INTERNATIONAL DESIGN CONFERENCE – DESIGN 2022
https://doi.org/10.1017/pds.2022.198

Avoid Service Design Trap by Guiding Product/Service System Design with Product-Service Dependency Knowledge Base

T. Sakao ✉, A. Neramballi and J. Matschewsky

Linköping University, Sweden

✉ tomohiko.sakao@liu.se

Abstract

This article aims to contribute to the knowledge on product/service system (PSS) design practice as follows. First, a new rationale for why PSS design in practice often does not exploit its full potential is given based on a theory on inseparability of services. Second, a representation of the dependencies between product design parameters (DPs) and service DPs in a form of potential-dependency knowledge base is proposed as a remedy. Third, reusability of the knowledge captured from product-service integration across sectorial borders is shown with PSS design at a complex product manufacturer.

Keywords: product-service systems (PSS), knowledge management, concurrent engineering (CE), design management, design methods

1. Introduction

An increasing number of manufacturers aim to design and provide product/service systems (PSSs) (Brambila-Macias *et al*., 2018). This trend of servitization is being strengthened by digitalization, which gives manufacturers opportunities to better design PSSs harvesting data of in-service products (Braune *et al*., 2018). A PSS is a system, where a whole is not a mere collection of parts but builds upon the relationships between its parts. Thus, in conceptual PSS design, addressing the dependencies between products and services is a critical issue (Meier *et al*., 2010). A well-designed PSS will be able to adequately cope with uncertainty of the product and services (e.g., product deterioration speed) in the use phase (Sakao and Neramballi, 2020) to create higher value.

Designing a PSS is more complex than designing its product part, not only because of the added service part but also due to the integration of inter-dependent products and services (see e.g. (Trevisan and Brissaud, 2017)). Previous empirical research has indicated the prevalence of challenges faced by manufacturers that hinder their potential to exploit the dependencies, particularly for the synthesis part of PSS design. The root cause for these challenges is not yet well documented. However, two reasons reported are the prevalence of i) spatial and temporal separation of product and service design activities due to the silo structure in a large organization (Matschewsky *et al*., 2018) and ii) limited attention afforded to the dependencies by experienced designers of manufacturing companies, even in a laboratory setting without the silo structure (Neramballi *et al*., 2021). These reports support the notion that designers need guidance to effectively integrate product and service elements and thus exploit the full potential of PSSs.

To effectively guide PSS designers in identifying and integrating such dependent product and service elements, a support tool that will reduce the random search for and iterations of integration of such elements is needed. However, such practical support for daily design tasks in industry is still limited in the literature. Therefore, this article contributes to the knowledge on PSS design practice as follows: I) a new rationale for why PSS design in practice often does not exploit its full potential is given based on a theory of inseparability between production and use of services (Section 2); II) a representation of the dependencies between product design parameters (DPs) and service DPs in a form of potential-dependency knowledge base is proposed as a remedy (Section 3); and III) reusability of the knowledge captured from product-service integration across sectorial borders is shown with PSS design at a complex product manufacturer (Section 4).

2. Problematizing PSS design activities in a manufacturer

In this article, products and services are physical goods and human activities, respectively, and both are included in an offering. Products and services differ substantially from each other: a primary difference in the context of production is denoted as the inseparability of production and use in the case of services. That is, whereas products are produced and subsequently used, services are produced and used simultaneously (Regan, 1963, Xin et al., 2013). The use of products and services occurs at any point in time after the offering is deployed to the user. Any design must be completed before its associated production. Thus, a contrast emerges between products and services about when design, production and use possibly take place, as depicted by (Figure 1). Before the start of product use, design and then production must occur for products, while only design can be carried out for services. After the start of product use, its design is generally frozen, while services can still be designed as a result of its inseparability principle. Thus, higher flexibility is afforded to when service design completes.

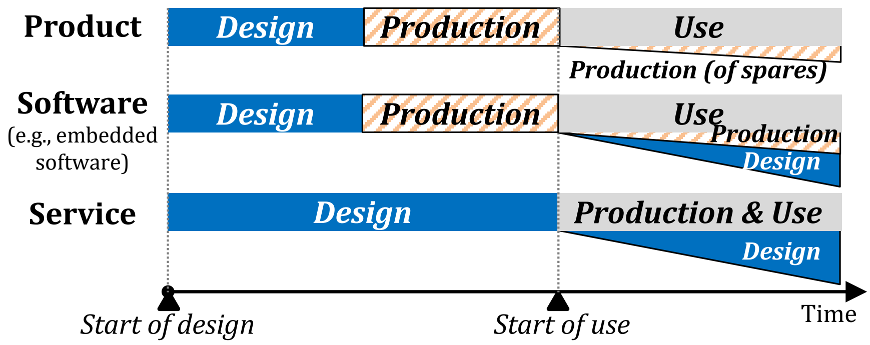

Figure 1. Possible time windows to design product, software and service.
Note: This figure is simplified, as concurrent design and production of products is not an issue of this article. Software is also compared here but is not a primary target in this article.

This higher flexibility on the temporal dimension allows PSS designers to postpone service design in part (i.e., not entirely) and thereby effectively and efficiently address the uncertainty arising in the use phase, which grants a competitive advantage to a manufacturer when providing PSSs. For instance, decisions on concrete overhaul operations with specific tools can be postponed until sufficient information about the product conditions during use by the specific use environment becomes available. However, this advantageous opportunity is often not optimally seized in practice by manufacturers. In practice, too many or too few decisions are made for the service design at the end of product design, e.g., assigning excessive resources (e.g., reserved service technicians and shipped spare parts) fixed to the service and keeping the mechanism principle for the overhaul undecided. This type of sub-optimal design causes substantial negative effects on the entire PSS (Matschewsky et al., 2018, Aurich et al., 2006, Baxter et al., 2009).

Furthermore, in practice, the needs of the services to be deployed during the product use phase tend not to be considered by or communicated to the product designers before the end of the product design phase, which may hinder effective service provision. For example, according to the authors' first-hand information from a manufacturer, physical components of a PSS were designed in a way that benefits manufacturing but paid insufficient attention to the needs of service. This led to a physical design that necessitated the use of a crane to disassemble the product for a frequent and basic maintenance task, as the critical component to be serviced was placed inconveniently.

https://doi.org/10.1017/pds.2022.198 Published online by Cambridge University Press

These issues present the risk for a lost opportunity for the optimal or improved design of PSSs. To efficiently communicate this risk, occurring in a planned manner or not, a novel term, service design trap, is proposed referring to a phenomenon where the ability of service designers over an extended timespan (shown in Figure 1) is restricted due to the constraints set by the frozen product design during the use phase. The word trap is employed here to mean a service designer getting into a confining situation in an unexpected and unwanted manner from the service designer's viewpoint. The service design trap is closely related to challenges previously described regarding PSS design, such as a lacking lifecycle perspective (Matschewsky *et al*., 2018). In this case, service design trap serves as an operationalization to support practitioners in taking concrete steps to overcome abstract challenges.

To facilitate the understanding of the risk of falling into the service design trap, the phenomenon is visualized by (Figure 2): A service designer is standing in the PSS design space. The service designer has access to both the product and service design spaces, meaning the possibility to interact and influence the design at the product and service design departments is present before the use phase. It is crucial to consider the interdependent needs of product and service design before the use phase, after which the product design will be frozen. However, the service designer is not sufficiently guided regarding the product-service inter-dependencies. Such a lack of consideration of the product-service inter-dependencies before the use phase may cause substantial negative impacts on the PSS performance in the later stages. The service designer may get trapped on the right-hand side after he or she enjoyed the higher freedom for the service design in an inappropriate manner on the left-hand side. The key questions for the service design are; which parts in the service design ought to be decided in integration with the product design and which parts ought to be addressed during the use phase. To answer these questions, understanding potential dependencies between the product and service design is central. Thanks to digitalization and smart products, the increasing possibilities and impacts of collecting and utilizing data for efficient PSS provision is key to manufacturers, necessitating an even more timely and tighter integration between product and service elements.

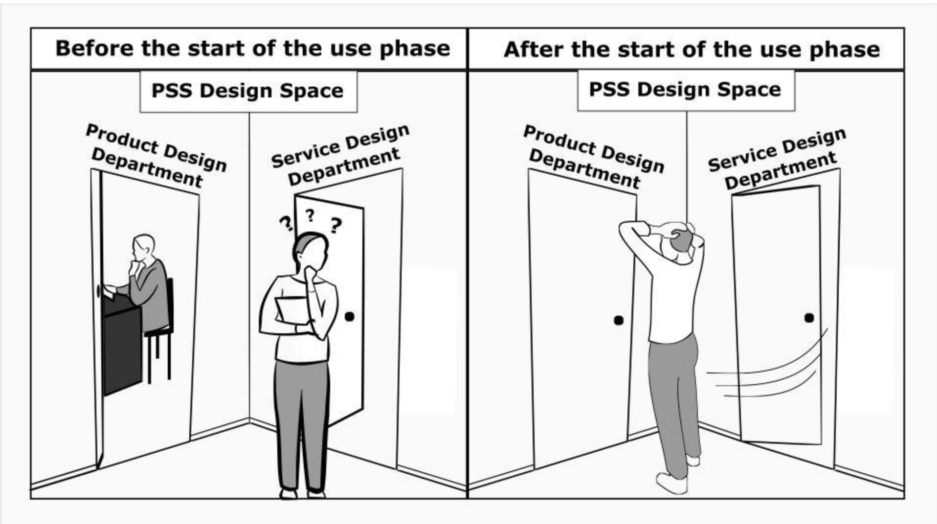

Figure 2. Service design trap visualized in the context of PSS design

3. Proposal of PSS design procedure for manufacturers

3.1. Overview of background method and proposal

This proposal builds upon a PSS design method that informs relatively important design parameters (DPs) through performing quality function deployment (QFD) and was validated and implemented at complex product manufacturers (Matschewsky *et al*., 2018). It aims to help PSS designers with effective and efficient value creation and takes the form of a procedure consisting of seven iterative steps; see (Figure 2). Its new elements are focused on below.

https://doi.org/10.1017/pds.2022.198 Published online by Cambridge University Press

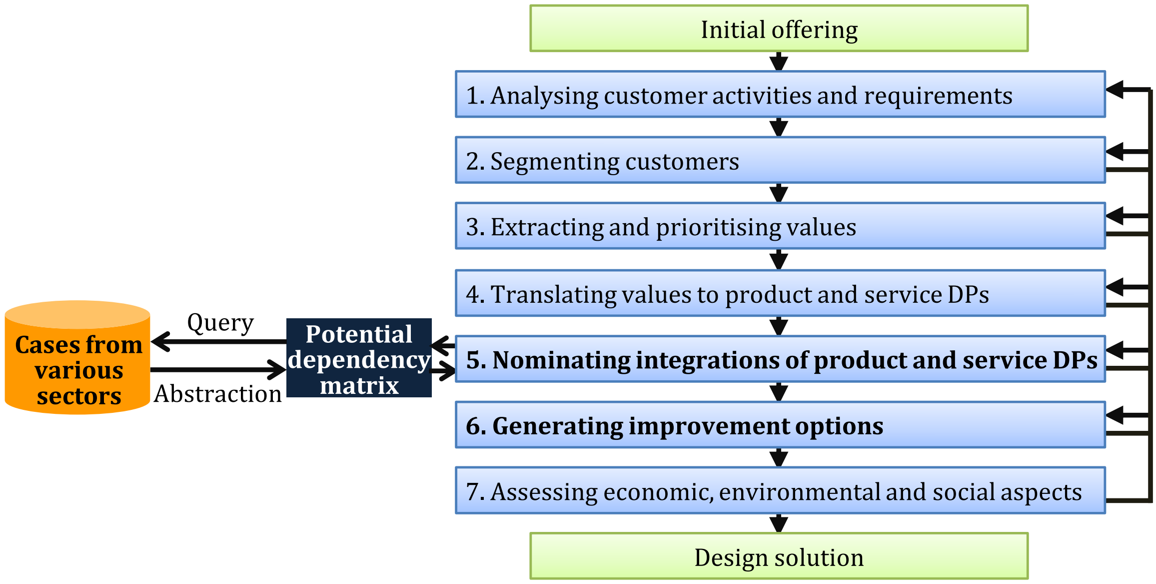

Figure 3. Procedure based on (Matschewsky *et al.*, 2018) with its new elements in bold

The new proposal aims to inform PSS designers of latent dependencies between important product and service DPs in a PSS at hand and identify the relevant opportunity to integrate them in an early design phase avoiding the service design trap. The integration here refers to the effective utilization of the dependencies between them according to the PSS design intent. For example, the maintenance frequency of a vehicle (a service DP) can be effectively designed for a better PSS by collecting and utilizing the time series data of the acceleration captured by attaching a sensor to a critical mechanical component (a product DP). The main effect lies in the "interweaving" of relevant product and service DPs, thereby preventing the allocation of a random value or void for the respective DPs. This "interweaving" is also expected, in the design phase prior to the use phase, to better inform product designers and service designers about the PSS in the use phase (see Figure 2) and thereby facilitate them better design products and services. Further, the DPs brought up at Step 5 are expected to aid PSS designers to also investigate the exchangeability between product and service DPs (Sakao and Lindahl, 2015), as this higher freedom of design is a major advantage of PSS design.

3.2. Potential dependency knowledge base between products and services

To provide knowledge on the dependencies between product and service DPs that has broad applicability, it is useful to create a set of generic DPs for products and services (similar to the PSS-pattern notion (Maleki *et al.*, 2018)). For this purpose, DP classes are proposed here based on the authors' review of case studies of PSS design practice by original equipment manufacturers (OEMs) of complex products reported in peer-reviewed scientific journal papers. This review was built upon a previous comprehensive review (Neramballi *et al.*, 2020) using the snowballing approach which was not specific to PSSs by OEMs. While the previous work focused on abstracting service design characteristics, this review was extended to include product design characteristics. The descriptions of PSS design practice in the case studies were analysed, explicitly described instances of dependencies between or integration of product and service elements reported in the cases were identified and shortlisted. A total of five product DP classes and seven service DP classes were abstracted from 42 shortlisted instances of dependencies or integration of product-service elements. The sources originate of sectors in production machines, construction equipment, vehicles, industrial pumps, aircraft engines, white goods and medical equipment; the details including the reference information are shown in the supplementary material to be published. The service DP classes are partly based on (Neramballi *et al.*, 2020). Although this is not an exhaustive list of DP classes, it represents multiple instances of product and service elements that were documented to be either dependent on or integrated with each other across multiple OEM sectors.

https://doi.org/10.1017/pds.2022.198 Published online by Cambridge University Press

The proposed product DP classes are:

- serviceability – the ability of a product that allows it to be serviced effectively and efficiently;
- affordability – the financial cost of the product to be incurred by the customer or provider;
- information transferability – the ability to collect and transfer information from the production or use phase of the product to the other phases;
- upgradeability – the ability of a product that allows it to be upgraded or evolved; and
- temporal functionality – the ability of product elements to address the functional requirements over use time.

The service DP classes are:

- labor intensity – the level of involvement of human resource in the service activity;
- capability – the ability of service personnel to address the requirements;
- responsiveness – the time taken to deliver a service/response to the customer;
- information transferability – the ability to collect and transfer information from the production and use phases to the other phases;
- affordability – the financial cost of the service to be incurred by the customer or provider;
- frequency – the number of services provided over a period of time; and
- reliability – the level of dependability of fulfilment of the intended functions of a service activity.

Table 1 shows the potential dependency matrix representing the identified dependencies, categorized under three archetypes. When a service element unilaterally depends on a product element, the unilateral dependency denoted as SuP is identified, but when a product element unilaterally depends on a service element, the unilateral dependency, PuS, is identified. When both PuS and SuP exist, the bilateral dependency PbS is identified. A total of 20, 11 and 11 cases were found for the dependency archetypes SuP, PuS and PbS, respectively.

Two dependencies are exemplified here. Aurich *et al*. (Aurich *et al*., 2006) reported that ensuring the effective operation of a hydraulic system component (the product DP class, temporal functionality) necessitated the specification of maintenance services (changing filters and cleaning the pumps) in periodical intervals (the service DP class, frequency), corresponding to the PuS dependency. Baxter *et al*. (Baxter *et al*., 2009) reported that a special tooling in service (abstracted as the service DP class, capability) was required to be developed to enhance the serviceability of the product (the product DP class, serviceability), corresponding to the SuP dependency. On the other hand, a null dependency is given to 20 combinations (out of 35) in this matrix (see the shaded cells in Table 1), e.g., the affordability of the product and the reliability of the service, and the product upgradeability and the service responsiveness.

Table 1. Potential dependencies between product and service DP classes. Note: SuP means a service element depends on a product element, PuS means a product element depends on a service element, PbS means bidirectional dependencies, and a shaded cell gives a null dependency.

Service DP class \ Product DP class	Serviceability	Affordability	Information transferability	Upgradeability	Temporal functionality
Labour intensity	PbS		PbS		
Capability	PbS		SuP	SuP	PuS
Responsiveness	PbS				
Information transferability	PuS		PbS	PuS	PuS
Affordability	SuP	PbS			
Frequency					PbS
Reliability					PuS

https://doi.org/10.1017/pds.2022.198 Published online by Cambridge University Press

3.3. Integrating product and service DPs with software support

The procedure can be performed partly with a support implemented on commercial software (Microsoft Excel) in the following order. This support is expected to reduce the time required to carry out Steps 5 and 6 of (Figure 2). See (Figure 3) for the software structure.

a) The designer inputs product and service DP instances in the software when performing QFD Phase I in Step 4.
b) The designer chooses respective DP classes for the DPs from the options (Section 3.2) in Step 5.
c) The software identifies and shows the specific combinations of DP classes with integration potential based on the matrix (Table 1) and thereby, those of DP instances. A designer facing a specific integration potential can jump with a hyperlink to the contextualized case descriptions (from the 42 instances) of the same kind of integration to assess the applicability to the PSS at hand.
d) These combinations can be compared for prioritization using the importance of the DPs, which is derived from Step 4 (Neramballi *et al*., 2020). In Step 6, the prioritized combinations are investigated to generate improvement options.

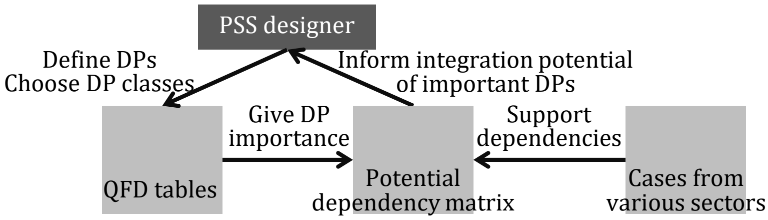

Figure 4. Software support automating the procedure in part.

4. Validation of proposed concept and procedure

4.1. Introduction

Companies Alpha and Beta were selected as partners for the research. They are both globally active industrial companies providing PSSs with strong legacies in the design and production of products and are among the world leaders in their respective domains. Beta is an OEM of logistics equipment and solutions. In recent years, the company has made substantial progress towards integrating products and services with a lifecycle perspective starting with the earliest design stages. Today, the majority of the company's revenue is generated through PSSs, most of which are use-oriented offerings. Alpha is a globally active industrial company in the area of power generation. They today generate a portion of their business through PSSs. The portions of products and services in terms of revenues are similar to each other. Although the design processes remain product-centric, an increasing push towards the integration of products and services towards a lifecycle-oriented offering has been made.

4.2. Pre-study with PSS provider Beta

A workshop was held in December 2020 with key Beta staff to validate the service design trap concept (see Section 2) as well as the potential dependency knowledge base (see Section 3.2). Two R&D managers, a group manager R&D mechanical design, and a manager in the area of integrated solutions & automated devices, participated in the validation workshop. Additionally, seven workshops with Beta staff were held in 2019 focusing on increasing the effectiveness and efficiency of PSS design, as part of an ongoing collaboration with Beta focused on PSS design since 2014. The data gathered during these efforts were a basis of the procedure presented in Section 3.

The participants enumerated several examples of having fallen into the service design trap, e.g., through excessive integration of components into hard-to-service modules or lacking consideration for particularly taxing use-cases leading to high wear and tear on underspecified components and, therefore, high lifecycle cost. Further, the increasing possibilities and impacts of collecting and

https://doi.org/10.1017/pds.2022.198 Published online by Cambridge University Press

utilizing data (e.g., data of product use environments) for efficient PSS provision was particularly key to practitioners, posing challenges in design and necessitating an even earlier and tighter integration between product and service design. The proposed procedure of tackling dependencies between these domains through the dependency knowledge base was seen as helpful to support the concurrent design of products and services, which is critical for their business. A particular benefit pointed out was an anticipated mutual understanding for constraints arising early between currently largely siloed product and service design divisions.

4.3. Study with PSS provider Alpha

4.3.1. Validation by PSS designers' viewpoints in Alpha

To assess the value and feasibility of using the procedure (Section 3.1), in particular, the potential dependency knowledge base (Section 3.2) and the software (Section 3.3), collaboration with Alpha was performed through a number of workshops with managers from product design, service design and sales, among others. In particular, Alpha performed a large-scale comprehensive QFD for their PSS, with the authors' facilitation, to guide the PSS design process with a lifecycle perspective (Steps 1 to 4 of Figure 2); a QFD that Alpha had performed only for the product was built upon. The QFD for PSS included a total of 20 and 16 DPs for the product and services, respectively, in the PSS.

In Step 5, first, a total of 7 and 13 specific DPs for the product and services, respectively, were found relevant to be classified under the DP classes (Section 3.2). The identified product DP classes covered all the registered ones, while the identified service DP classes were capability and information transferability. Most of the product DP instances (13 out of 20) were found not belonging to any of the registered classes. After applying the potential dependency knowledge base, the majority (ca 60%) of all the derived dependencies were judged to be sensible: the implausible dependencies derived were mostly related to the service DP class *capability* – the reason is its definition is too broad to match the DP instance precisely with the reported cases (e.g., the temporal functionality of a product depends on the capability of a service on the class level, but, as their respective instances, the ability of a component degrading over use time and the quality of logistic planning do not really hold the dependency). Also, the majority (ca 80%) of the combinations that were screened out by the matrix were assessed to be irrelevant for integration: the overlooked dependencies were mostly related to product DP instances that concern the flexibility and adaptability of the product to its use environment – the reason is that no such product DP class is registered at present.

The obtained potentially critical dependencies, including the archetypes, were then evaluated with respect to the current level of exploitation by the managers. Many were already considered for exploitation, but a part of them was not allowed by their customer to be exploited because of the secrecy of the data of in-service products. Many were also actually exploited; however, some were discovered to be worthy of a more thorough investigation. Among others, the issue that failures in the condition monitoring system may lead to false failures being reported by service was found easily overlooked without the proposed matrix in place.

4.3.2. Other feedback on the proposal from managers in Alpha

The managers enumerated several examples of having fallen into the service design trap. For example, the service staff had to cut through metal to get to the gearbox of the product, so that they could carry out the service that had been promised. Overall, the concept of the service design trap and the procedure were considered sensible to be used, as they help with the identification and prioritization of the important dependencies in design as well as the pedagogy to less experienced designers. Also, it was made clear that a formal way of taking account of these dependencies is missing in the existing design process at Alpha and thus remains largely dependent on the initiative of the relevant individuals. Further, a focus on exploiting the dependencies between products and services is, although partly implicitly, present; however, there is a lack of such focus when it comes to update and upgrade design projects, which, in turn, will benefit from an explicit support procedure. The software was also assessed useful to shorten the time and to get outcomes based on scientific evidence from other sectors.

https://doi.org/10.1017/pds.2022.198 Published online by Cambridge University Press

5. Discussion

5.1. Evaluation of the proposal

Section 2 contributed to enhancing the theoretical comprehension of PSS design challenges through describing the service design trap, which practitioners confirmed to be prevalent in manufacturers' practice (Section 4). Then, the systematic PSS design procedure proposed as a remedy was, in real practice, found effective, particularly in nominating only the pairs of product and service DPs that are relevant for integration (Section 4.3). For using the matrix (Section 3.2), the information required is considered available at PSS providers in general and in a conceptual stage as well as familiar to practitioners in industry; the effort additionally required is assumed to be outweighed by the impacts. The software (Section 3.3) was also found to increase efficiency by partly automating the designers' process. These together indicate that the procedure is a systematic yet practical easy-to-use support, and they imply a high potential of its uptake in the early stages of PSS design in industry.

The knowledge base (Section 3.2) built upon cases of integration from other sectors than the case company Alpha (Section 4.3), showing the possibility to reuse knowledge about PSS integration across sectors; in most of the advanced works on knowledge reuse in PSS design such as (Maleki et al., 2018, Baxter et al., 2009), the cross-sectoral reuse was not the primary aspect of validation. It should be noted that some implausible dependencies are obtained from the knowledge base because the knowledge rests mostly on other sectors implying the presence of different contextual dependencies. Also, this knowledge base is supposed to be updated with new information recognized in PSS design in industry; a few new combinations are foreseen to have dependencies in real practice, e.g., the information transferability of the product and the responsiveness of the service.

5.2. Implications to theory, practice and society

To design a complex system effectively and efficiently, analysing dependencies between DPs is a critical task and developing a support for design with the dependencies is fundamental. The dependency design is addressed also in other types of systems than PSSs. In systems engineering, the concept stage, when the systems engineer coordinates the activities of engineers from many disciplines (INCOSE, 2006), is relevant to the service design trap. Technically speaking, the INCOSE handbook (ibid) recommended, in the architectural design process, use of an integrated product team breaking down communication silos and architecture and design patters as an approach. Also, mechatronic engineering design involves mechanical, electrical and control engineering that need to work closely together for optimal designs (Prösser *et al.*, 2013). Further, cyber physical systems involve the necessity of an integrated, simultaneous design of network and process (Penas *et al.*, 2017). Design research investigated, for instance, knowledge representation in different disciplines or domains (e.g., Eisenbart *et al.*, 2013). However, little research shed light on differences in the time dimension of the design process between disciplines: this article is novel in explaining causes for poor design of the dependencies through the differences on the time dimension and addressing the dependencies between product and service DPs through the knowledge base aiming for the adequate fusion of product and service elements.

User experience (UX) design, which includes designing interaction between a user and a design solution (Sharp *et al.*, 2007), aims to design user experience with the products and/or services. It can incorporate the results of user test on the products and/or services by simulating what happens in the use phase (Ekşioğlu, 2016). In theory, it is possible to test an entire PSS with users before finalizing the design. However, it is unrealistic and even not optimal to do so in industrial practice with a high pressure on shortening the lead-time. Therefore, it is vital to better support the design phase, which the proposed procedure aims to contribute to.

How the dependencies are addressed in this paper has commonality to the contradiction matrix with 39 features by TRIZ (Altshuller and Altov, 1996), which is developed to date based on more than one million patents, in that DPs are abstracted and thereby designers benefit from patterns of problem solving captured from cases reported in other sectors. The authors' intention is to continue to enrich and improve the knowledge base, including the archived cases as well as product and service DPs, and

https://doi.org/10.1017/pds.2022.198 Published online by Cambridge University Press

thereby to increase its effectiveness. This will be carried out together with application of the proposed procedure to design in industry.

When manufacturers with a strong legacy in product design carry out PSS design in practice, they are likely to add service DPs to existing product DPs, resulting in more DPs than functional requirements (FRs); see Suh's axiomatic design (AD) (Suh, 1998), where the number of FRs and DPs is ideally equal (Suh, 1998). By embedding specific relationships, as design, between product and service DPs after consulting the dependency knowledge base, the number of DPs to be handled simultaneously could be decreased. Thereby, the complexity of PSS design at hand can be decreased. Further, it can be used to maintain the independence axiom (Suh, 1998) in PSS design. Importantly, this type of design can be performed by OEMs because of their advantage to control both design and operation, but not by a third-party service supplier.

When zooming out to the production industry, vertical integration is among the relevant issues with the current digitalization trend (Culot *et al*., 2020). This article provided concrete guidance to a manufacturer's micro-level vertical integration through design. From the societal perspective, this integration of products and services via design provides a possible ground to decrease system failures in the service sector; (Nakao *et al*., 2009) analysed failures in the service sector with the AD lens and showed ca 70% of them involved uncoupled but unrealized design where an FR that is not affected by the available DPs existed. This problem could be resolved, in theory, by adequate integration between product and service DPs, as explained in this article. If not resolved in practice, it can be better understood thanks to the service design trap. The picture in (Figure 2) was used to explain the concept of the service design trap in this article and is expected to improve the learning of the concept, because such improvement with pictures is widely recognized in different domains such as education (Bobek and Tversky, 2016) and health communication (Houts *et al*., 2006). Furthermore, the control of the whole lifecycle is more demanded by the transformation to a circular economy, and this proposal will also be useful in this context.

6. Conclusion

This article provided a new rationale for the PSS design challenges with the help of the service design trap and proposed a procedure involving the potential dependency knowledge base and the software support to integrate product and service elements in conceptual PSS design. Future work will include registration of more cases in the repository and applications to more PSS design cases, where readers are invited to participate by using the software to be freely accessible (the Excel-based software is to be made accessible as the supplementary material at the data base of the authors' institution). Another potential future work may involve the investigation of the effects of the P-S dependency knowledge base on conceptual PSS design processes.

Acknowledgements

This research was supported by the Mistra REES (Resource Efficient and Effective Solutions) program (Grant No. 2014/16) and funded by Mistra (The Swedish Foundation for Strategic Environmental Research). The drawing of the service design trap (Figure 2) was performed by Ambra Giulianelli of Linköping University with the authors.

References

Altshuller, G. and H. Altov (1996). And suddenly the inventor appeared-TRIZ, the theory of inventive problem solving. Worcester: MA, Technical Innovation Center.

Aurich, J. C., C. Fuchs and C. Wagenknecht (2006). "Life cycle oriented design of technical Product-Service Systems." Journal of Cleaner Production 14(17): 1480-1494 DOI: 10.1016/j.jclepro.2006.01.019.

Baxter, D., R. Roy, A. Doultsinou, J. Gao and M. Kalta (2009). "A knowledge management framework to support product-service systems design." International Journal of Computer Integrated Manufacturing 22(12): 1073-1088 DOI: 10.1080/09511920903207464.

Bobek, E. and B. Tversky (2016). "Creating visual explanations improves learning." Cognitive Research: Principles and Implications 1(27) DOI: 10.1186/s41235-016-0031-6.

Brambila-Macias, S., T. Sakao and C. Kowalkowski (2018). "Bridging the Gap between Engineering Design and Marketing: Insights for Research and Practice in Product/Service System Design." Design Science 4 DOI: 10.1017/dsj.2018.3.

Braune, A., G. Hüttemann, M. Klein, U. Löwen and M. Thron (2018). Usage Viewpoint of Application Scenario Value-Based Service. Berlin: 28.
Culot, G., G. Orzes, M. Sartor and G. Nassimbeni (2020). "The future of manufacturing: A Delphi-based scenario analysis on Industry 4.0." Technological Forecasting & Social Change 157 DOI: 10.1016/j.techfore.2020.120092.
Eisenbart, B., K. Gericke and L. T. M. Blessing (2013). "An analysis of functional modeling approaches across disciplines." Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27 DOI: 10.1017/S0890060413000280.
Ekşioğlu, M. (2016). "User Experience Design of a Prototype Kiosk: A Case for the İstanbul Public Transportation System." International Journal of Human–Computer Interaction 32 DOI: 10.1080/10447318.2016.1199179.
Houts, P. S., C. C. Doak, L. G. Doak and M. J. Loscalzo (2006). "The role of pictures in improving health communication: A review of research on attention, comprehension, recall, and adherence." Patient Education and Counseling 61: 173-190 DOI: 10.1016/j.pec.2005.05.004.
INCOSE (2006). "Systems Engineering Handbook - a guide for system life cycle processes and activities." C. Haskins ed., International Council on Systems Engineering. INCOSE-TP-2003-002-03.2. International Council on Systems Engineering.
Maleki, E., F. Belkadi, N. Boli, B. J. van der Zwaag, K. Alexopoulos, S. Koukas, M. Marin-Perianu, A. Bernard and D. Mourtzis (2018). "Ontology-Based Framework Enabling Smart Product-Service Systems: Application of Sensing Systems for Machine Health Monitoring." IEEE Internet of Things Journal 5(6): 4496-4505 DOI: 10.1109/JIOT.2018.2831279.
Matschewsky, J., M. L. Kambanou and T. Sakao (2018). "Designing and providing integrated productservice systems – challenges, opportunities and solutions resulting from prescriptive approaches in two industrial companies." International Journal of Production Research 56(6): 2150-2168 DOI: 10.1080/00207543.2017.1332792.
Meier, H., R. Roy and G. Seliger (2010). "Industrial Product-Service Systems - IPS²." CIRP Annals - Manufacturing Technology 59(2): 607-627 DOI: 10.1016/j.cirp.2010.05.004.
Nakao, M., K. Tsuchiya and K. Iino (2009). "Three typical failure scenarios of the mind process of design from the Axiomatic Design perspective." CIRP Annals - Manufacturing Technology 58(1): 165-168 DOI: 10.1016/j.cirp.2009.03.018.
Neramballi, A., T. Sakao and J. Gero (2021). How do designers think in systems? – Empirical insights from protocol studies of experienced practitioners designing Product/Service Systems. Proceedings of the 9th International Conference on Design Computing and Cognition (DCC'20).
Neramballi, A., T. Sakao, S. Willskytt and A.-M. Tillman (2020). "A design navigator to guide the transition towards environmentally benign product/service systems based on LCA results." Journal of Cleaner Production 277 DOI: 10.1016/j.jclepro.2020.124074.
Penas, O., R. Plateaux, S. Patalano and M. Hammadi (2017). "Multi-scale approach from mechatronic to Cyber-Physical Systems for the design of manufacturing systems." Computers in Industry 86 DOI: 10.1016/j.compind.2016.12.001.
Prösser, M., P. Moore, X. Chen, C.-B. Wong and U. Schmidt (2013). "A new approach towards systems integration within the mechatronic engineering design process of manufacturing systems." International Journal of Computer Integrated Manufacturing 26 DOI: 10.1080/0951192X.2013.785026.
Regan, W. J. (1963). "The Service Revolution." Journal of Marketing 47(July): 57-62 DOI: 10.1177/002224296302700312.
Sakao, T. and M. Lindahl (2015). "A method to improve integrated product service offerings based on life cycle costing." CIRP Annals - Manufacturing Technology 64(1): 33-36.
Sakao, T. and A. Neramballi (2020). "A product/service-system design schema: Application to big data analytics." Sustainability 12 DOI: 10.3390/su12083484.
Sharp, H., Y. Rogers and J. Preece (2007). Interaction Design: Beyond Human-Computer Interaction.NY, Wiley.
Suh, N. P. (1998). "Axiomatic Design Theory for Systems." Research in Engineering Design 10: 189-209 DOI: 10.1007/s001639870001.
Trevisan, L. and D. Brissaud (2017). "A system-based conceptual framework for product-service integration in product-service system engineering." Journal of Engineering Design 28(10-12): 627-653 DOI: 10.1080/09544828.2017.1382683.
Xin, Y., K.-H. Chai, V. Ojanen and A. Brombacher (2013). "The moderating effects of service solution characteristics on competitive advantage." The Service Industries Journal 33(15-16): 1633-1658 DOI: 10.1080/02642069.2011.638918.