Data-Centric Architecture Model for the Development of Smart PSS

Yevgeni Paliyenko[1], Sean Ryan Mueller[1], Daniel Roth[1], Matthias Kreimeyer[1]

[1]University of Stuttgart – Institute for Engineering Design and Industrial Design

Abstract: In the current industrial landscape, the growing integration of digital technologies and interconnectivity have been instrumental in reshaping business paradigms. Central to this transformation are data-driven business models that leverage data to create value. Particularly impactful are smart Product-Service Systems (sPSS), which embody the convergence of smart products and digital services into cohesive solutions that meet specific customer demands. This paper introduces a data-centric architecture model for sPSS that is constructed around building blocks representing essential system entities. This model facilitates the detailed mapping and management of data and information flows among these components. It delineates specific data needs necessary for the effective performance and ongoing evolution of sPSS as well as laying the groundwork for the inception of new functionalities or services by revealing comprehensive data and information resources. Thus, the model supports the value-oriented development of sPSS by focusing on the data needs essential for system performance.

Keywords: smart Product-Service-Systems, data-driven design, systems engineering, architecture model, product development

1 Introduction and problem clarification

The rapid evolution of digital technologies and the advancing interconnectivity have profoundly changed the industrial landscape over the last decades. This transformation, often referred to as the fourth industrial revolution or Industry 4.0, is characterized by digitalization and networking, causing a revolutionary shift in the conception, development, and usage of products and services. This era is marked by the emergent role of data and information as central pillars of value creation, leading not only to the optimization of existing offerings but also to the creation of new functionalities and business models (Abramovici, 2018; Otto et al., 2018).

At the heart of this revolution is the emergence of digital business models that create value through the extensive use of data. This development is enabled by the application of information and communication technologies (ICT) such as the Internet of Things (IoT), smart products, and big data technologies (Chowdhury et al., 2018). In this context, smart Product-Service Systems (sPSS) play a crucial role. They embody the seamless integration of intelligent products and digital services into a comprehensive solution tailored to customer needs, thereby generating additional value for all participants in the value creation network (Valencia et al., 2015; Porter and Heppelmann, 2014). The high performance and low costs of ICT drive this development forward, yet they also open up new challenges. The ability of sPSS to flexibly respond to changing requirements and market conditions represents a significant competitive advantage. This adaptability is enabled through continuous development and improvement across the entire lifecycle, necessitating constant data and information exchange as well as intensive collaboration with customers and other stakeholders.

Despite the numerous opportunities digitalization brings, the implementation and further development of sPSS pose significant challenges, thus companies face the task of dealing with complexity and the efficient processing of large volumes of data (Paliyenko et al., 2022; Heuchert et al., 2020). These problems manifest in the need to develop suitable structures and processes to effectively manage the underlying data and information flows. Implementing appropriate processes for data utilization and processing is crucial to generating value from available data (Zheng et al., 2021; Eigner, 2021). A deeper analysis and understanding of these flows are crucial to optimize the performance of sPSS and support their continuous improvement. Therefore, the development and implementation of sPSS require a methodical approach that overcomes the complexity of these systems and effectively utilizes the multifaceted data streams (Paliyenko et al., 2023b; Rizvi and Chew, 2018).

From these challenges and gaps the central problem statement of this work emerges: there is a lack of a comprehensive model that structures the data and information flows in a sPSS. Such a model would not only enable the identification of different entities and their relationships but also reveal the specific data and information demand for the operation and further development of sPSS. Therefore, the objective of this work is to develop a model that provides an overview of the data and information flows in an sPSS. The model aims to: 1) Identify the various entities within the sPSS and their relationships to each other; 2) Show the specific data and information needs for the service delivery and the continuous development of the sPSS; 3) Provide a basis for the development of new functionalities or services by uncovering all available data and information; 4) Support the feedback of data into development to make future sPSS more efficient and innovative. To achieve these goals, a methodical approach is pursued that encompasses both theoretical foundations and practical application examples.

2 Methodology

An in-depth analysis of the literature concerning sPSS alongside data and information flows was conducted, leading to an examination of different methods to model data flows within sPSS. Following this, a detailed model was designed to outline the complex dynamics present in sPSS, which included pinpointing a variety of stakeholders and system components, as well as delineating data (including types, availability, and sources) and their utilization in the development and operation of sPSS. The model's validation was achieved through academic rigor, integrating critiques to ensure its alignment with the existing practices. The methodology's flow diagram is depicted in Figure 1.

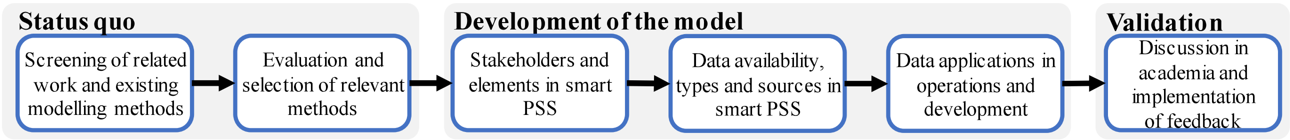

Figure 1. Approach to research

Numerous potentially feasible methods for modeling sPSS are available, with certain methods particularly designed for depicting data flows, while others concentrate on the architecture of these systems. Relevant models and methods to this study are outlined in Table 1. These methodologies provide diverse perspectives and strategies for representing the complex architecture of sPSS, playing a crucial role in the development and comprehension of these systems.

Table 1. Methods and models for sPSS

Methods for modeling data flows	Models for PSS system architectures
Model-Based Systems Engineering (Weilkiens, 2014)	**Value network of sPSS** (Paliyenko et al., 2023b)
MBSE is a methodology for holistic system development based on the use of models. It utilizes a variety of interconnected models to depict different aspects of a system and form a cohesive overarching model. The Systems Modeling Language (SysML) is often used for MBSE. Activity Diagrams and Internal Block Diagrams are particularly suitable for modeling data flows.	This approach describes the interaction of stakeholders in the sPSS value creation network. Five clusters of actors were identified: Customer, Technology Provider, Solution Integrator, Data Communication and Environment. The interactions in terms of system flows between these clusters are depicted using a GEMINI model. *Key findings:* Clusters of actors; high-level data flows
Business Process Model and Notation (OMG, 2011)	**Meta model of a CPPSS** (Rizvi and Chew, 2018)
BPMN is a method for the graphical representation of complex business processes. It uses activities, events, gateways and flows to represent different processes and decisions during a process. The responsibilities and results of activities can also be shown.	Rizvi and Chew describe two meta-models. The first model shows the interaction between CPS and PSS to create value, while the second depicts the development process of a CPPSS. It considers the entire life cycle and includes aspects such as business models and service types. An important aspect is the feedback of information to the provider for continuous improvement. *Key findings:* General sPSS structure and the connection to CPS; feedback of data into development
GEMINI Method (Echterhoff, 2018)	
A method for developing the service offering and structure for innovative business models. The method can depict enterprises, resources and activities needed for value creation. The relationships between entities are described as cash, service and information flows.	**Structure model for sPSS (**Liu et al., 2018)
FLOW method (Stapel and Schneider, 2012)	Liu's model describes sPSS as platform service ecosystems with different layers and stakeholders that are connected to this platform via ICT. The layers include smart devices, network, data management and applications. *Key findings:* Relevant elements and stakeholders in different hierarchical dimensions
This method is used to visualize information flows in software development processes. It uses activities and information with a solid (documents) or liquid (people or organizations) aggregate state.	
The following additional feasible methods and models were identified and considered:	
Methods: LINQ-Technique (Thuan et al., 2017), Knowledge Modeling and Description Language (Gronau et al., 2010), Knowledge and Information Management Model (Bastos et al., 2014), Information Channel Diagram (Durugbo et al., 2012), Data-flow diagram (Tangkawarow and Waworuntu, 2016) **Models:** Functional modelling of sPSS (Wu et al., 2021), Value Network Map (Olivotti et al., 2018), Architecture of platform-based ecosystems with sPSS (Bulut and Anderl, 2022), Generic PPS meta model (Idrissi et al., 2017)	

Frank and van Laak categorize the requirements for modeling business processes into three distinct groups: formal, user-related, and application-specific requirements (Frank and van Laak, 2003). Formal and user-related requirements are concerned with the syntactic and semantic accuracy, alongside the ease of use and practicality of the method. These fundamental requirements were considered in the initial search for modeling methods. The application-specific needs of the modeling method draw upon the criteria for a developmental framework for sPSS as established by Paliyenko et al., building on the foundational work of Keller and Binz (Paliyenko et al., 2023a; Keller and Binz, 2009). Additionally, precise requirements related to data flows were defined.

From this, seven key requirements emerged: **Adaptability** – The model must be flexible to accommodate various types of sPSS; **Interdisciplinarity** – It should visualize the connections and interactions among different (smart) products, (digital) services, and stakeholders; **ICT Integration** – It should illustrate the creation, storage, processing, and application of data and information via ICT; **Customer Focus** – The model needs to incorporate the customer, their issues, and needs centrally; **Expandability** – It should allow for expansion or modification to encompass additional PSS aspects not directly related to data flows; **Clear Representation of Data Flows** – It should clearly and comprehensively display the nature, source, and application of all data flows; **Data Feedback** – It must include a mechanism for feeding back data and information generated during use into the development process for validation and adaptation of the sPSS.

The evaluation of the collected methods and models was conducted through a two-phase approach, applying criteria derived from these requirements. Initially, a broad set of potential methods was preliminarily screened using a selection list to identify unsuitable candidates. The methods that passed this preliminary phase were then subject to a detailed utility analysis, where the importance of each evaluation criterion was determined through pairwise comparison. A key evaluation metric was the method's capacity to facilitate various combinations of systems elements. While many methods adeptly model data flows for specific processes or activities, few have the capacity to represent a large, complex system in its entirety. The evaluation identified Model-Based Systems Engineering (MBSE) as the most suitable for modeling data flows in sPSS, attributed to its versatile modeling capabilities and effective representation of complex systems integrating various element types. MBSE's efficacy across various applications, including its previous application to PSS, underscores its suitability for this purpose (cf. Halstenberg, 2022; Apostolov et al., 2018; Orellano et al., 2019)

3 Results

3.1 Architecture model of the data flows in sPSS

In order to develop the structural model for the data and information flows between different actors and elements the ‘Logical Architecture Diagram’ within Arcadia Capella, a freely available MBSE software developed by the Eclipse Foundation, was utilized (Voirin, 2018). This diagram is very similar to the ‘Internal Block Diagram’ used by SysML and enables the representation of logical structures, functions, and the interconnections among various components and actors.

The architecture model is divided into five main domains (Figure 2): Smart Product, Smart Service Platform, Communication Infrastructure, Provider, and Customer. Furthermore, additional individual elements for other actors and partners (suppliers, service providers and third parties) are included in the model. The main domains are divided into subsystems (dark blue) or sub-actors (light blue). Within these subsystems, functions (green) depict how data is processed by the entities. Different types of data and information are transferred between these elements to fulfill the functions of the systems. A detailed description of the data types and respective applications follow after the introduction of the model.

Smart Product: The smart product is prominently placed at the upper center of the model and is characterized as a cyber-physical system. Central to its design are elements such as sensors, actuators, software, and the Human-Machine Interface (HMI), as illustrated in Figure 3, which details their functions and interactions. Data from sensors, capturing information from both the product and its surroundings, is processed by the product's controller. This process involves analyzing the data to generate control signals for the actuators, based on the product’s operational settings and configurations. Subsequently, this data is aggregated and transmitted to the cloud for further storage and processing, as discussed in VDI/VDE 2206 (2021).

The product's behavior can be modified through various settings and configurations, adapting to factors like usage patterns, personal preferences, data availability, or optimization objectives. Such adjustments may be implemented by diverse entities including service providers, data-driven services, customers, or users, as noted by Zheng et al. (2017). Interaction with the smart product is facilitated through the HMI, enabling users to control the product, alter settings, and view real-time data and product status. Additionally, the HMI serves as a gateway for users to request and access various data-driven services, as outlined by Stecken et al. (2019). It also records data related to user interactions and usage patterns, as described by Meyer et al. (2022).

To optimize the data transmission to the cloud, preprocessing and filtering of the data are performed based on the specific needs of the data-driven services. This step not only ensures data relevance and quality but also reduces the burden on communication infrastructure by minimizing the volume of raw data sent, which is necessary given the high data output of smart products. This selective data handling is tailored to each use case, as Zambetti et al. (2019) emphasize.

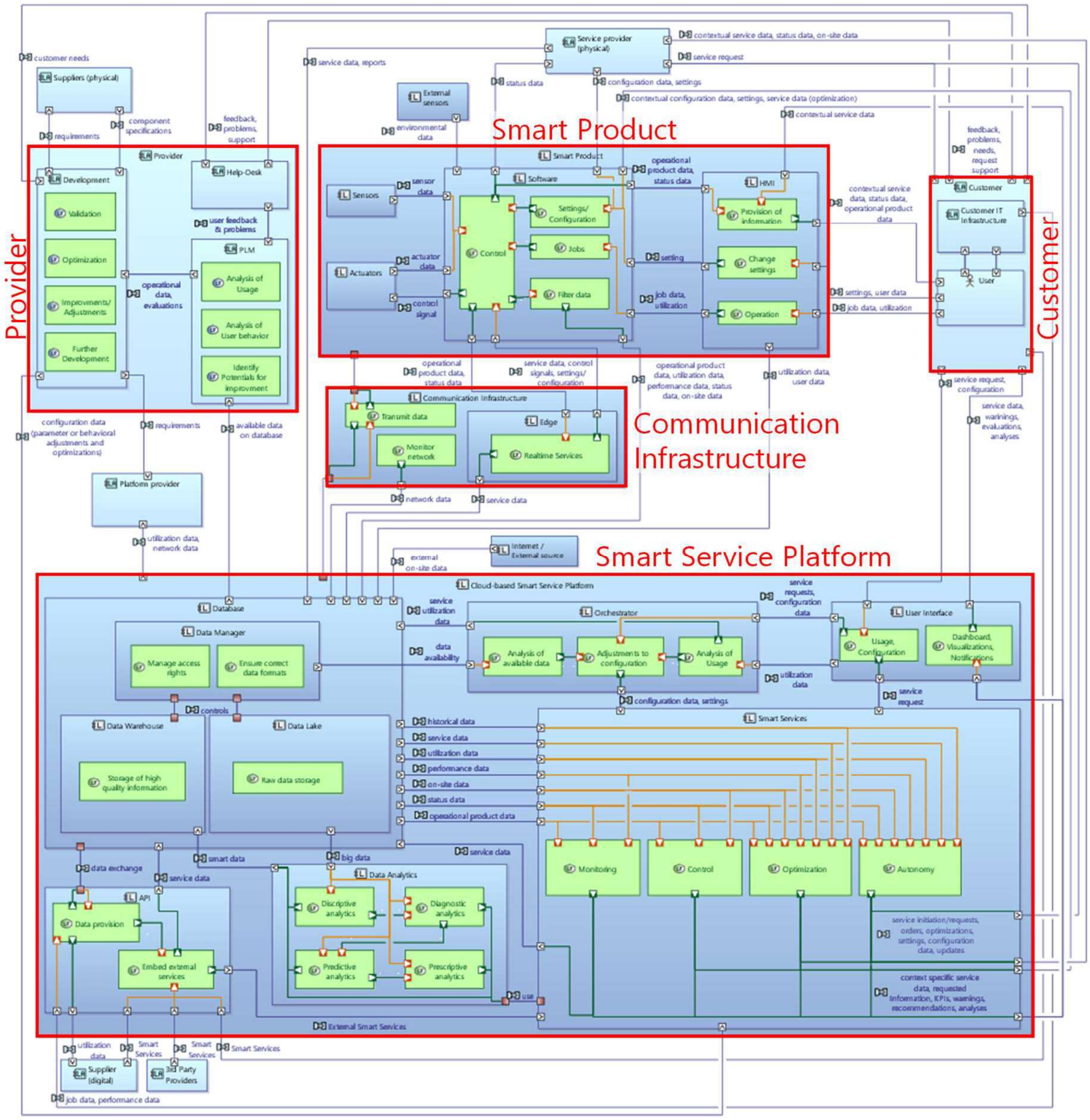

Figure 2. sPSS architecture model

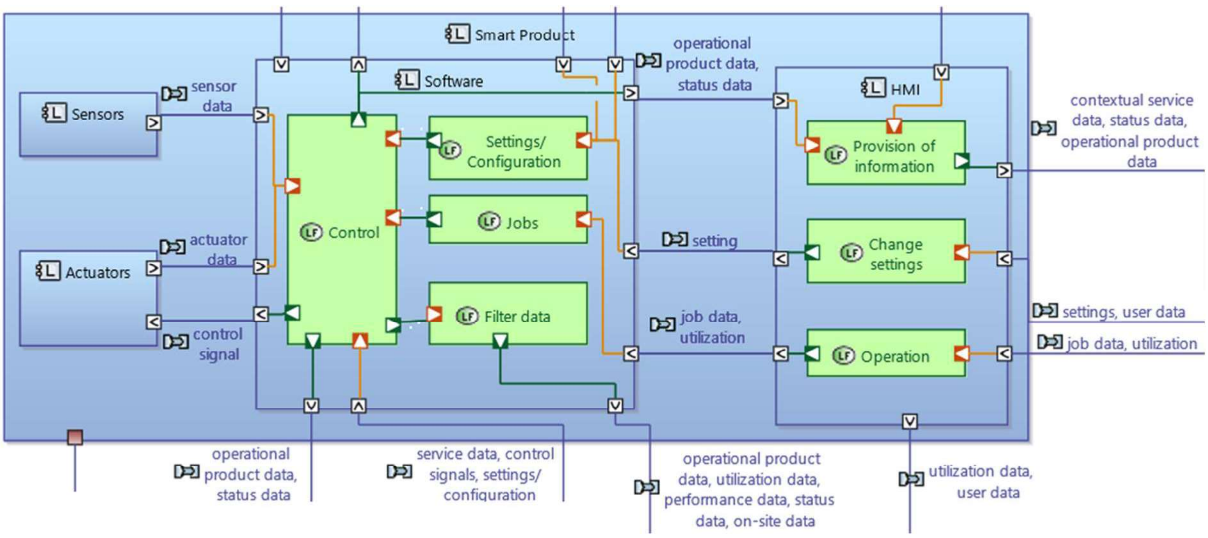

Figure 3. Detail view – smart product

Communication Infrastructure: The communication infrastructure serves a supportive role within the system, focusing primarily on facilitating data transfer among the various elements and subsystems rather than directly contributing to value creation. It is strategically positioned between the smart product and the smart service platform in the model to ensure the functionality of the sPSS. Additionally, the network is capable of handling computationally demanding tasks at the edge, as noted by Al-Ali et al. (2020). This edge computing capability enables the execution of real-time services, such as simulations and optimizations, directly within the network, thus bypassing the need for data transmission to the cloud, as detailed by Cao et al. (2020). Another secondary function of the communication infrastructure is to monitor network traffic and performance. Analyzing this data helps optimize data transfer processes, enhancing the efficiency and effectiveness of the overall system.

Smart Service Platform: The smart service platform primarily delivers data-driven services, connects elements and actors of the sPSS, and manages data storage and analysis. It comprises various segments including the database, data analytics, PSS orchestrator, user interface (UI), application programming interface (API), and the data-driven services themselves as illustrated in Figure 4. The platform, which is cloud-based, is orchestrated both automatically by the orchestrator and manually via the UI. The orchestrator configures the data-driven services based on available data, specific user needs, and system usage, as described by Nebauer et al. (2023). Users interact with the platform through the UI, accessing dashboards and other visual tools for system configuration and setup. Analyzing how the platform is used can reveal customer habits, allowing for service adaptations (Zheng et al., 2017).

Data is collected in the database from system use, primarily originating from the smart product and the customer. This includes both raw and processed data as well as analyses and data-driven services. External data, often contextual, is also incorporated into the platform, stored in either a data lake for unprocessed data (Porter and Heppelmann, 2015) or a data warehouse for processed data, facilitating both immediate use and future analytical needs (Eigner, 2021; Weber et al., 2018). Data management is overseen by a data manager, ensuring access is restricted to authorized individuals, data is in the correct format, and is appropriately used for analysis and services (Paliyenko et al., 2023b; Damjanovic-Behrendt and Behrendt, 2019). The database supports data-driven services by processing various data types into valuable insights or *smart data* from the larger *big data* pool, tailored to specific services and use cases.

While the cloud or platform provider has a minor direct role in the data flows within the sPSS, they offer platforms either standardized or customized to system needs. The platform operator may also gather usage and network data to enhance platform operations (Stecken et al., 2019). Integration of data-driven services by digital suppliers, third-party providers, and customers through the API is facilitated, allowing for the specification of data integration and usage (Zheng et al., 2021; Zambetti et al., 2019; Bulut and Anderl, 2022).

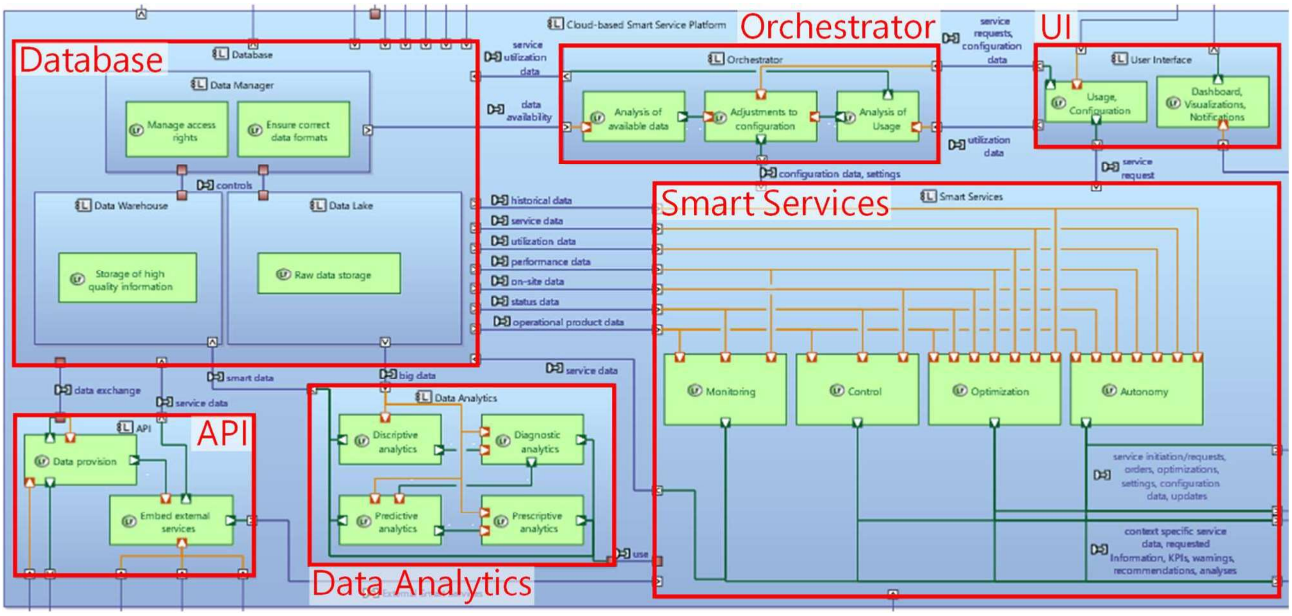

Figure 4. Detail view – smart service platform

Customer: The customer holds a crucial role within the sPSS, not only as a user but also as an influencer of its design. As such, the customer is essential in providing data and information to the sPSS provider to help enhance and refine the system (see Figure 5). Interaction between the customer and the sPSS occurs through the user interface on the Smart Service Platform and the Human-Machine Interface (HMI) of the smart product. Additionally, the customer's information system can be integrated with an API on the Smart Service Platform. Through these interfaces, the customer can access data generated by data-driven services, that provide contextually relevant information, alerts, or recommendations specific to their context. This setup also facilitates the customer's ability to request services or share data back to the sPSS, further contributing to the evolution and customization of the system (Zheng et al., 2017).

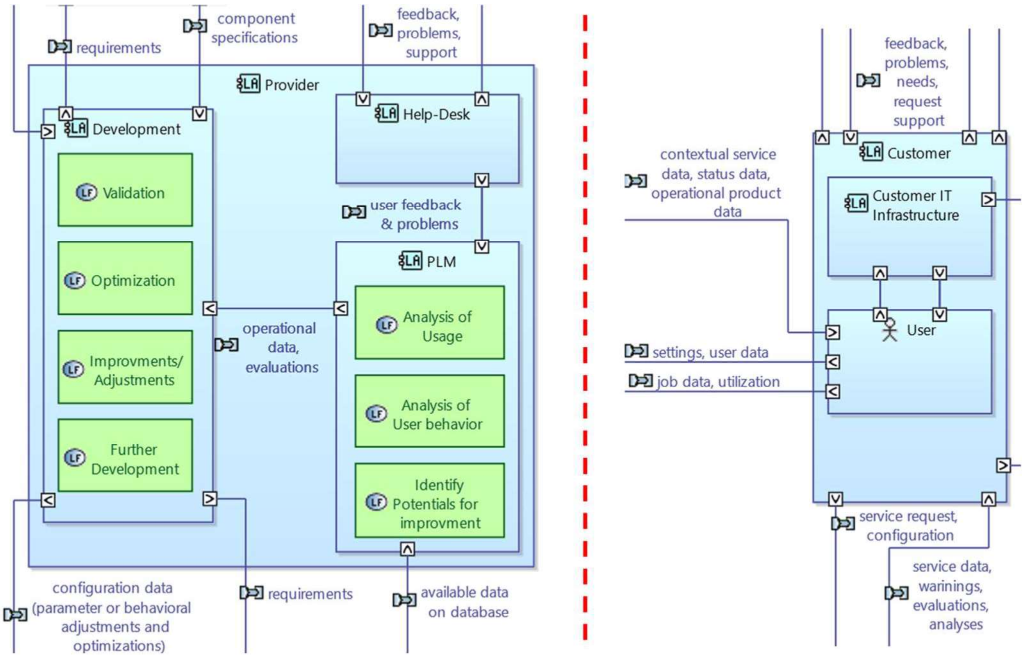

Figure 5. Detail view – provider (left) and customer (right)

Smart PSS Provider: The primary responsibility of the sPSS provider in this data flow model involves incorporating feedback data into ongoing development. The provider collects all operational data from the sPSS, which is then analyzed to generate insights. These insights are crucial for enhancing the sPSS, with improvements continually integrated back into the system to optimize its performance. The model identifies several applications for this usage data (Machchhar et al., 2022). Support for system enhancement is also provided by the Helpdesk, which gathers information from customers about issues and assists in resolving them. This feedback is potentially stored in the Product Lifecycle Management (PLM) system (Paliyenko et al., 2023b). Furthermore, the sPSS provider collaborates with numerous partners, including suppliers and service providers, within the value creation network. The most critical aspect of these partnerships is the sharing of requirements and technical data, facilitating cooperative development of the sPSS.

3.2 Data-based applications in sPSS operations and development

The model maps and structures various data- and information-types from a wide variety of sources. Table 2 provides an organized overview of occurring data-types with descriptions and examples of data sources. For easy of reading, data and information are used synonymously; nevertheless, the authors recognize the difference between the two.

Table 2. Data-types and data-sources in sPSS

Utilization data: order data (ERP/MES)[2,5], function- and service-utility[2], user settings[2,3] user interaction and behavior data[2], operating times[3,4], data on misuse, user data (skills, qualifications)[1,2,3]
Data that occurs when the customer or user utilizes the sPSS. The data describes how and for what purpose the system is used and which settings are made. It is also possible to determine how the user interacts with the system, i.e. which functions are used, how the user behaves and if incorrect use occurs. This data can be put in relation to the user's data.
Operational product data: sensor generated data/product behavior data[2,5,6], actuator data[2,5], control signals, consumption (resources, energy)[1,2,3], emissions[2]
Describes the actual behavior of the product during operation. The data comes primarily from sensors, actuators and the product's control system. This data is generated continuously and in large quantities. Examples are temperature, vibration, pressure, rotational speed, speed, energy consumption and voltage.
Performance data: product quality[2], customer feedback[2,6], product failure[1], KPIs[2]
Data that quantifies the performance of the sPSS. This includes data on the output of the sPSS (quality data) and various key performance indicators (KPIs), that aggregated multiple data sources as KPIs. These can either be determined directly by the sPSS, provided by the customer or derived from other data. Customer feedback in the form of reviews, suggestions for improvement or complaints may also be used. This data can vary greatly in terms of structure and quality and therefore requires more effort to analyze.

Service data: reports (maintenance, repair, service)[1,2,3,4,5,6], smart data[7], service-delivery data
The service data contains all information (details) about the services performed, i.e. physical services: maintenance or repairs. These are often stored in the form of semi-structured reports. Further, this data also includes 'smart data' which is generated during processing and analysis by data-driven services.
Status data: configuration data (hard- & software)[1,2], machine status[2], wear, logs (events, errors)[1,2,3,5]
Data describing the current condition of the product or state of health. This involves data on the wear or stress of components, the operational state, and occurring errors. The configuration data describes how the sPSS is assembled. This includes setting parameters and modules used, removed, or replaced (hardware, software, service). Failure and error data are additional indicators of system performance. This data is particularly important for carrying out reliability analyses and improvements.
On-site data: environmental data[1,2], network-data (ICT-network)[1,2], geo-location[1,2], weather data[1]
On-site data describes the external conditions under which the sPSS operates (ambient conditions, such as temperature or humidity, weather data and location data).
References: [1] Machchhar et al., 2022; [2] Meyer et al., 2022; [3] Abramovici and Lindner, 2011; [4] Deng et al., 2019; [5] Cavalcante and Gzara, 2019; [6] Kanovska and Tomaskova, 2019; [7] Freitag and Wiesner, 2018

The importance of enriching each type of data with metadata cannot be overstated. **Metadata**, which includes elements such as time stamps, data sources, and product serial numbers, plays a crucial role in ensuring that all recorded data can be precisely aligned and integrated (Freitag and Wiesner, 2018). This meticulous approach to data annotation enhances the reliability and usability of data for various applications. Furthermore, the organized storage of data, lays the groundwork for creating a repository of **Historical Data**. Such historical data, encompassing many relevant datasets, forms a valuable database. This database, as highlighted by (Kanovska and Tomaskova, 2019), serves as a foundational asset for conducting future analyses and developing data-driven services. The systematic accumulation and organization of data not only facilitate retrospective studies but also paves the way for innovative service offerings and analytical insights.

The data can be leveraged in data-driven applications to generate value in both sPSS operations (e.g. services) and development (e.g. V&V – Verification and Validation). The applications can be organized according to the types of added value generated by the data-driven services. Similar categorizations for data-based services are often found in the literature (cf. Porter and Heppelmann, 2015). Table 3 maps potential applications with the utilized data-types for these applications.

Table 3. Data-sources and data-demands for operational and developmental applications in sPSS

	Operational applications				**Developmental applications**		
Data type	Monitoring	Control	Optimization	Autonomy	sPSS V&V	Understand user	Requirements V&V
Utilization data			X	X		X	X
Operational product data	X	X	X	X	X		X
Performance data	X		X	X	X		X
Service data			X	X		X	X
Status data	X	X	X	X	X	X	
On-site data		X	X	X			X
Historical data			X	X	X	X	X

Operational applications such as **Monitoring** services offer new insights into the system, making data and information available and provide insights into running operations. This allows the system's condition (Condition Monitoring) to be tracked, or alerts to be signaled when specific events occur. **Control** services provide users with new ways of interacting with the system, with remote services being particularly important. **Optimization** improves the sPSS in various aspects, thereby creating value enhancement in operation. This can take the form of action recommendations or parameter adjustments. **Autonomy** refers to services that improve operation without the need for human interaction. Tasks can also be automated, making the operation more efficient and cost-effective. A selection of more specific uses of data during operation are shown in Table 4. (Porter and Heppelmann, 2015)

Table 4. Operational applications in sPSS

Monitoring	Information provision[3,4,5], condition monitoring[1,2,3,4,5], alerting[1,2], remote diagnosis[5]
Control	Remote control[1,2], remote maintenance[4,5], operational support[3], planning support[2], service support[1,2,5]
Optimization	Predictive maintenance[1,3,4,5], process optimization[1,2,3,4], analytics[2,3], updates and upgrades[1,3,4,5]
Autonomy	Automatic ordering[1,2,3,5], spare parts management[2], maintenance management[4,5], resource management[2], process automation[5]
References: [1] Rabe et al., 2018; [2] Mittag et al., 2018; [3] Schuh et al., 2022; [4] Chowdhury et al., 2018; [5] Heuchert et al., 2020	

Central to data-based services is the use of data to generate additional value. For modeling the data-intensive functions, it's crucial to know which data are needed for which service. This greatly depends on the type and complexity of the service. For example, the services for monitoring and control require significantly fewer data as they describe or influence the state of the sPSS, but do not conduct complex analyses. Services for optimization and autonomy require significantly more data. Potentially, all data from the sPSS can be used here, as there are many different possibilities for optimization and automatization (Hunke et al., 2021). In real applications, data-based services do not always need all these data, and the required quality and quantity of data can vary. For example, predictive maintenance may only require sensor data (such as vibration), quality data, and wear data (Lee et al., 2015).

In addition to operational exploitation, data can be reintegrated back into development to enhance both the system and its development process. For effective data feedback, it is vital to factor in the feedback mechanism during the sPSS's planning and development stages. This ensures data of necessary quality is readily available and that the system is prepared for future evolution and adjustments. Such an approach aids in identifying the required data types, their quality standards, and the essential processing and analysis techniques. It facilitates the preliminary planning for the integration of sensors into the product and highlights areas within the sPSS that need to be designed for flexibility (such as modularity and scalability) to accommodate subsequent modifications and enhancements.

Validating and verifying the sPSS is crucial for understanding its operational behavior and failures, directly impacting the product quality. By analyzing reliability, wear, and stress on components under real-life conditions, designers can create more robust and efficient sPSS. Additionally, error analysis helps pinpoint causes of failure and mitigate their impacts, while service analysis improves efficiency of the service delivery. (Machchhar et al., 2022; Meyer et al., 2022). **Understanding user behavior** by analyzing the interaction with the sPSS informs enhancements in user experience and usability, including adjustments based on user preferences for settings and features. This analysis also identifies areas for improvement in operation and usability (Meyer et al., 2022). Analyzing usage data **validates existing requirements** and identifies new needs based on real-life usage, influencing future sPSS designs to better meet user and service provider needs (Machchhar et al., 2022). Identifying improvement potentials across the value chain is key to improving customer satisfaction and value. This includes increasing efficiency, improving usability, error mitigation, enhancing safety, streamlining processes, and reducing costs. Techniques like correlation analysis and machine learning are essential for uncovering these potentials, offering deep insights into complex datasets (Meyer et al., 2022; Wang et al., 2022).

4 Discussion and outlook

The proposed data-centric architecture model for sPSS offers a foundational framework aimed at enhancing the integration and utility of data throughout the system lifecycle. However, it is critical to acknowledge that the model represents a highly generalized scenario. Many aspects are either overly simplified or aggregated, which could hinder its direct applicability to specific real-world cases without substantial modifications. In particular, the detailing of services and the associated data requirements necessitate significant refinement. This includes a more granular approach to data analysis to adequately address the needs concerning data volume, quality, and format. Furthermore, the modeling of the smart product should incorporate a detailed representation of the various sensors and the diverse data they generate, suggesting the need to specify data types such as temperature or vibration rather than of a generic term like sensor data.

The continuous development and validation of this model, along with addressing the aforementioned open questions, promises substantial contributions to the understanding and optimization of sPSS development. Insights gained from such endeavors can aid the scientific community and industry in designing innovative and efficient sPSS solutions. Moreover, the adoption of MBSE allows for future expansions of the model to include new aspects not initially considered, potentially evolving into a holistic system model that serves as a framework for sPSS development.

As it stands, the model and its schema have not been validated against real-world applications, which is a limitation in confirming its effectiveness. Future research should focus on empirical testing and validation to determine the model's performance and to identify potential improvements. Engaging with industry expert to evaluate the applicability and utility of the model can provide critical insights to further refine and enhance the model's relevance and effectiveness.

The nature of data in sPSS (amount, quality, format, frequency) varies significantly depending on the application context and type of data involved. Such variability has a profound impact on how data analysis and related data-driven services are executed. Additionally, the availability of data might be restricted due to customers' reluctance to share information over networks, further complicated by non-disclosure agreements and regulations concerning data protection and security.

To effectively align the development of sPSS with user needs from the outset, utilizing prototypes with reduced functionality (Minimum Viable Product) can be instrumental. This strategy, supported by early user feedback and initial behavioral insights, facilitates efficient initial improvements and optimizations. Such prototypes, possibly enhanced with additional sensors, can significantly contribute to the early stages of sPSS development, ensuring that the final products are well-tuned to customer demands and expectations. The reintegration of data and insights into product development is a critical aspect of improving sPSS. By systematically feeding operational and user data back into the ongoing development process, companies can significantly refine product functionality, improve service offerings, and better align with customer expectations. This re-integration process involves the utilization of advanced data analytics to extract actionable insights from large datasets. These insights enable designers and engineers to make informed decisions about product updates, feature modifications, and service improvements. Furthermore, incorporating machine learning algorithms can predict trends and user needs, facilitating proactive adjustments to the product lifecycle.

Acknowledgments

This work has been supported by the German Federal Ministry of Education and Research through the research project “bi.smart” (grant no. 02J19B043).

References

Abramovici, M., 2018. Engineering smarter Produkte und Services Plattform Industrie 4.0 Studie. acatech – Deutsche Akademie der Technikwissenschaften, Munich.

Abramovici, M., Lindner, A., 2011. Providing product use knowledge for the design of improved product generations. CIRP Annals - Manufacturing Technology 60, 211-214.

Al-Ali, A.R., Gupta, R., Batool, T.Z., Landolsi, T., Aloul, F., Al Nabulsi, A., 2020. Digital Twin Conceptual Model within the Context of Internet of Things. Future Internet 12, 1-15.

Apostolov, H., Fischer, M., Olivotti, D., Dreyer, S., Breitner, M.H., Eigner, M., 2018. Modeling Framework for Integrated, Model-based Development of Product-Service Systems. 10th CIRP Conference on Industrial Product-Service Systems, 9-14.

Bastos, C.A.M., Moreira, M.R., Bruno, A.C.M., Filho, S.M., de Farias Filho, J.R., 2014. Information Flow Modeling. A Tool to Support the Integrated Management of Information and Knowledge. Proceedings of the International Conference on Knowledge Management and Information Sharing, 76-86.

Bulut, S., Anderl, R., 2022. Towards Ecosystems with Smart Product-Service Systems. 32nd CIRP Conference 109, 221-226.

Cao, K., Liu, Y., Meng, G., Sun, Q., 2020. An Overview on Edge Computing Research. IEEE Access 8, 85714-85728.

Cavalcante, J., Gzara, L., 2019. Product-Service Systems Lifecycle Management in Industry: Interests and Exploited Data, in: Product Lifecycle Management in the Digital Twin Era. Springer, Cham, 389-398.

Chowdhury, S., Haftor, D., Pashkevich, N., 2018. Smart Product-Service Systems (Smart PSS) in Industrial Firms: A Literature Review. 10th CIRP Conference on Industrial Product-Service Systems 73, 26-31.

Damjanovic-Behrendt, V., Behrendt, W., 2019. An open source approach to the design and implementation of Digital Twins for Smart Manufacturing. International Journal of Computer Integrated Manufacturing 32, 366-384.

Deng, Q., Wellsandt, S., Hribernik, K.A., Thoben, K.D., 2019. Towards Understanding the Role of Product Usage Information in Product Design Improvement, in: Fortin, C., Rivest, L., Bernard, A., Bourad, A. (Eds.), Product Lifecycle Management in the Digital Twin Era. Springer, Cham, pp. 369-378.

Durugbo, C.M., Tiwarl, A., Hutabarat, W., Alcock, J.R., 2012. Information channel diagrams: An approach for modelling information flows. Journal of Intelligent Manufacturing, 22, 1-13.

Echterhoff, B., 2018. Methodik zur Einführung innovativer Geschäftsmodelle in etablierten Unternehmen. Heinx Nixdorf Institut, Paderborn.

Eigner, M., 2021. System Lifecycle Management – Engineering Digitalization (Engineering 4.0). Springer, Wiesbaden.

Frank, U., van Laak, B.L., 2003. Anforderungen an Sprachen zur Modellierung von Geschäftsprozessen. Universität Koblenz-Landau.

Freitag, M., Wiesner, S., 2018. Smart Service Lifecycle Management: A Framework and Use Case, in: Moon, I., Lee, G.M., Park, J., Kiritsis, D., Cieminski, G. (Eds.), Advances in Product Management Systems. Springer, Cham, pp. 97-104.

Gronau, N., Bahrs, J., Hake, M., Heinze, P., Lembcke, R., Scharff, C., Vladova, G., 2010. Wissensorientierte Modellierung im Lebenszyklus von Dienstleistungen, in: Dienstleistungsmodellierung 2010. Physica-Verlag, Heidelberg, pp. 3-23.

Halstenberg, F.A., 2022. Methodik zur modellbasierten Systemarchitekturdefinition von Smart-circular Product-Service Systems. Universität Berlin, Berlin.

Heuchert, M., Verhoeven, Y., Cordes, A.K., Becker, J., 2020. Smart Service Systems in Manufacturing: An Investigation of Theory and Practice. Proceedings of the 53rd Hawaii International Conference on System Sciences, 1686-1695.

Hunke, F., Heinz, D., Satzger, G., 2021. Creating customer value from data: foundations and archetypes of analytics-based services. Electron Markets 32, 503-521.

Idrissi, N.A., Boucher, X., Medini, K., 2017. Generic conceptual model to support PSS design processes. 9th CIRP IPSS Conference: Circular Perspectives on Product/Service-Systems 64, 235-240.

Kanovska, L., Tomaskova, E., 2019. Data Gained from Smart Services in SMEs – Pilot Study, in: Silhavy, R., Silhavy, P., Prokopova, Z. (Eds.), Computational and Statistical Methods in Intelligent Systems. Springer, Cham, pp. 183-200.

Keller, A., Binz, H., 2009. Requirements on engineering design methodologies. International Conference on Engineering Design 2009, Standford, USA, 2203 – 2214.

Lee, J., Ardakani, H.D., Yang, S., Bagheri, B., 2015. Industrial big data analytics and cyber-physical systems for future maintenance & service innovation. The Fourth International Conference on Through-life Engineering Services 38, 3-7.

Lenarduzzi, V., Taibi, D., 2016. MVP Explained: A Systematic Mapping Study on the Definitions of Minimal Viable Product. 42th Euromicro Conference on Software Engineering and Advanced Applications, 112-119.

Lindemann, M., Nuy, L., Briele, K., Schmitt, R., 2019. Methodical data-driven Integration of perceived quality into the Product Development Process. 29th CIRP Design 84, 406-411.

Liu, Z., Ming, X., Song, W., Qiu, S., Qu, Y., 2018. A perspective on value co-creation-oriented framework for smart product-service system. 10th CIRP Conference on Industrial Product-Service Systems 73, 155-160.

Machchhar, R.J., Toller, C.N.K., Bertoni, A., Bertoni, M., 2022. Data-driven value creation in Smart Product-Service System design: State-of-the-art and research directions. Computers in Industry 137, 1-21.
Meyer, M., Panzner, M., Koldewey, C., Dumitrescu, R., 2022. 17 Use Cases for Analyzing Use Phase Data in Product Planning of Manufacturing Companies. Procedia CIRP 107, 1053-1058.
Mittag, T., Rabe, M., Gradert, T., Kühn, A., Dumitrescu, R., 2018. Building blocks for planning and implementation of smart services based on existing products. 10th CIRP Conference on Industrial Product-Service Systems 73, 102-107.
Nebauer, S., Schneider, M., Schöllhammer, O., Rauh, L., Schel, D., Bauernhansl, T., Evcenko, D., Kett, H., Wirth, J., Bauer, K., Sawilla, I., Höding, M., 2023. Serviceorientierte Produktion der Zukunft. Fraunhofer IPA, Stuttgart.
Olivotti, D., Dreyer, S., Kölsch, P., Herder, C.F., Breitner, M.H., Aurich, J.C., 2018. Realizing availability-oriented business models in the capital goods industry. 10th CIRP Conference on Industrial Product-Service Systems 73, 297-303.
OMG, 2011. Business Process Model and Notation (BPMN), Object Management Group, Milford.
Orellano, M., Medini, K., Lambey-Checchin, C., Neubert, G., 2019. A system modelling approach to collaborative PSS design. 11th CIRP Conference on Industrial Product-Service Systems 83, 218-223.
Otto, B., Hompel, M., Wrobel, S., 2018. Industrial Data Space - Referenzarchitektur für die Digitalisierung der Wirtschaft, in: Neugebauer, R. (Ed.), Digitalisierung – Schlüsseltechnologien für Wirtschaft & Gesellschaft. Springer, München, pp. 113-133.
Paliyenko, Y., Heinz, D., Schiller, C., Tüzün, G.J., Roth, D., Kreimeyer, M., 2023a. Requirements for a smart Product-Service System Development Framework. ICED23, 3085-3094.
Paliyenko, Y., Salinas, R., Roth, D., Kreimeyer, M., 2023b. Vorgehen zur Modellierung des Wertschöpfungsnetzwerks smarter Produkt-Service-Systeme. 34. DfX-Symposium, 1-10.
Paliyenko, Y., Tüzün, G.J., Roth, D., Kreimeyer, M., 2022. Inquiry and Analysis of Challenges in the Development of Smart Product-Service Systems. Proceedings of the Design Society 2, 1935-1944.
Porter, M.E., Heppelmann, J.E., 2014. How Smart, Connected Products Are Transforming Competition. Harvard Business Review.
Porter, M.E., Heppelmann, J.E., 2015. How Smart, Connected Products Are Transforming Companies. Harvard Business Review.
Rabe, M., Asmar, L., Kühn, A., Dumitrescu, R., 2018. Planning of Smart Services based on a Reference Architecture. International Design Conference 15, 2949-2960.
Rizvi, M.A.K., Chew, E., 2018. Towards Systematic Design of Cyber-Physical Product-Service Systems. International Design Conference 15, 2961-2974.
Schuh, G., Stroh, M.F., Hicking, J., Kremer, S., 2022. Smart Products For Smart Production - A Use Case Overview. 3rd Conference on Production Systems and Logistics, 504-514.
Stapel, K., Schneider, K., 2012. Methodenbeschreibung zur Anwendung von FLOW. Leibniz Universität Hannover, Hannover.
Stecken, J., Ebel, M., Bartelt, M., Poeppelbuss, J., Kuhlenkötter, B., 2019. Digital Shadow Platform as an Innovative Business Model. 11th CIRP Conference on Industrial Product-Service Systems, 204-209.
Tangkawarow, I.R.H.T., Waworuntu, J., 2016. A Comparative of business process modelling techniques. IOP Conference Series: Materials Science and Engineering 128, 012010.
Thuan, N.H., Swann, D., Chiu, Y.T., Antunes, P., 2017. Understanding and Modelling Organisational Information Flows. IEEE 21st International Conference on Computer Supported Cooperative Work in Design, 85-90.
Valencia, A., Mugge, R., Schoormans, J.P.L., Schifferstein, H.N.J., 2015. The Design of Smart Product-Service Systems (PSSs): An Exploration of Design Characteristics. Internal Journal of Design, 13-28.
VDI/VDE 2206, 2021. Entwicklung mechatronischer und cyber-physischer Systeme. Verein Deutscher Ingenieure, Düsseldorf.
Voirin, J.L., 2018. Model-based System and Architecture Engineering with the Arcadia Method. ISTE Press and Elsevier, London.
Wang, Z., Zheng, P., Li, X., Chen, C.H., 2022. Implications of data-driven product design: From information age towards intelligence age. Advanced Engineering Informatics 54, 1-28.
Weber, C., Wielang, M., Reimann, P., 2018. Konzepte zur Datenverarbeitung in Referenzarchitekturen für Industrie 4.0. Datenbank Spektrum 18, 39-50.
Weilkiens, T., 2014. Systems Engineering mit SysML/UML. dpunkt.verlag GmbH, Heidelberg.
Wu, C., Chen, T., Li, Z., Liu, W., 2021. A function-oriented optimising approach for smart product service systems at the conceptual design stage: A perspective from the digital twin framework. Journal of Cleaner Production 297, 1-19.
Zambetti, M., Pinto, R., Pezzotta, G., 2019. Data lifecycle and technology-based opportunities in new Product Service System offering towards a multidimensional framework. 11th CIRP Conference on Industrial Product-Service Systems 83, 163-169.
Zheng, M., Ming, X., Wang, L., Yin, D., Zhang, X., 2017. Status review and future perspectives on the framework of Smart Product Service Ecosystem. 9th CIRP IPSS Conference: Circular Perspectives on Product/Service-Systems 64, 181 – 186.
Zheng, P., Chen, C.H., Wang, Z., 2021. Smart Product-Service Systems. Elsevier, Amsterdam.