

# Software Product Structure in the context of Service-Dominant (S-D) Logic - A Scoping Review

Akwasi G. Adusei<sup>1</sup>, Janne Härkönen<sup>2</sup>

<sup>1</sup> University of Oulu, Industrial Engineering and Management, Oulu, Finland

<sup>2</sup> University of Oulu, Industrial Engineering and Management, Oulu, Finland

## Abstract

This scoping review paper aims to identify and synthesize the existing literature on software product structure in the context of service-dominant (S-D) logic. The paper highlights the importance of software product structure as a critical aspect in the software industry especially in software development as it can help to ensure that the software product meets the needs of all stakeholders. Despite the growing importance of software product structure and its perceived insight in the context of S-D logic, few studies have explored this topic. This paper, therefore, seeks to fill this gap by examining the findings of studies that have explored this topic. The review aims to provide a comprehensive overview of the current state of knowledge in this area and identify gaps that require further investigation. The findings of this study will be beneficial for software developers, service providers, and customers to co-create value effectively and ensure that the software product meets the needs of all stakeholders. Overall, this scoping review aims to contribute to the understanding of software product structure in the context of S-D logic and provide insights that can help software developers, service providers, and customers create value effectively.

## Keywords

Software product structure, Software productization, Software product, Service-dominant logic

## 1. Introduction

Software products have become an integral part of the modern digital economy [1]. Consequently, the concept of software product structure has gained significant attention in recent years [1];[2]. However, there has also been a growing discussion and application of new service dominant (S-D) logic thinking in service science research since its introduction in 2004 by Vargo and Lusch [3]. S-D logic emphasizes the importance of service exchange processes and co-creation among stakeholders [4].

Given the complexity of the software development process and the involvement of multiple stakeholders including developers, customers, and end-users, software product structure is a critical aspect that can help ensure the software product meets the needs of all

stakeholders. As such, understanding software product structure from an S-D logic perspective is essential for software developers, service providers, and customers. Despite the growing importance of software product structure and its perceived importance in the context of S-D logic, few studies have explored this topic, and those that have are often limited in scope, methodology, and context.

To address this gap in the literature, this scoping review aims to identify and synthesize the existing literature on software product structure in the context of S-D logic. The review will provide insights into what is currently known about software product structure in the context of S-D logic by identifying the key factors that influence software product structure design and their implications for service providers and customers. The research question that guides this scoping

TKTP 2023: Annual Symposium for Computer Science 2023, June 13-14, 2023, Oulu, Finland)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

review is, "What is known from literature about software product structure in the context of service-dominant logic?"

The findings of this study will be beneficial for software developers, service providers, and customers to co-create value effectively and ensure that the software product meets the needs of all stakeholders. The review will examine the research aims, methodologies, and findings of the studies that have explored this topic. By doing so, the review will provide a comprehensive overview of the current state of knowledge in this area and identify gaps that require further investigation.

Overall, this scoping review aims to contribute to the understanding of software product structure in the context of S-D logic and provide insights that can help software developers, service providers, and customers create value effectively.

## 2. Key Concepts

This section presents an overview of key concepts in the literature related to software product structure from the perspective of Service-Dominant (S-D) Logic.

### 2.1. Software productization

Software product sector has in recent years become the most rapidly growing industry, becoming a significant economic contributor around the world [1]. In recent years, increasing attention has been paid to software productization due to its importance in bringing software products to market [1]. As a result, most customer-driven organizations have trended towards standardizing software as a product for market purchase. The digital nature of software platforms is in line with digitalization, allowing for new ways of value co-creation [5].

Software products are intangible products that consist of different computer programmes, procedures, documents as well as delivery of data to users and other offerings [6]. According to [2], software products are packaged configurations of software components or software-based services with auxiliary materials released for trade in a specific market. However, when it comes to productization of software products, various authors view it differently. In the software development context, the concept of productization is related to the product release process that involves all stakeholders [7]. [1]

view productization in the software perspective as a process of turning customer specific or customised software to a standardised software product.

Software productization in general enables the understanding of software offering as well as management of the software product portfolios by systematising, tangibilising and formalising the software product [8]. Standardization of service offerings through productization allows for the problems such as heterogeneity that face the software service industry to be addressed [9]. Productization through product structure facilitates the organisation of software products in a manner that is logical and related for customer product configuration. [8]. However, some companies struggle to optimize their product structure to be compatible with data systems, leading to conflicts with their existing product portfolio [2].

[10] emphasised the need for traditional product sales of software to move to Software as a Service (SaaS). The author stated that software life cycle has shown that initially, companies made a lot of revenue from software products but gradually shifted to a mixture of software products and SaaS. SaaS managers should productize SaaS product such that they can be delivered effectively to customers. He then explained the possible productization of service as having a standardized process framework, design reuse and computer aided design tools. [8] proposed a product structure model to facilitate the identification and clarification of different SaaS packages, which can often be complex and difficult for customers and companies to differentiate. Product structure is known to bring a logical understanding of every product through standardization approach to clarify the product offering to all stakeholders. [8]; [11]. It presents different views of the product to different business departments such as sales, research and development, product development, etc in the company. [11]. The product structure defines the list of product items that makes it possible for customers to configure standardised product to their choice.[11]

Despite the increasing concept of productization by researcher in general, few studies can be found on productization as it relates to the software industry. Specifically, insufficient study can be found in relation to how companies can convert their software packages into a standardised software product. [2].

## 2.2. Service-Dominant logic

Until 2014, the traditional exchange of physical products and services popularly known as the goods-dominant (G-D) logic was the trend and logic of the marketing economy [4], [12]–[14]. However, in the past two decades, there has been a paradigm shift from the traditional G-D logic of marketing to a new dominant logic called the Service dominant (S-D) logic, fundamentally since its first publication [3]. The S-D logic is defined as “the application of specialized competences such as knowledge and skills through deeds, processes and performances” [4], [15].

In the new S-D logic, intangibility, exchange process and relationship have been the central pillars. [4]. This S-D logic thinking highlights that customers do not in actual sense buy goods or services as in the traditional G-D logic approach in marketing, rather customers buy offerings which produce value. In addition, the S-D logic postulates that service is exchanged for service and value is co-created with the customer. [14], [16]. This service centered thinking presents a philosophy that is relevant to the marketing offering including even tangible products in the provision of the service process [4]. [16] emphasised that the relationship between customers and producers should be longitudinal, interactive, and dynamic as well. This is because, the S-D logic suggests that the co-creation between customers and producers are is such that firms makes a value proposition whilst the customers determine the value through the usage or experience. [16]

The S-D logic was developed on 8 foundational premises in 2004 and has progressed into 11 foundational premises (Table 1). These foundational premises provide the basics on which the service dominant logic can be understood. [4], [17].

In the value formation process, there is a need for all stakeholders to understand operand and operant resources. The former are those resources on which an operation or an act is performed to produce an effect, whilst the latter are those employed to create value for the operand resources. [4]. Operant resources are especially used to provide benefits to other actors, and therefore in the service dominant logic, service is exchanged for service among the actors. [14], [18]. All customers, suppliers, employees can thus

be categorised under the bracket of operant resources. [19]

Many researchers from different fields have discussed the topic of the S-D logic widely and attempted to apply this new logic to their different research areas. [3]. For instance, Figure 1 and 2 shows the number and trend of types of publications on S-D logic per year.

Again, [18] reported that Vargo and Lusch together with other authors have for the past 17 years of existence of the service dominant logic produced 146 publications on the S-D logic alone.

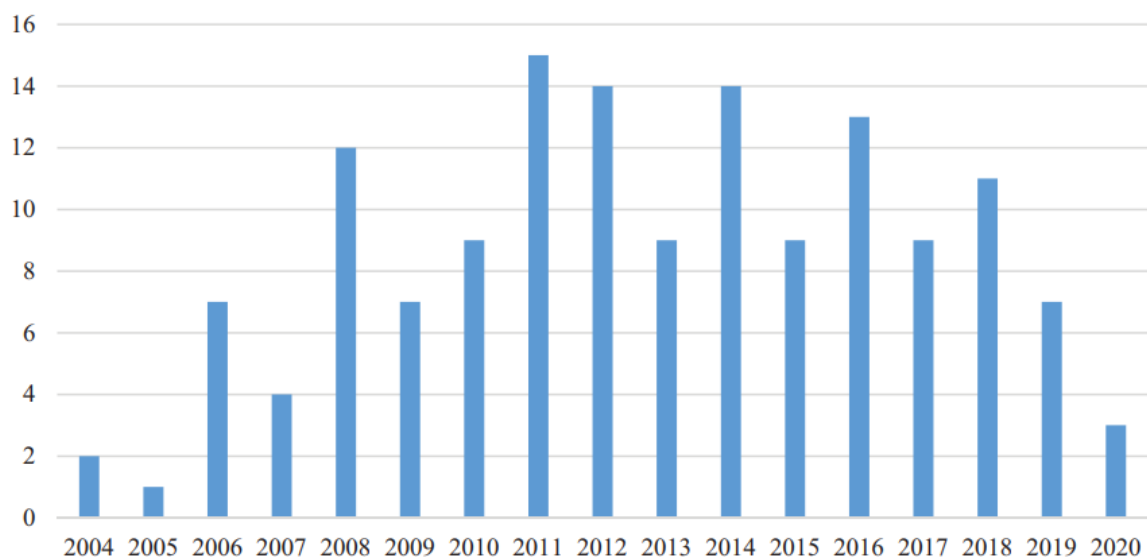
Specifically, [13] discussed sustainable tourism in the light of S-D logic by examining value co-creation which is one of the foundational premises of S-D logic perspective. The author’s study contributed to the S-D logic discussion by introducing features that are sustainable enough to facilitate value co creation within the tourism offering, for example improving well-being.

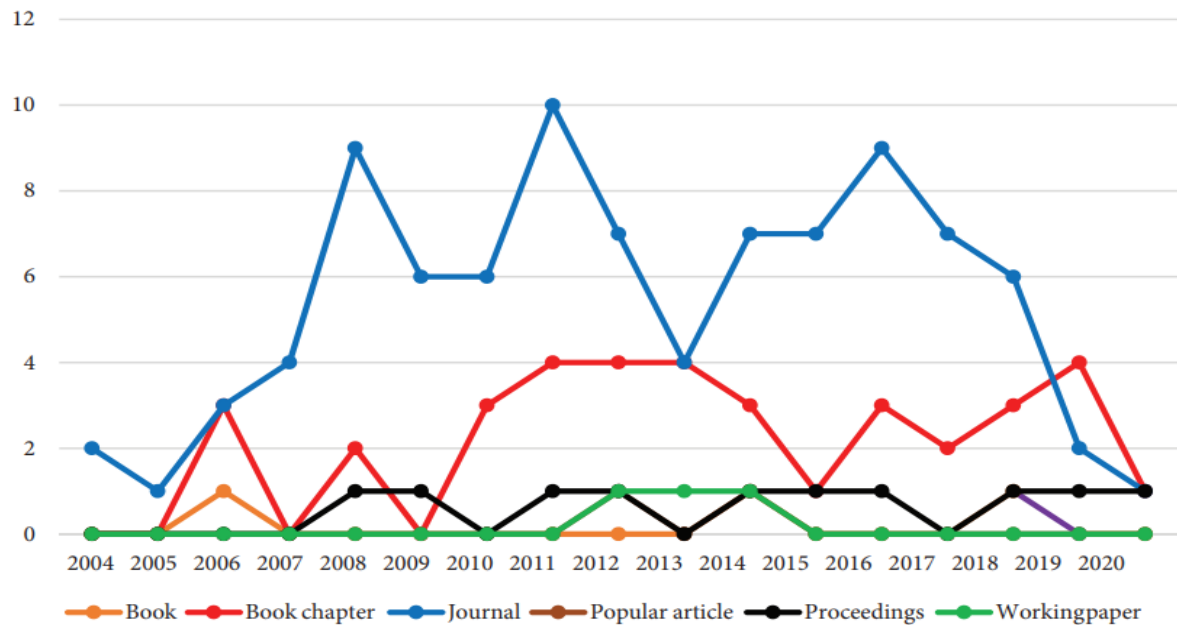
From the software perspective, [12] is one of the few studies that examined the ways by which SaaS releases could be improved by applying S-D logic principles. This study revealed the knowledge concerning the value co-creation that should exist between SaaS suppliers and their customers by improving and fine-tuning relationships and communications between them. Examples of other research areas that have applied the S-D logic thinking include Medicine [13], [20], [21], education [22], governance [23], supply chain [24], manufacturing [25], and Energy [26]. This shows how vast the new service dominant logic has since then been discussed. However, throughout the literature search, no study seems to have been conducted on software productization and product structure through the S-D logic perspective.

**Table 1**

Premises of service-dominant logic (source: adapted from [17])

Number	Premise
1	Service is the foundation of social and economic exchange
2	Indirect service masks the foundation of social and economic exchange
3	Tangible products are vehicles for service delivery
4	Intangible and dynamic resources are the basic origin of reciprocal benefit
5	All economies (with or without tangible products) are service economies, namely direct and indirect service
6	Value is co-created by multi-actors, such as producer, consumer, supplier, and other actors
7	Actors cannot convey value but can create value propositions
8	A service-focused mindset is essentially beneficiary oriented and interactional.
9	All social and economic actors integrate public, private, and market-facing resources
10	Value is individually decided by the beneficiary, such as producer, consumer, supplier, and other actors
11	Co-creation of value is arranged through institutions (norms, rules, values, rule of the game, belief, cognitive models) and institutional logics.

**Figure 1:** Number of articles published per year (source: [18])



**Figure 2:** Types of publication per year (source: [18])

### 3. Analysis Framework

This chapter discuss the methodology employed for this study which aimed to explore the literature on software productization and product structure in the context of Service-Dominant (S-D) Logic. To achieve this objective, a scoping review was chosen as the research methodology. A Scoping review is a type of systematic review that aims to map the existing literature on a particular topic, identify gaps in the literature and provide an overview of the available evidence [27]. Figure 3 shows the modified five stage scoping process that was utilized for this study [27]

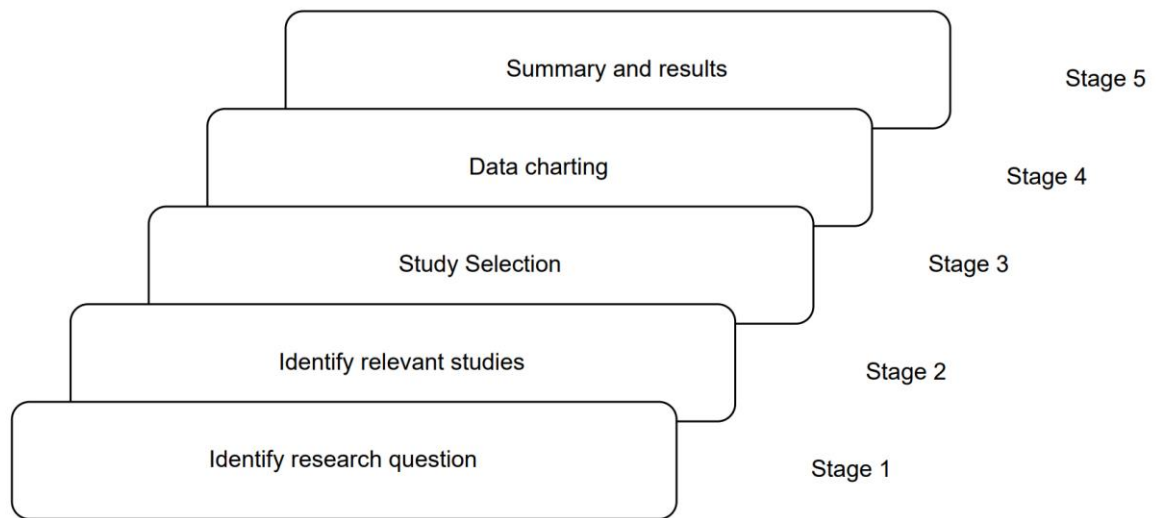
Firstly, our research question “what is known from literature about software product structure from the contest of service dominant logic” was identified. Guided by the key concepts of the study and our research questions, we compiled key words that could be utilized for the literature search from the chosen data base. These key words are “software product structure”, “software productization”, “software product”, and “service dominant logic”.

In the second stage, key words in relation to software productization and S-D logic were combined to conduct a database search. The search for the related studies was conducted mainly in Scopus database. To increase the scope of search, a Google Scholar was also utilized to get some selected and hand-picked research

papers after considering all overlapping papers. The search was limited to research articles that were published in English between 2015 and 2023. All types of studies including qualitative and quantitative research papers were considered. In total, 829 initial research papers were identified.

In the study selection stage, the inclusion criteria were applied such that, each article should be related to at least S-D logic, software productization or product structure, and must provide some evidence in relation to the main theme of software productization and S-D logic. This was accomplished by first reading through the titles and abstracts and finally a full text reading. Altogether, 358 documents were assessed and identified to have covered the study context to an extent after reading through the title and abstract. However, after the full text assessment, 56 articles were identified to fit the study criteria.

In the fourth stage, a predefined data extraction form was utilized to chart the data. These extracted data, include the study aim, key findings, and results. The extracted data were synthesised to study the pattern of S-D logic in relation to software productization and product structure. Finally, the findings were presented in a descriptive summary, highlighting patterns and themes and gaps in the literature related to software product structure in the S-D logic context.



**Figure 3:** Scoping review stages (modified from Arksey & O'Malley, 2005)

## 4. Results and Discussion

This chapter presents the findings and discussions from the analysis of literature on software product structure in the context of S-D logic utilizing a scoping review methodology. In order to provide a more comprehensive understanding of the subject matter, the study have identified five key themes that emerged from the selected literature. These themes include the Service-Dominant Logic (S-D Logic) Perspective, Value Co-Creation, Service-Oriented Product Development, Co-Creation of Knowledge, and Value Co-Creation Networks. The purpose of organizing the findings into these themes is to highlight the common patterns that exist in the literature and to present a more coherent picture of the state of research in this area. By doing so, the study aims to contribute to the ongoing discourse on software product structure through the lens of S-D logic and to provide insights that can inform future research efforts.

### 4.1. Service-Dominant logic (S-D Logic) perspective

Software product structure has become an important area of research in the context of Service-Dominant Logic (S-D Logic) perspective. One of the key themes that emerged from the

review is the importance of applying S-D Logic to the design of service-dominant business models for digital innovation in software product development [28], [29]. These studies highlight the need for software companies to shift from a product-centric approach to a service-centric approach that emphasizes value co-creation with customers. This involves understanding customer needs, co-designing solutions with them, and continuously iterating and improving the product based on customer feedback.

Another important theme that emerged from the review is the role of institutions and technology in the value co-creation process of restaurant consumers [28]. This study highlights how the adoption of S-D Logic can help restaurant businesses to create value by leveraging institutional and technological resources to co-create experiences with customers.

The adoption of S-D Logic can also help companies to achieve radical service innovation in manufacturing, by focusing on value co-creation with customers and other stakeholders [25]. This involves identifying and leveraging new sources of value, such as customer data, and integrating them into the product development process.

Applying S-D Logic to translation service provision can help service providers to better understand and meet customer needs, by co-creating solutions with [29]. This involves understanding the customer's context, preferences, and expectations, and designing solutions that are tailored to their needs.

The adoption of S-D Logic can also help to address the challenges of energy transitions, by providing an integrative framework for assessing the impact of new technologies and policies on value co-creation with [26]. This involves understanding the needs and preferences of energy consumers and designing solutions that are aligned with their values and goals.

Finally, applying S-D Logic to smart city development can help to create sustainable and livable cities that provide value to all stakeholders [30]. This involves understanding the needs and preferences of different stakeholders and designing solutions that are tailored to their needs and preferences.

In conclusion, the adoption of S-D Logic can provide a useful framework for understanding and designing software product structures that are focused on value co-creation with customers and other stakeholders.

## 4.2. Value Co-creation

Value co-creation is an important aspect of service-dominant logic (S-D logic), which emphasizes the importance of collaboration between service providers and customers in value creation. In recent years, there has been an increasing interest in exploring the concept of value co-creation across various industries, including the software industry. In this scoping review, we examine the literature on software product structure in the context of service-dominant logic and its relation to value co-creation.

One of the key subject that emerged from the literature is the importance of aligning resources, actors, and contexts for value creation. [31] argue that knowledge management can play a critical role in this alignment process, by facilitating the sharing of knowledge and expertise between different actors involved in value co-creation. This is particularly relevant in the context of software development, where different stakeholders, including developers, designers, and users, need to collaborate and share information to create value.

Also, [32] discussion on the role of customer participation in value co-creation suggest that customers should be seen as active participants in the value creation process, rather than passive recipients of value. This is especially relevant in the context of software, where users often play an important role in providing feedback and

suggestions for improving the software product. Similarly, [32] demonstrate how digital platforms like Airbnb can facilitate value co-creation by providing users with opportunities to share their experiences and insights with others.

A related context that emerged from the literature is the importance of brand communities in value co-creation [33] develop a brand community engagement model that emphasizes the importance of building strong relationships with customers and fostering a sense of community among users. This is particularly relevant in the software industry, where many software products have dedicated user communities that provide support and feedback to developers.

Finally, the literature also highlights the importance of innovation in value co-creation. [34] argue that innovation is a critical driver of value creation in service industries, and that firms need to continually innovate and adapt to changing customer needs and preferences. This is particularly relevant in the software industry, where rapid technological advancements and changing user expectations require firms to continually innovate and improve their products.

Overall, the findings from this scoping review suggest that value co-creation is a critical aspect of software industry in the context of service-dominant logic. By aligning resources, actors, and contexts, engaging customers in the value creation process, leveraging digital platforms and brand communities, and fostering a culture of innovation, firms can create software products that provide value to both the firm and its customers.

In conclusion, this review sheds light on the importance of value co-creation in the software industry and its relation to service-dominant logic. By understanding and leveraging the principles of S-D logic, software firms can create products that better meet the needs and preferences of their customers, while also driving innovation and growth.

## 4.3. Service-Oriented product development

The software product development process has undergone significant changes in recent years with the increasing emphasis on service-oriented product development. This approach focuses on creating software products that offer services to the customers, rather than just selling a product.

Service-dominant (S-D) logic provides a theoretical framework for understanding the value co-creation process in service-oriented product development.

The scoping review of software product structure in the context of service-dominant logic examines the various design aspects of service-dominant business models in software development. The evaluation of the design of service-dominant business models is crucial to the success of the software product development process. The study by [35] provides a qualitative method for evaluating the design of service-dominant business models. The study emphasizes the importance of understanding the customer's value proposition and how the software product can deliver value to the customers.

The design principles for digital value co-creation networks in service-dominant logic are essential in software product development. The study by [36] proposes five design principles for digital value co-creation networks: (1) modularity, (2) openness, (3) connectivity, (4) heterogeneity, and (5) dynamicity. These design principles enable software developers to create software products that offer services to the customers in a collaborative manner.

The service-dominant business model design for digital innovation in smart mobility is another critical aspect of software product development. The study by [37] proposes a service-dominant business model design for digital innovation in smart mobility. The study emphasizes the importance of understanding the value proposition of the customers and how the software product can deliver value in the context of smart mobility.

The service-dominant logic perspective also provides insight into the paths towards radical service innovation in manufacturing companies. The study by [25] highlights the importance of understanding the customer's value proposition and the co-creation of value with the customers in service-oriented product development. The study emphasizes the need for a shift from a product-dominant logic to a service-dominant logic perspective in the software product development process.

In conclusion, the theme of service-oriented product development is essential in the software product development process. The scoping review of software product structure in the context of service-dominant logic provides insights into the various design aspects of service-dominant business models in software development. The

studies by [25], [35]–[37] highlight the importance of understanding the customer's value proposition and the co-creation of value with the customers in service-oriented product development. The service-dominant logic perspective offers a theoretical framework for understanding the value co-creation process in service-oriented product development, and software developers can utilize this framework to create software products that offer services to the customers.

#### **4.4. Co-creation of knowledge**

The concept of co-creation of knowledge is an important theme in the context of service-dominant (S-D) logic, which emphasizes the importance of collaboration between service providers and customers in the creation of value. In this scoping review on software product structure in the context of S-D logic, several articles shed light on the importance of co-creation of knowledge in software development.

One article that highlights the significance of bringing the individual into the co-creation of value is [38] study on individual involvement in value co-creation. The study emphasizes the role of customer participation in the co-creation process, which allows software developers to gain valuable insights into customer needs and preferences. This, in turn, helps software developers to design and develop software products that better meet customer requirements.

Similarly, [31] discuss the alignment of resources, actors, and contexts for value creation, which emphasizes the importance of knowledge management in service-dominant logic. The authors argue that knowledge management plays a crucial role in facilitating the co-creation of knowledge between service providers and customers. In the context of software development, knowledge management can help developers to gain insights into customer needs, preferences, and behaviors, which can be used to design and develop software products that better meet customer requirements.

Another article by [39] explores the antecedents and outcomes of service co-creation in the sharing economy. The study emphasizes the importance of customer participation in the co-creation process, which is essential for creating value in the sharing economy. The authors argue that customer participation in the co-creation process helps to create a sense of ownership



among customers, which can lead to increased loyalty and satisfaction. In the context of software development, this sense of ownership can help to ensure that software products are developed in a way that meets customer requirements and expectations.

Lastly, [40] discuss the concept of co-creation of knowledge in higher education, which emphasizes the importance of collaboration between educators and students in the creation of knowledge. The authors argue that co-creation of knowledge in higher education can help to create a more engaging and effective learning experience for students. In the context of software development, this concept can be applied to the development of educational software products, where collaboration between educators and software developers can help to create software products that better meet the needs of students and educators.

In conclusion, the theme of co-creation of knowledge is an essential concept in the context of service-dominant logic, and several articles highlight its importance in the development of software products. By emphasizing the importance of customer participation, knowledge management, and collaboration, software developers can design and develop software products that better meet customer requirements and expectations.

#### **4.5. Value Co-creation Networks**

The theme of value co-creation networks is central to the software product structure in the context of service-dominant (S-D) logic. The concept of value co-creation networks posits that value is created through the collaboration of multiple actors in a network, such as customers, employees, and partners [41]. In the software industry, value co-creation networks are increasingly important, as software products are becoming more complex and require collaboration across various stakeholders. This scoping review explores the literature on software product structure in the context of S-D logic, with a focus on the theme of value co-creation networks.

One study that contributes to the theme of value co-creation networks is the research by [36], which proposes design principles for digital value co-creation networks from a service-dominant logic perspective. The authors argue that value co-creation networks in the digital space require a

design that is based on collaboration, transparency, and interoperability. This study highlights the importance of designing value co-creation networks in software development, as software products are increasingly becoming a collaboration of various actors.

Another study that contributes to the theme of value co-creation networks is the research by [37], which proposes a service-dominant business model design for digital innovation in smart mobility. The authors argue that a service-dominant business model can facilitate value co-creation networks in smart mobility, as it encourages collaboration among stakeholders and the use of digital technologies. This study highlights the importance of adopting a service-dominant business model in software development, as it can facilitate the creation of value co-creation networks.

Furthermore, the study by [12] applies service-dominant logic to recurrent release of software in an action research study. The authors argue that service-dominant logic provides a framework for continuous improvement in software development, as it emphasizes the importance of collaboration among stakeholders and the co-creation of value. This study highlights the importance of adopting service-dominant logic in software development, as it can facilitate the creation of value co-creation networks.

Moreover, the research by [25] explores paths toward radical service innovation in manufacturing companies from a service-dominant logic perspective. The authors argue that service-dominant logic can facilitate radical service innovation by enabling collaboration among stakeholders and the co-creation of value. This study highlights the importance of adopting service-dominant logic in software development, as it can facilitate the creation of value co-creation networks and enable radical innovation.

Finally, the research by [28] examines the role of institutions and technology in the value co-creation process of restaurant consumers from a service-dominant logic perspective. The authors argue that value co-creation networks in the food service industry require collaboration among actors and the use of technology to enable value co-creation. This study highlights the importance of technology in enabling value co-creation networks in software development, as software products increasingly require collaboration among various actors.

In conclusion, the theme of value co-creation networks is central to the software product

structure in the context of service-dominant logic. The studies reviewed in this scoping review demonstrate the importance of adopting service-dominant logic and designing value co-creation networks to facilitate collaboration among stakeholders and the co-creation of value in software development. These findings have implications for software developers who seek to create software products that are responsive to the needs of various actors in value co-creation networks.

#### 4.6. Key synthesis and summary

Figure 8 provides a synthesis and summary of the five key themes related to the software product structure within the context of S-D logic, as identified from the articles reviewed in this study. These themes highlight how S-D logic is relevant to various aspects of the software industry. They emphasize the importance of adopting a service-centric approach and value co-creation strategies from an S-D logic perspective, enabling companies to better understand and meet customer needs, design tailored solutions, and continuously improve products based on customer feedback.

However, it appears that the specific aspects of software product structure within the context of S-D logic, which is the main research question of this study, are not well-defined in the reviewed articles, and hence this study's relevance is established. Additionally, the literature lacks discussion on productization through product structure for software product releases, which is an important business practice in the software industry.

Further research is needed to address these identified gaps and advance the understanding of software product structure within the context of S-D logic. This will provide practical guidance for software companies in implementing S-D logic and value co-creation strategies. This scoping review serves as a valuable resource for future research and aims to generate interest among researchers in this topic.

### 5. Conclusion

In conclusion, this study highlights the significance of approaching software product structures from a Service-Dominant Logic (S-D Logic) perspective. The study employed a scoping review methodology to examine the literature on

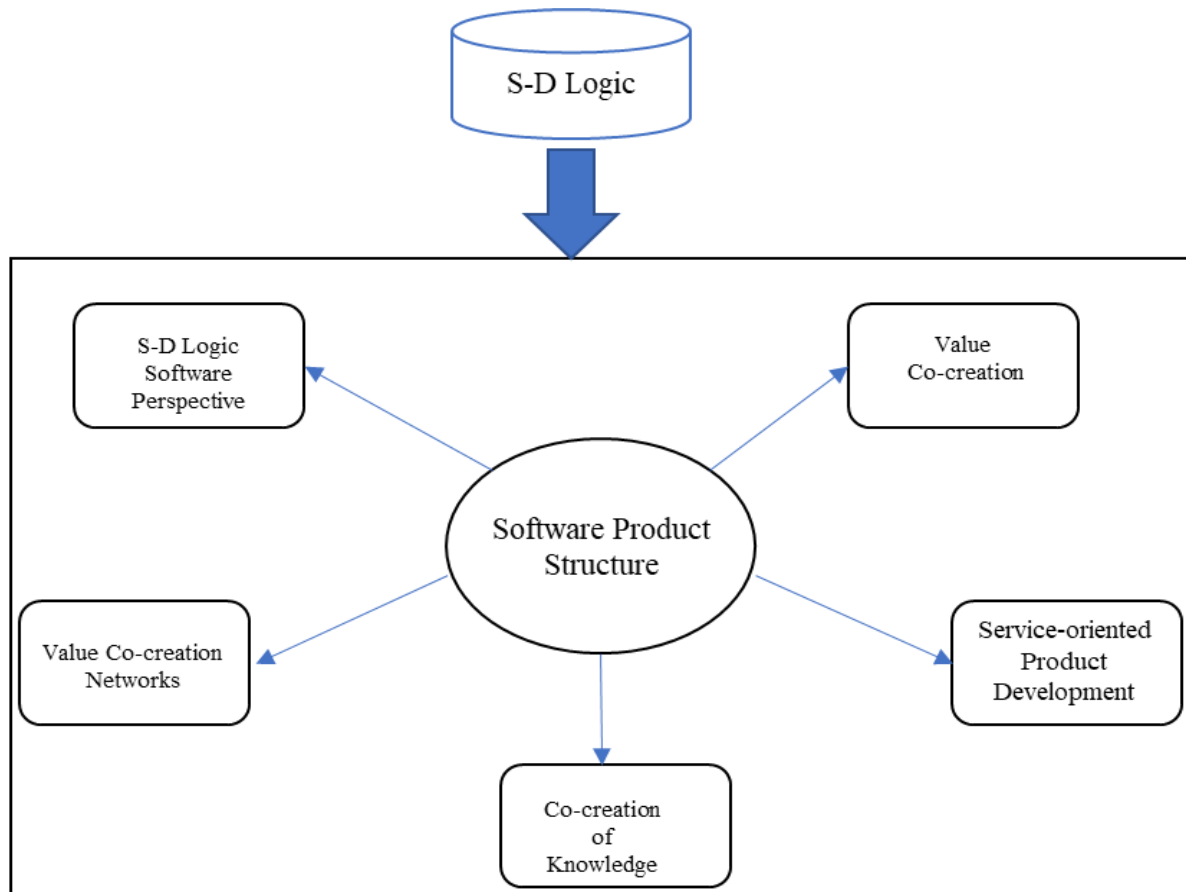
software productization and product structure in the context of Service-Dominant (S-D) Logic. The five-stage scoping process by [27] including identifying the research question, compiling relevant studies for the literature search, applying inclusion criteria to select relevant articles, and extracting data to chart and synthesize the findings was adopted. A total of 56 research articles were selected for the final analysis. The study found patterns and gaps in the literature related to software product structure in the S-D logic context, which can inform future research and development in this area.

The review identifies five key themes: the S-D Logic Perspective, Value Co-Creation, Service-Oriented Product Development, Co-Creation of Knowledge, and Value Co-Creation Networks. These themes provide a comprehensive picture of the state of research in this area software productization from S-D logic perspective and suggest several avenues for future research.

The review provides a valuable resource for researchers interested in this topic by synthesizing the existing literature on software product structure in the context of S-D Logic. It highlights the importance of value co-creation, which has become an important aspect of S-D Logic and has been shown to drive innovation and enhance customer satisfaction. Additionally, the review underscores the need for software companies to shift from a product-centric approach to a service-centric approach that emphasizes co-creation of value with customers.

From a managerial perspective, the findings of this review have important implications for software companies. The adoption of S-D Logic can help companies to better understand and meet customer needs, design solutions that are tailored to their needs, and continuously iterate and improve the product based on customer feedback. Additionally, the review highlights the importance of aligning resources, actors, and contexts for value creation, which underscores the need for companies to foster strong relationships with customers and to leverage institutional and technological resources to co-create experiences with them.

The review also emphasizes the importance of designing value co-creation networks in software development, as software products increasingly require collaboration among various stakeholders. The reviewed studies demonstrate the importance of adopting service-dominant logic and designing value co-creation networks to facilitate



**Figure 4:** Key Literature findings synthesised

collaboration among stakeholders and the co-creation of value in software development.

However, the review also has some limitations. The scoping review methodology may not have captured all relevant literature on the topic, and the literature analyzed focused mainly on the software industry, which may limit the generalizability of the findings. Additionally, while the review identified several avenues for future research, the specific research questions that need to be addressed in order to advance the understanding of software product structure in the context of S-D Logic are not well-defined.

In summary, this scoping review provides a comprehensive understanding of the literature on software product structure in the context of S-D Logic. It highlights the importance of adopting a service-centric approach to software product development, emphasizes the role of value co-creation, and identifies several avenues for future research. The findings of this review have important implications for software companies, as they underscore the need for companies to foster strong relationships with customers and to

leverage institutional and technological resources to co-create experiences with them. While the review has some limitations, it provides a valuable resource for researchers interested in this topic and lays the groundwork for future research in this area.

## 6. Acknowledgements

Foundation for Economic Education supported this work: [Grant Number 22-12080]

## 7. References

- [1] P. Artz, I. van de Weerd, and S. Brinkkemper, "Productization: The process of transforming from customer-specific software development to product software development".
- [2] L. Xu and S. Brinkkemper, "Concepts of Product Software: Paving the Road for Urgently Needed Research," The first International Workshop on Philosophical Foundations of Information Systems Engineering (PHISE'05), pp. 523–528, 2005.

- [3] M. A. Akaka, K. Koskela-Huotari, and S. L. Vargo, "Further Advancing Service Science with Service-Dominant Logic: Service Ecosystems, Institutions, and Their Implications for Innovation," in *Handbook of Service Science*, Volume II, P. P. Maglio, C. A. Kieliszewski, J. C. Spohrer, K. Lyons, L. Patrício, and Y. Sawatani, Eds., in *Service Science: Research and Innovations in the Service Economy*. Cham: Springer International Publishing, 2019, pp. 641–659. doi: 10.1007/978-3-319-98512-1\_28.
- [4] S. L. Vargo and R. F. Lusch, "Evolving to a New Dominant Logic for Marketing," *Journal of Marketing*, vol. 68, no. 1, pp. 1–17, Jan. 2004, doi: 10.1509/jmkg.68.1.1.24036.
- [5] S. Gbadegeshin, "The Effect of Digitalization on the Commercialization Process of High-Technology Companies in the Life Sciences Industry," *Technology Innovation Management Review*, vol. 9, no. 1, pp. 49–63, 2019, doi: <http://doi.org/10.22215/timreview/1211>.
- [6] S. A. Fricker, "Software Product Management," in *Software for People: Fundamentals, Trends and Best Practices*, A. Maedche, A. Botzenhardt, and L. Neer, Eds., in *Management for Professionals*. Berlin, Heidelberg: Springer, 2012, pp. 53–81. doi: 10.1007/978-3-642-31371-4\_4.
- [7] K. Rautiainen, L. Vuornos, and C. Lassenius, "An experience in combining flexibility and control in a small company's software product development process," in *2003 International Symposium on Empirical Software Engineering, 2003. ISESE 2003. Proceedings.*, Sep. 2003, pp. 28–37. doi: 10.1109/ISESE.2003.1237962.
- [8] A. G. Adusei, J. Härkönen, and E. Mustonen, "Productization and Product Structure: Extending The Perspective to Software Business," *Int. j. bus. adm. stud.*, vol. 7, no. 2, Apr. 2021, doi: 10.20469/ijbas.7.10004-2.
- [9] S. Mohapatra and R. Roy, "Moving to products: entrepreneurial IT firms in India," *International Journal of Project Organisation and Management*, vol. 4, no. 1, pp. 86–97, Jan. 2012, doi: 10.1504/IJPOM.2012.045361.
- [10] M. A. Cusumano, "The Changing Software Business: Moving from Products to Services," *Computer*, vol. 41, no. 1, pp. 20–27, Jan. 2008, doi: 10.1109/MC.2008.29.
- [11] K.-V. Hanna, H. Harri, J. Olli, and P. Kongkiti, "Product Configuration Management in ICT Companies: The Practitioners' Perspective," *Technology and Investment*, vol. 2011, Nov. 2011, doi: 10.4236/ti.2011.24028.
- [12] N. Barqawi, K. Syed, and L. Mathiassen, "Applying service-dominant logic to recurrent release of software: an action research study," *Journal of Business & Industrial Marketing*, vol. 31, no. 7, pp. 928–940, Jan. 2016, doi: 10.1108/JBIM-02-2015-0030.
- [13] X. Font, R. English, A. Gkritzali, and W. (Stella) Tian, "Value co-creation in sustainable tourism: A service-dominant logic approach," *Tourism Management*, vol. 82, p. 104200, Feb. 2021, doi: 10.1016/j.tourman.2020.104200.
- [14] J. Williams, "The logical structure of the service-dominant logic of marketing," *Marketing Theory*, vol. 12, no. 4, pp. 471–483, Dec. 2012, doi: 10.1177/1470593112457745.
- [15] S. L. Vargo and R. F. Lusch, "Service-dominant logic: continuing the evolution," *J. of the Acad. Mark. Sci.*, vol. 36, no. 1, pp. 1–10, Mar. 2008, doi: 10.1007/s11747-007-0069-6.
- [16] A. F. Payne, K. Storbacka, and P. Frow, "Managing the co-creation of value," *J. of the Acad. Mark. Sci.*, vol. 36, no. 1, pp. 83–96, Mar. 2008, doi: 10.1007/s11747-007-0070-0.
- [17] S. L. Vargo and R. F. Lusch, "Institutions and axioms: an extension and update of service-dominant logic," *J. of the Acad. Mark. Sci.*, vol. 44, no. 1, pp. 5–23, Jan. 2016, doi: 10.1007/s11747-015-0456-3.
- [18] A. J. Ibnu WIBOWO, U. Sumarwan, B. Suharjo, and M. Simanjuntak, "17 Years of Service-Dominant Logic: Vargo and Lusch's Contributions," *Business: Theory & Practice*, vol. 22, no. 2, pp. 482–492, Jul. 2021, doi: 10.3846/btp.2021.13050.
- [19] S. L. Vargo and R. F. Lusch, "Service-dominant logic 2025," *International Journal of Research in Marketing*, vol. 34, no. 1, pp. 46–67, Mar. 2017, doi: 10.1016/j.ijresmar.2016.11.001.
- [20] M.-K. Vespestad and A. Clancy, "Service dominant logic and primary care services," *International Journal of Quality and Service Sciences*, vol. 11, no. 1, pp. 127–140, Jan. 2018, doi: 10.1108/IJQSS-02-2018-0012.
- [21] R. A. Wells, M. G. Gallarza, L. Andreu, and R. Calero, "Application of Service Dominant Logic to the medical device industry," *Int Rev Public Nonprofit Mark.*, vol. 12, no. 3, pp. 207–235, Oct. 2015, doi: 10.1007/s12208-015-0126-7.
- [22] M. Díaz-Méndez, M. R. Paredes, and M. Saren, "Improving Society by Improving Education through Service-Dominant Logic: Reframing the Role of Students in Higher

Education,” *Sustainability*, vol. 11, no. 19, Art. no. 19, Jan. 2019, doi: 10.3390/su11195292.

[23] D. Yar Hamidi and S. Machold, “Governance, boards and value co-creation: Changing perspectives towards a service dominant logic,” *European Management Journal*, vol. 38, no. 6, pp. 956–966, Dec. 2020, doi: 10.1016/j.emj.2020.06.001.

[24] K. K. Lindsey Hall, J. (Miracle) Qi, R. G. Richey, and R. K. Patil, “Collaboration, feedback, and performance: Supply chain insights from service-dominant logic,” *Journal of Business Research*, vol. 146, pp. 385–397, Jul. 2022, doi: 10.1016/j.jbusres.2022.03.055.

[25] R. Chester Goduscheit and R. Faullant, “Paths Toward Radical Service Innovation in Manufacturing Companies—A Service-Dominant Logic Perspective,” *Journal of Product Innovation Management*, vol. 35, no. 5, pp. 701–719, 2018, doi: 10.1111/jpim.12461.

[26] D. Sarno and P. Siano, “Exploring the Adoption of Service-Dominant Logic as an Integrative Framework for Assessing Energy Transitions,” *Sustainability*, vol. 14, no. 15, Art. no. 15, Jan. 2022, doi: 10.3390/su14159755.

[27] H. Arksey and L. O’Malley, “Scoping studies: towards a methodological framework,” *International Journal of Social Research Methodology*, vol. 8, no. 1, pp. 19–32, Feb. 2005, doi: 10.1080/1364557032000119616.

[28] G. Scarlett, R. Reksoprawiro, N. Amelia, and A. J. I. Wibowo, “Institutions and technology in the value co-creation process of restaurant consumers: a service-dominant logic perspective,” *The TQM Journal*, vol. 34, no. 3, pp. 357–376, Jan. 2021, doi: 10.1108/TQM-10-2020-0255.

[29] M. Kujamäki, “Applying Service-Dominant Logic to Translation Service Provision,” 2020, Accessed: Apr. 03, 2023. [Online]. Available: <https://erepo.uef.fi/handle/123456789/8258>

[30] P. Ekman, J. Rödell, and Y. Yang, “Exploring smart cities and market transformations from a service-dominant logic perspective,” *Sustainable Cities and Society*, vol. 51, p. 101731, Nov. 2019, doi: 10.1016/j.scs.2019.101731.

[31] F. J. Carrillo, B. Edvardsson, J. Reynoso, and E. Maravillo, “Alignment of resources, actors and contexts for value creation: Bringing knowledge management into service-dominant logic,” *International Journal of Quality and Service Sciences*, vol. 11, no. 3, pp. 424–438, Jan. 2019, doi: 10.1108/IJQSS-08-2018-0077.

[32] Y. Jiang, M. S. Balaji, and S. Jha, “Together we tango: Value facilitation and customer participation in Airbnb,” *International Journal of Hospitality Management*, vol. 82, pp. 169–180, Sep. 2019, doi: 10.1016/j.ijhm.2019.05.004.

[33] K. Haverila, M. Haverila, and C. McLaughlin, “Development of a brand community engagement model: a service-dominant logic perspective,” *Journal of Consumer Marketing*, vol. 39, no. 2, pp. 166–179, Jan. 2022, doi: 10.1108/JCM-01-2021-4390.

[34] R. F. Lusch, S. L. Vargo, and M. Tanniru, “Service, value networks and learning,” *J. of the Acad. Mark. Sci.*, vol. 38, no. 1, pp. 19–31, Feb. 2010, doi: 10.1007/s11747-008-0131-z.

[35] R. Gilsing et al., “Evaluating the Design of Service-Dominant Business Models: A Qualitative Method,” *Pacific Asia Journal of the Association for Information Systems*, vol. 13, no. 1, Mar. 2021, doi: 10.17705/1pais.13102.

[36] M. Blaschke, U. Riss, K. Haki, and S. Aier, “Design principles for digital value co-creation networks: a service-dominant logic perspective,” *Electron Markets*, vol. 29, no. 3, pp. 443–472, Sep. 2019, doi: 10.1007/s12525-019-00356-9.

[37] O. Turetken, P. Grefen, R. Gilsing, and O. E. Adali, “Service-Dominant Business Model Design for Digital Innovation in Smart Mobility,” *Bus Inf Syst Eng*, vol. 61, no. 1, pp. 9–29, Feb. 2019, doi: 10.1007/s12599-018-0565-x.

[38] K. R. Ranjan and S. Read, “Bringing the individual into the co-creation of value,” *Journal of Services Marketing*, vol. 33, no. 7, pp. 904–920, Jan. 2019, doi: 10.1108/JSM-02-2019-0056.

[39] F. Evangelista et al., “Antecedents and Outcomes of Service Co-Creation in the Sharing Economy,” *SAGE Open*, vol. 12, no. 2, p. 21582440221096440, Apr. 2022, doi: 10.1177/21582440221096438.

[40] M. de F. Cruz, H. Alves, and R. Gouveia Rodrigues, “A service-dominant logic of co-creation in higher education: emerging topics and conceptualizations,” *Journal of Marketing for Higher Education*, vol. 0, no. 0, pp. 1–26, Oct. 2022, doi: 10.1080/08841241.2022.2134957.

[41] C. K. Prahalad and V. Ramaswamy, “Co-creation experiences: The next practice in value creation,” *Journal of Interactive Marketing*, vol. 18, no. 3, pp. 5–14, Jan. 2004, doi: 10.1002/dir.20015.