktor Kreativität: Strategien, Konzepte und Werkzeuge zur Steigerung der Dienstleistungsperformance. Gabler, Wiesbaden, pp 153-159

Lucia AD, Fasano F, Oliveto R, Tortora G (2007) Recovering traceability links in software artifact management systems using information retrieval methods. ACM Trans. Softw. Eng. Methodol. 16(4):13-63

Nuseibeh BA, Easterbrook SM (2000) Requirements engineering: a roadmap. In: 22nd international conference on software engineering. ACM Press, New York, pp 35-46