

Wulfert, Tobias

Article — Published Version

Boundary resource management in innovation ecosystems: The case of e-commerce

Electronic Markets

Provided in Cooperation with:

Springer Nature

Suggested Citation: Wulfert, Tobias (2023) : Boundary resource management in innovation ecosystems: The case of e-commerce, Electronic Markets, ISSN 1422-8890, Springer, Berlin, Heidelberg, Vol. 33, Iss. 1, <https://doi.org/10.1007/s12525-023-00651-6>

This Version is available at:

<https://hdl.handle.net/10419/306373>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by/4.0/>



Boundary resource management in innovation ecosystems: The case of e-commerce

Tobias Wulfert¹

Received: 17 November 2022 / Accepted: 27 April 2023 / Published online: 29 May 2023
© The Author(s) 2023

Abstract

Major challenges in contemporary information systems development demand new development approaches, such as innovation ecosystems, consisting of a focal innovation platform surrounded by independent developers. The attraction of third-party developers and the amplification of generativity in extension development call for sophisticated boundary resources. As demarcation points between an innovation platform and external developers, boundary resources entail a strategic value and need to be managed carefully. However, existing research does not provide an approach for the holistic management of boundary resources from design to retirement. We apply a design science research approach consisting of two design cycles involving a literature review identifying 26 application lifecycles and the analysis of eight innovation platforms in e-commerce. We evaluated the model for boundary resource management in interviews with nine domain experts. We develop an integrated boundary resource management lifecycle as a specialization of application lifecycle management for the holistic management of boundary resources in innovation ecosystems addressing third-party developers. The boundary resource management lifecycle consists of four layers: boundary resource governance, continuous communication, technical boundary resources, and supplementary boundary resources. It decouples technical and supplementary boundary resources while emphasizing their intertwined nature for ecosystem participants. Owners of innovation platforms can instantiate the reference procedure model for attracting developers.

Keywords Boundary resources · Innovation platform · Application lifecycle management · E-commerce · Design science research

JEL classification L140 · L220 · O320

Introduction

Major challenges in contemporary information systems (IS) development, such as rapid technological advancements, continuously changing requirements (Norbjerg, 2018), increasing customer demands, shortened application release cycles (Babb et al., 2017), more complex application systems, and limited resources, demand new development approaches. The transformation of complex and monolithic

application systems into innovation platforms that provide basic application services and attract external developers for an innovation ecosystem increases the overall generativity with additional apps, themes, and software integrations (Aulkemeier et al., 2016a; Hein et al., 2020). This transformation allows for continuous application engineering and delivery (Dittrich et al., 2018). However, it requires the implementation of boundary resources that enable collaboration and value co-creation among different types of actors by translating diverging viewpoints and interfacing an innovation platform's application services (Ghazawneh & Henfridsson, 2013; Star, 2010). Boundary resources as a means for opening an innovation platform propel the ecosystem's generativity, influence the relationship between a platform owner and third-party developers, and thus increase the platform adoption by third-party developers (Eaton et al., 2015; Engert et al., 2022; Miremadi et al., 2023). Examples

Responsible Editor: Paul Timmers

✉ Tobias Wulfert
tobias.wulfert@icb.uni-due.de

¹ University of Duisburg-Essen,
Chair for Business Informatics and Integrated Information Systems,
Universitätsstraße 2, 45141 Essen, Germany

of boundary resources include application programming interfaces (APIs), integrated development environments (IDEs), and software development kits (SDKs) with related documentation (Dal Bianco et al., 2014; Eaton et al., 2015; Ghazawneh & Henfridsson, 2010) that provide external developers with access to the core functions of the platform (Karhu et al., 2018). Platform boundary resources need to be altered and updated continuously to cope with new market developments, create new business opportunities, and protect the platform against exploitation by third parties (Eaton et al., 2015; Ghazawneh & Henfridsson, 2010; Jansen et al., 2013; Karhu et al., 2018). As demarcation points between an innovation platform and ecosystem participants (Santos & Eisenhardt, 2005), boundary resources bring about a strategic value to the platform owner and need to be managed carefully (Bondel et al., 2021; Vester, 2018; Y. Yoo et al., 2010). Obtaining control over the boundary resources of an open platform is a key element of a platform owner's ecosystem governance (Baldwin & Woodard, 2009; Boudreau, 2010; Pon et al., 2014; Schilling, 2000).

Following the seminal works of Star and Griesemer (1989) as well as Ghazawneh and Henfridsson (2010), the research interest on boundary resources in the context of digital platforms and their surrounding innovation ecosystem is on the increase. Dal Bianco et al. (2014) analyzed the use of boundary resources in several software ecosystems. Karhu et al. (2018) compared the use of boundary resources with the licensing of a platform's intellectual property. While Petrik and Herzwurm (2020) investigated boundary resources for industrial internet of things scenarios, Wulfert et al. (2022) focused on the implementation of boundary resources in e-commerce ecosystems. Although boundary resources are generally applicable in innovation ecosystems in different domains (Ghazawneh, 2012), scientific literature on the management of boundary resources is scarce.

Owing to their strategic relevance for innovation platforms (Boudreau, 2010; Y. Yoo et al., 2010), a comprehensive management approach that covers the whole lifecycle from the design to the retirement of the boundary resources is necessary to exploit their full potential. A lifecycle model for boundary resources can be used "as a common reference for communication and understanding" (ISO, 2017, p. 25) among ecosystem participants. Since boundary resources involve and are concerned with software artifacts, we apply an application lifecycle perspective for developing a boundary resource management approach. The concept of application lifecycle management (ALM) is made popular in the Information Technology Infrastructure Library (ITIL) with its application management functions (Fischbach et al., 2013). Many ALM lifecycles in the literature (e.g., Hallerstedte et al., 2011) are inspired by the application

management cycle included in the ITIL framework (AXELOS, 2012). The ALM considers the entire lifecycle of a software artifact by describing how the software development process should take place from the initial development phase up to the delivery phase and subsequent maintenance (Lacheiner & Ramler, 2011). The global market for ALM is expected to grow to USD 5 billion by 2027, with USD 2.8 billion specifically advocated for ALM tools (StrategyR, 2022). The most popular tools for ALM are, among others, Jama Software, Atlassian Jira, Microsoft Azure DevOps, and Visure (Aston, 2022; Jwo et al., 2013).

Although ALM bears a high potential for innovation ecosystems and although ALM seems to be a promising approach for managing boundary resources, the current body of literature does not consider boundary resources. Research investigated several specializations of ALM for, among others, security management (Demchenko et al., 2017; Zeng et al., 2021), data management (Alhassan et al., 2016; Chen & Zhao, 2012; Khatri & Brown, 2010), cloud application monitoring (Aversa et al., 2015), cloud application provisioning (Benfenatki et al., 2017), and sensor networks (Kabac et al., 2016). Moreover, existing research only focuses on single lifecycle phases, such as retirement (Yu et al., 2017), or investigates APIs as specific artifacts. Several practitioner-based lifecycle approaches exist for API management (Ebert, 2012; Medjaoui et al., 2021; Rossberg, 2014; Vester, 2018; Weir, 2019; Zulian & Bouza, 2021), but without identifying an agreed-upon approach (Bondel et al., 2021).

We therefore go beyond an isolated view on APIs and take a broader perspective on boundary resources for innovation platforms that include technical (e.g., APIs, SDKs) and non-technical (e.g., documentation) artifacts. With this approach, we cover the whole lifecycle of a boundary resource from design to retirement. We provide literature on ALM, with a dedicated lifecycle model on boundary resources required for inter-company and ecosystem-wide collaboration and value co-creation (Engert et al., 2022; Wulfert et al., 2022). While platform owners gain a customizable reference procedure model for the management of boundary resources and the attraction of third-party developers, these developers can anticipate boundary resource evolutions with regard to the reference model. We make use of initial but roughly designed processes for boundary resource management (e.g., Ghazawneh & Henfridsson, 2010; Hein et al., 2019; Star, 2010) and provide a comprehensive reference procedure model. Furthermore, we also consider organizational aspects to ensure an adequate degree of platform openness and ecosystem generativity (Blaschke & Brosius, 2018; Boudreau, 2010). Against this backdrop, we raise the following research question:

How can boundary resources of focal platforms in innovation ecosystems be managed?

To address this research question, we apply a design science research (DSR) approach (Takeda et al., 1990; Vaishnavi et al., 2019). Our DSR approach involves a literature synthesis on ALM in general and boundary resource management in particular (vom Brocke et al., 2015; Webster & Watson, 2002), analyzing the boundary management of selected innovation platforms in e-commerce from the theoretical lens of ALM (Niederman & March, 2019; Yin, 2018). In addition, our DSR approach also involves an evaluation of these findings, with the analysis of nine expert interviews to check our developed lifecycle model and to gain further insights on internal boundary resource development (Misoch, 2015). We chose the e-commerce domain for our boundary resource analysis, as several types of participants are orchestrated using digital technologies in e-commerce ecosystems (Wulfert et al., 2022). Furthermore, collaborative and distributed software development supports diverse electronic business models (Alt, 2020; Aulkemeier et al., 2016a; Timmers, 1998). We propose the boundary resource management lifecycle as a reference procedure model for owners of innovation platforms to implement sophisticated boundary resources and to attract additional third-party developers with their extensions. We provide a detailed reference procedure model for the management of boundary resources to the literature on ALM. Moreover, we emphasize the role of supplementary social boundary resources to enable the use of technical boundary resources by third-party developers.

The remainder of this research paper proceeds with an elicitation on related literature on innovation platforms as the technical foundation of e-commerce ecosystems, existing approaches for boundary resource management, and ALM introduced as the theoretical lens of this research. Second, we elaborate on our DSR approach, incorporating a systematic literature review, a multi-case study, and an evaluation based on an interview analysis. Third, we present our boundary resource management lifecycle by describing the derived layers and single lifecycle stages for technical boundary resources and by mapping them with related supplementary boundary resources. Finally, we discuss our results, provide a research agenda for potential avenues for future research, and briefly present our conclusions.

Related literature

Innovation platforms in e-commerce ecosystems

Facilitated by the ongoing digitalization of the retail sector, (parts of) retail and wholesale functions (e.g., bridging, information, storing) are performed digitally to a certain degree in e-commerce (Laudon & Traver, 2020). In e-commerce,

transactions involving information, products, and services are executed using information and communication technology (Choi et al., 1997; Nath et al., 1998; Zwass, 1996). E-commerce ecosystems are conceptualized as manifestations of digital business ecosystems in the context of e-commerce, centered around a focal transaction platform (Wulfert et al., 2022; Yi & Ming, 2011). The transaction platform orchestrates several types of independent ecosystem participants, such as manufacturers, customers, and service providers (Böttcher et al., 2021), who share a common interest, depend on each other, and co-create value (Blaschke et al., 2018; Iansiti & Levien, 2004; McIntyre & Srinivasan, 2017; Wareham et al., 2014). Benefiting from direct and indirect network effects (Armstrong, 2006; Shapiro & Varian, 1998), the value of an e-commerce ecosystem for each participant increases with additional participants joining (Hinz et al., 2020). The purpose of an e-commerce ecosystem is the exchange of physical and digital products for the benefit of a customer (Adner, 2017; Wulfert et al., 2022).

The technology-supported transactions in e-commerce ecosystems are enabled by innovation platforms that provide the necessary application services for the exchange of goods between and value co-creation among ecosystem participants (Aulkemeier et al., 2016b; Blaschke et al., 2019b; Wulfert & Karger, 2022). Hence, innovation platforms form the technological infrastructure for e-commerce ecosystems (Wulfert & Karger, 2022). While Blaschke et al., 2019a p. 1) termed innovation platforms as the “technical core” of a surrounding ecosystem, Baldwin and Woodard (2009, p. 19) defined an innovation platform as “a set of stable components that supports variety and evolvability in a system by constraining the linkages among the other components.” By benefitting from these linkages, third-party developers can extend the features of an innovation platform with a variety of extensions, which they achieve by taking advantage of the innovation platform’s architectural modularity and the technical and organizational components’ re-configurability (Jansen & van Capelleveen, 2013; Tiwana et al., 2010). These extensions make use of the application services provided by the innovation platform via a set of boundary resources, which are implemented by the platform owner (Ghazawneh & Henfridsson, 2010). By implementing a set of boundary resources as part of its governance function (Boudreau, 2010), the platform owner defines the openness of and the access to the innovation platform (Boudreau, 2010; Karhu et al., 2018; Tiwana, 2014). A certain degree of platform openness invokes positive network effects and increases generativity for the ecosystem (Blaschke & Brosius, 2018; Karhu et al., 2018).

Tiwana et al. (2010, p. 675) emphasized that third-party developers can use these stable components as an “extensible codebase [...] that provides core functionality shared by the modules that interoperate with it and the interfaces through which they interoperate.” While the stable components form the core of the innovation platform, the third-party modules resemble the periphery (Staykova & Damsgaard, 2015). The

platform core and periphery are evolvable, with only the boundary resource specifications that should remain stable over a longer period of time to minimize necessary extension adjustments for the developers (Baldwin & Woodard, 2009). According to Zutshi and Grilo (2019), a digital platform is composed of six different layers, namely business, user interaction, development, integration, data, and the information technology layer. Blaschke et al. (2019a) mentioned four layers of digital platforms. They described the business layer as service dimension and called the user interaction the ecosystem dimension that allows interaction with the platform core. The platform core includes application services and boundary resources (i.e., extensible codebase) necessary for the development of platform extensions. The platform core is executed by a digital infrastructure.

Boundary resource management

The concept of boundary resources stems from the social sciences, in which “boundary objects” (Star & Griesemer, 1989, p. 393) function as “demarcation points” (Santos & Eisenhardt, 2005, p. 491) between different social worlds (Santos & Eisenhardt, 2005; Star & Griesemer, 1989). Boundary resources enable the collaboration and co-creation of value among different types of actors by translating between diverging viewpoints (Star, 2010), and are defined as “software tools and regulations that serve as the interface for the arm’s-length relationship between the platform owner and the application developer” (Ghazawneh & Henfridsson, 2013, p. 175). Since boundary resources serve as interfaces between different ecosystem participants, they move within companies’ organizational boundaries (Santos & Eisenhardt, 2005). Hein et al. conceptualize a set of boundary resources as “affordances” (Hein et al., 2020, p. 91) of the digital platform. These affordances form the boundary and the demarcation between the digital platform core and third-party developers (i.e., complementors) providing platform extension and amplifying generativity. Hence, boundary resources enable collaboration and value co-creation among different ecosystem participants (Autio, 2021; Engert et al., 2022; Hein et al., 2019; Wulfert et al., 2022).

Innovation platforms in e-commerce ecosystems provide boundary resources that support the design, implementation, and provision of third-party developers’ extensions (Ghazawneh & Henfridsson, 2013). Boundary resources lower entrance barriers to the ecosystem for new developers (Alves et al., 2017; Hein et al., 2020) and facilitate access to the focal innovation platform’s core resources (Baldwin & Woodard, 2009; Eaton et al., 2015). The set of boundary resources provided influences the innovation platform’s openness and the associated ecosystem’s generativity (Ghazawneh & Henfridsson, 2010; Karhu et al., 2018; Parker et al., 2016).

The concept of boundary resources includes a variety of manifestations (Eaton et al., 2015). Star and Griesemer

(1989) introduced repositories, ideal type, coincident boundaries, and standardized forms as boundary resource types. Dal Bianco et al. (2014) described application boundary resources as interfaces for the third-party extensions’ interaction with the focal platform, development boundary resources as facilitating the third-party extensions’ design and implementation, and social boundary resources as enabling knowledge transfer and the coordination of extension development and boundary resource usage (Carlile, 2002). Application and development boundary resources provide access to the integration and development layer introduced by Zutshi and Grilo (2019). Karhu et al. (2018) developed a set of 16 boundary resources for open digital platforms. Building upon the publications of Ghazawneh (2012) and Dal Bianco et al. (2014), Wulfert et al. (2022) synthesized 14 distinct boundary resources. These boundary resources can be grouped using the boundary resource layers of Dal Bianco et al. (2014). Social boundary resources involve documentation, roadmaps, registration forms, forums, support, and hackathons. While application boundary resources include APIs and workflows, development boundary resources include SDKs, software libraries, software repositories, debugging tools, prototypes, and mockups. We utilize Dal Bianco et al. (2014) conceptualization and differentiate technical boundary resources (i.e., application and development boundary resources) and supplementary social boundary resources. As suggested by Hein et al. (2020), we integrate both perspectives in our boundary resource management lifecycle.

Existing approaches for developing and managing boundary resources in the context of innovation platforms highlight an iterative approach. Star (2010) introduced a model for describing the evolution of boundary resources consisting of standardization and residualization phases. While standardization describes a process in which certain aspects of the boundary object are standardized for a wider application, residualization refers to a process of manipulation of the boundary resource for the specific purpose and use of a group of users. Hein et al. (2019) adapted this initial model, with an additional state for the boundary resource as a result of the standardization. If a boundary resource is successfully standardized and not transferred into a residual, it can become part of standardized IT infrastructure. Ghazawneh and Henfridsson (2010) emphasized a platform governance perspective in their iterative boundary resource development, balancing platform control, and ecosystem generativity.

Application lifecycle management as theoretical lens

ALM was introduced as a complement to product lifecycle management (PLM) for the development, maintenance, and optimization of application systems in organizations

(Kääriäinen, 2011; Stark, 2011). While PLM is concerned with hardware lifecycles, the focus of ALM is on software (Kääriäinen, 2011). This requires an integration of both concepts for smart products (Beverungen et al., 2019; Deuter & Rizzo, 2016). In analogy to lifecycles of species in biology (Schuh & Rudolf, 2015), ALM covers the whole software lifecycle from requirements engineering to the deprecation of the application (Stark, 2015). In contrast to this broad understanding of ALM, several research papers focus on software development and their investigations end with the deployment of the software artifact. Software configuration management is seen as a building block of ALM (Kääriäinen, 2011); it provides a repository for software components, supports development activities, and controls the overall development (Estublier, 2000). ALM goes beyond these software development management activities, managing artifacts and documentation that is created along the entire lifecycle of an application, and ALM tools integrate them in overarching tools (Kääriäinen, 2011). In this research paper, we apply the broad perspective of ALM as a theoretical lens (Niederman & March, 2019). This perspective complements application development and operation involving, among other phases, the design, implementation, maintenance, and optimization in a continuous cycle (Table 5).

Although the term application lifecycle management is widely used in software engineering and management, an agreed-upon definition still does not exist (Lacheiner & Ramler, 2011; Pirklbauer et al., 2009). ALM involves “the coordination of activities and the management of artefacts (e.g., requirements, source code, test cases) during the software product’s lifecycle” (Kääriäinen & Välimäki, 2009, p. 150). Rossberg (2014) emphasized that an application’s lifecycle starts with first thoughts on application development and ends with the application’s actual retirement, with possible changes along the application’s runtime. Besides the interplay between application development and operation, ALM involves governance as a third major activity (Tüzün et al., 2019). Pirkelbauer et al. (2009) suggested an integrated perspective (i.e., concepts, management, engineering) on ALM and emphasized the role of ALM in business–IT alignment. Chanda and Foggon (2013) focused on coordination and management activities for people, investment, and technical artifacts over the entire lifecycle from design to retirement. We theoretically ground the DSR process in this broad understanding of ALM.

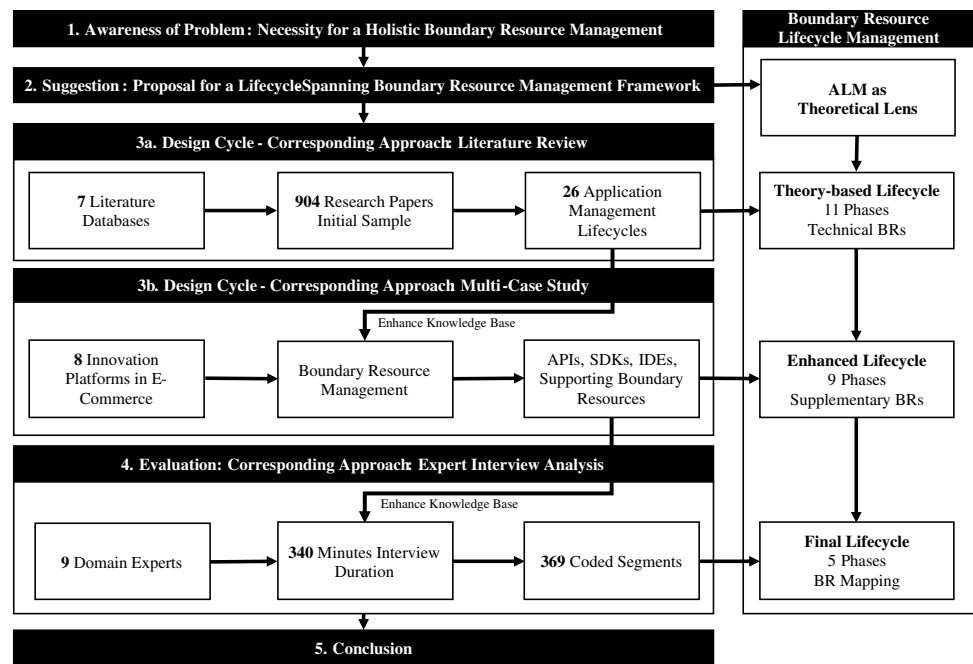
Popular application lifecycles are organized along the following phases, which might carry different names: design, build, deploy, operate, and optimize (AXELOS, 2012). ALM therefore emphasizes the interaction between application development and operation, and describes the interaction as a continuous cycle in which the completed development of a version is followed by deployment and optimization.

This subsequently leads to new requirements for the next development iteration and the possible decommissioning of the existing application. We provide an overview of ALM phases covered in selected research papers in Table 5 in the Appendix. This phased approach guided the development of the initial boundary resource lifecycle (Table 3).

Design science research approach

This paper aims to provide a theoretically grounded and empirically validated reference procedure model for boundary resource management in platform-centered innovation ecosystems. We identified a need for a holistic management of boundary resource as an important building block of innovation ecosystems in e-commerce (Boudreau, 2010; Santos & Eisenhardt, 2005). Developing and applying a designed artifact to better understand a problem and generate knowledge is the paradigm underlying DSR (Hevner, 2007; Hevner et al., 2004). The boundary resource management lifecycle as a developed artifact should support owners of innovation platforms in developing and implementing proper boundary resources to attract and orchestrate third-party developers (problem space) (Drechsler & Hevner, 2018; Vom Brocke et al., 2020). The attraction and governance of third-party developers should increase the focal platform’s overall generativity (Blaschke & Brosius, 2018). The boundary resource management lifecycle as a reference model is a meta-artifact (Gregor & Jones, 2007) that can be instantiated for the peculiarities of specific innovation platforms and that can guide boundary resource implementation (solution space). For the development of our meta-artifact, we follow strategy one as described by Iivari (2015).

We applied the DSR approach by Vaishnavi et al. (2019), focusing on the design phase, and developed a lifecycle model for the holistic management of boundary resources in innovation ecosystems. In total, we performed two design cycles in which the latter design iteration is informed by the previous one for an incremental development (Baskerville et al., 2019; Diederich et al., 2020; Norbjerg, 2018; Rosemann & Vessey, 2008). The theoretical foundation of our lifecycle model lies in the ALM literature. Hence, we conducted a systematic literature review to identify relevant lifecycle phases that involve boundary resources in the first design cycle (a) (Vom Brocke et al., 2009). We informed the theory-based lifecycle with a multi-case analysis on selected innovation platforms in e-commerce ecosystems and their communication of boundary resource developments in the second design cycle (b) (Yin, 2018). We evaluated the enhanced lifecycle with nine interviews with domain experts employed at innovation platforms in e-commerce. The evaluation was used to investigate the designed artifact’s applicability

Fig. 1 Overview of the design science research approach

and to further inform the design process with data on the boundary resources' internal design and implementation (Alsaawi, 2014; Myers & Newman, 2007).

To ensure a holistic management of boundary resources from design to retirement, we apply a lifecycle management perspective throughout the two design cycles (ISO, 2017; Kääriäinen & Välimäki, 2009). Since our perspective on boundary resources is focused on a broad understanding of the lifecycle concept, the literature review (i.e., coding of the papers), the case data, and the interview analysis are investigated through a theoretical lens developed during the elaboration on ALM (Niederman & March, 2019). This especially involves the elicitation on the lifecycle phases and their impact on boundary resources. In the following sections, we describe in more detail how the three methods intertwine conceptually to develop the lifecycle model for boundary resource management, and we provide details on the intermediary lifecycle models. The overall DSR approach is visualized in Fig. 1.

Systematic literature review

The first design cycle aims to analyze the existing literature on ALM in general and their relation to boundary resources in particular to derive a theory-based lifecycle model for the management of boundary resources. A systematic literature search offers an appropriate framework to ensure the results' traceability, systematicity, and reproducibility, with the literature search involving the application of process and quality criteria (Cram et al., 2020). In line with these principles, we apply vom Brocke et al.'s (2015) approach for literature analysis in combination with that of Webster and Watson

(2002). The foundation of the literature search was the conceptualization of the object under consideration, i.e., the application lifecycle management as established in the previous chapter, and its consideration of boundary resources. We conducted a broad search for application lifecycles and their respective management. On September 5, 2022, we searched seven literature databases (i.e., ACM Digital Library, AISel, EBSCOhost, IEEE Xplore, ScienceDirect, Scopus, and the Web of Science) for title, abstract, and keywords. Our search resulted in an initial sample of 904 research papers (Fig. 2).

To ensure an appropriate level of quality, we conducted two phases of selection. First, we excluded non-English language articles, panels, and commentaries. Second, we applied additional quality criteria to the selection process. We excluded the following research papers: those that describe the lifecycle of an application at runtime, those with a narrow perspective on ALM, those that do not consider boundary resources, those with a focus on software development, and those that focus on single phases of the lifecycle without considering an overarching lifecycle. We followed Bandara et al. (2015) approach to filter the initial literature sample based on title, abstract, and content, using our quality criteria. After the exclusion of duplicates, our analysis led to 18 papers in which the application lifecycle forms a broad perspective in the subject under discussion. We augmented this set of literature with literature resulting from a one-way backward and forward search adding eight research papers for a total of 26 research papers in the final set (vom Brocke et al., 2015). The team of researchers analyzed the articles within the sample independently using full-text screening, and they coded the relevant text passages to extract the information on lifecycle phases. We applied

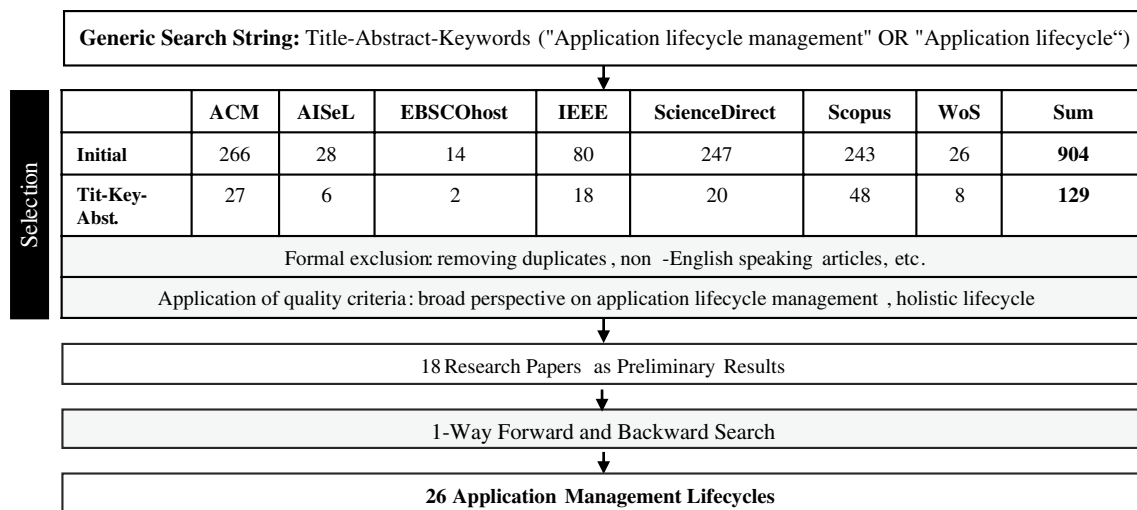


Fig. 2 Overview of the literature selection process

the generic application lifecycle phases (AXELOS, 2012) as initial coding scheme for the extraction of knowledge on the phases, and iteratively derived the theory-based lifecycle model. We summarized the research articles identified and analyzed, in a concept matrix presented in the Appendix (Webster & Watson, 2002). The identified research articles covered a period of 14 years, with the first article published in 2008. The number of lifecycle phases varies between three and ten, with an average of six lifecycle phases (Table 4).

The theory-based boundary resource lifecycle consisted of a total of 11 lifecycle phases derived from the literature review approach (Table 4). These phases follow each other successively, with the last phase (i.e., governance) leading to the first phase of another cycle run (i.e., requirements). This structure follows the structure of existing application lifecycles (e.g., AXELOS, 2012; Duda et al., 2022; Ebert, 2012; Khabou et al., 2019). Since ALM is concerned with the management of a software artifact, the theory-based lifecycle phases focused on technical boundary resources, such as APIs and IDEs (AXELOS, 2012; Bondel et al., 2021; Vester, 2018). Supplementary boundary resources are rarely discussed as results of single phases (e.g., test and release phase) or main activity of the communication phase (Kääriäinen, 2011; Matthes et al., 2009). However, the management of boundary resources does not only involve technical boundary resources (Dal Bianco et al., 2014; Hein et al., 2020; Star, 2010). Therefore, we enhanced the theory-based lifecycle with knowledge derived from innovation platforms successfully attracting third-party developers and a high generativity indicated by the number of extensions available.

Multi-case study

To enhance the theory-based lifecycle with prescriptive knowledge derived from real-world cases, we conducted a multi-case analysis on eight innovation platforms (i.e.,

BigCommerce, Magento Commerce, OXID eShop, Salesforce Commerce Cloud, SAP Commerce Cloud, Shopify, Shopware, and WooCommerce) in e-commerce ecosystems (Eisenhardt, 1989; Yin, 2018).¹ These innovation platforms form the center of larger innovation ecosystems in the context of e-commerce (Table 1). They provide the necessary application services for business models in e-commerce (Aulkemeier et al., 2016b; Timmers, 1998; Wulfert & Karger, 2022) and provide an extensible codebase for third-party developers enabling the provision of platform extensions (Jansen & van Capelleveen, 2013; Tiwana et al., 2010). As we focus our multi-case analysis on innovation platforms, we did not consider Amazon with its leading transaction platform Amazon Marketplace (Amazon, 2022).²

The multi-case study approach enabled us to derive, more broadly, information on boundary resource management from a series of cases that might differ in environmental aspects but share a common phenomenon (Yin, 2018). Moreover, selecting multiple cases is paramount in identifying cross-case patterns, and represents the basis for comparative analysis (Eisenhardt, 1989; Van Aken, 2004). It is an essential approach to transcend the specificity of each individual case and abstract generally applicable knowledge that benefits the class of the artifact (Gregor et al., 2013;

¹ The asterisks in table 1 have the following meaning: * = Year of publication of the SAP Cloud Platform and ** = Values refer to the OXID eSales AG.

² Although Amazon is sometimes described as a hybrid platform in research because of its subsidiary Amazon Web Services providing cloud services to professional as well as private customers (Amazon, 2022; Schütte & Wulfert, 2022), Amazon does not provide its e-commerce system, powering the Amazon Marketplace, for use at standalone sites or the involvement of third-party developers. For the Amazon Marketplace, only APIs for sellers are available (Github, 2023).

Table 1 Meta-information on the selected innovation platforms

	Big commerce	Adobe commerce	OXID eShop	Salesforce commerce cloud	SAP commerce cloud	Shopify	Shopware	Woo commerce
Establishment	2009	2008	2003**	1999	2012*	2006	2004	2008
Number of employees	1300	375	51–200**	73,000	105,000	10,000	300	--
Total turnover (million euro)	219.9	107.1	--	26,900	27,804	46,848	73	--
Customers	Fujitsu Ben & Jerry's Gillette General Electric	HP Liebherr Shoebacca Rossignol	AIDA Bitburger Hellweg Mercedes Benz	Adidas Amazon Web Services Barclays Coca Cola	Tesco Casey's Moen The Enter-tainer	Freelatics Babbel Heinrich Five Skincare	Aston Martin Borussia Dortmund Melitta Thyssenkrupp	Chaka Khan All Blacks Weber Singer

Lee et al., 2011). Hence, we could analyze the “same phenomenon in [...] in a variety of situations” (Yin, 1981, p. 101). The shared phenomenon of our eight cases is their success in e-commerce measured by the number of supported e-commerce websites (builtwith, 2022; Datanyze, 2021), their design as an extensible codebase enabling third-party developers to design third-party extensions (Tiwana et al., 2010), and as a provider of application services for e-commerce businesses (Wulfert & Karger, 2022). We have selected these platforms to derive knowledge from successful boundary management and go “beyond a single success story” (Chandra Kruse & Seidel, 2017, p. 186).

For the platform analyses, we leveraged a plethora of primary and secondary data sources to collect architectural information on each case that is publicly available (Hänninen et al., 2018). First, we searched for information on the official websites, including textual information, graphics, and videos of the eight innovation platforms. In this part of the analysis, we tried to limit ourselves to objective information. We critically analyzed advertising texts or content that aimed at making a profit and did not include them in our analysis if we had any doubts. We also analyzed information and texts from other media that dealt with the eight platforms, such as newspapers, industry journals, and practitioner magazines. The case data is analyzed through the theoretical lens of lifecycle management for boundary resources (Niederman & March, 2019), with the objective of synthesizing a holistic management model.

For the development of the enhanced boundary resource management lifecycle, we aggregated the information on boundary resource management derived from our multi-case analysis in a cross-case approach (Yin, 2018) and updated the theory-based lifecycle accordingly. In particular, the case analysis revealed a number of potential supplementary social boundary resources, such as quality criteria, status cockpit, and deprecation schedule, that augment the design, development, publication, and maintenance of

technical boundary resources (Adobe, 2022c; SAP, 2022c; Shopify, 2022e). Consequently, we were able to decouple the lifecycle of technical and supporting boundary resources while retaining a lifecycle phase-based relation between them (Fig. 3). Moreover, the multi-case analysis emphasized the significance of boundary resource governance and communication when it comes to the management of boundary resources in innovation ecosystems (BigCommerce, 2022b; OXID, 2021). We therefore introduced the layered structure of the boundary resource management lifecycle, with governance and communication as core layers. We also introduced the description continuous to the communication phase in order to linguistically emphasize the significance of the communication between the owner of a focal platform in innovation ecosystems and third-party developers providing platform extensions (Jansen & van Capelleveen, 2013; Staykova & Damsgaard, 2015). The number of lifecycle phases for the management of boundary resources was reduced to nine, with boundary resource governance and continuous communication as core activities.

Expert interviews

We evaluated the enhanced boundary resource management lifecycle with a series of interviews with domain experts (Schreier, 2014). Within the interviews, we addressed the implementation and management of boundary resources on selected innovation platforms in the context of e-commerce. We chose to conduct semi-structured interviews to directly address the knowledge holders within the domain. In total, we conducted nine expert interviews with an average duration of 38 min (Table 2). To create a broad and diverse knowledge base, we generalized information from a diverse set of interviewees (Iivari et al., 2021). Within the data collection process, we contacted a total of 46 innovation platforms based in Europe, Latin America, and the USA through

publicly available information of the companies and profiles of platform representatives on professional social networks such as LinkedIn. Although we mainly contacted platform representatives at management level, we were forwarded to interviewees directly involved in the design and development of (technical) boundary resources. We classified the interviewees by their company's internal role according to innovation platform layers identified by Zutshi and Grilo (2019), and by related roles, such as business process owner, user interface designer, and internal platform developer (Wulfert et al., 2022). We could acquire an almost equal number of roles for the interviews (Table 2). As requested by the interviewees, we did not mention any company names.

Our semi-structured interviews were based on a previously known guide to ensure that all conversations covered a similar range of questions regarding the management of boundary resources while allowing the greatest possible flexibility in exploring explicit knowledge (Merton & Kendall, 1946; Misoch, 2015). The interview guide was created based on our research question, using ALM as the theoretical lens, and the enhanced boundary resource management lifecycle. We were able to discuss all lifecycle phases of the enhanced lifecycle and go into further detail for the supplementary boundary resources. The expert interviews were conducted via electronic communication media, between December 2021 and June 2022. Depending on the interviewees' preferences, the interviews were conducted either in English or in German. Interviewees 2 and 7 provided additional textual answers for our interview guide. We introduced each interviewee to the topic and the purpose of the study at the beginning of the interview (Alsaawi, 2014).

For the knowledge extraction, the interviews were anonymized and transcribed to perform a qualitative content analysis (Nunkoosing, 2005), which aims at systematically analyzing an object of communication regardless of its form (Schreier, 2014). Specifically, we used the content structuring approach as a form of deductive category assignment to extract information according to predefined categories (Mayring, 2014). We used our enhanced boundary resource lifecycle as the initial categorization system and iteratively adapted and expanded this scheme with our

nine expert interviews. We reached a stable set of codes after the first five interviews. We chose a single sentence as the coding unit, thus ensuring the highest possible level of detail, while a paragraph of an interview can be considered as a context unit (Mayring, 2014). Multiple codes can also be applied to a single sentence if the sentence has several meanings. We used MAXQDA for the coding (Krippendorff, 2013) and the subsequent construction of a quantitative graphical representation (Kuckartz & Rädiker, 2022). In total, we coded 369 segments (Fig. 1). We include leading quotes from the interviews in the results section (Yin, 2018).

The coding of the interview transcripts resulted not only in an aggregation of the lifecycle phases for boundary resource management, but also in a more detailed allocation of supplementary boundary resources to these phases as depicted in Fig. 3. The small-stepped and detailed phase descriptions seemed not to be applicable in practice where, for instance, plan, design, and development activities happen even iteratively yet in an agile manner (#2). In total, the final boundary resource management lifecycle consists of five phases for technical boundary resources (Fig. 3). Two interviewees made an even broader segmentation of the boundary resource management lifecycle, describing an internal perspective concerned with the design and development of boundary resources, an external perspective when boundary resources can be used by third-party developers, and a release phase as transition between the internal and external phase (#2, #4). Moreover, the evaluation led to several additions to the supplementary boundary resources and changes in the mapping to boundary resource management phases. For instance, our interviewees frequently reported the necessity for developer-requested changes such that we introduced change requests as supplementary boundary resource (#1, #3, #7). Another example would be the mapping of the supplementary boundary resource "terms of use" that was initially mapped to the release phase during the multi-case analysis. However, the interview coding revealed strong evidence for an even earlier consideration of terms of use already in the design phase and even in the governance in which external responsibilities are determined (#2, #7).

Table 2 Expert interview overview

#	Platform internal role	Staff	Revenue (in \$)	Boundary lifecycle phases	Duration
1	User interface developer	~ 4700	~ 800 M	4.1, 4.2, 4.4, 4.6, 4.7	46 min
2	Internal platform developer	~ 2500	~ 5 M	4.1, 4.2, 4.3, 4.2, 4.7	20 min
3	Internal platform developer	~ 49,000	~ 17.1 B	4.3, 4.4, 4.6, 4.7	20 min
4	Business process owner	~ 270	~ 28 M	4.1, 4.2, 4.4, 4.6, 4.7	58 min
5	Business process owner	~ 107,000	~ 28 B	4.1, 4.3, 4.5	42 min
6	Business process owner	~ 107,000	~ 28 B	4.1, 4.5	42 min
7	Internal platform developer	~ 40	n.a.	4.1, 4.4, 4.7	33 min
8	Internal platform developer	~ 107,000	~ 28 B	4.1, 4.4, 4.5	34 min
9	User interface developer	~ 107,000	~ 28 B	4.1, 4.2, 4.3	45 min

Table 3 Overview of the development process of our boundary resource management lifecycle

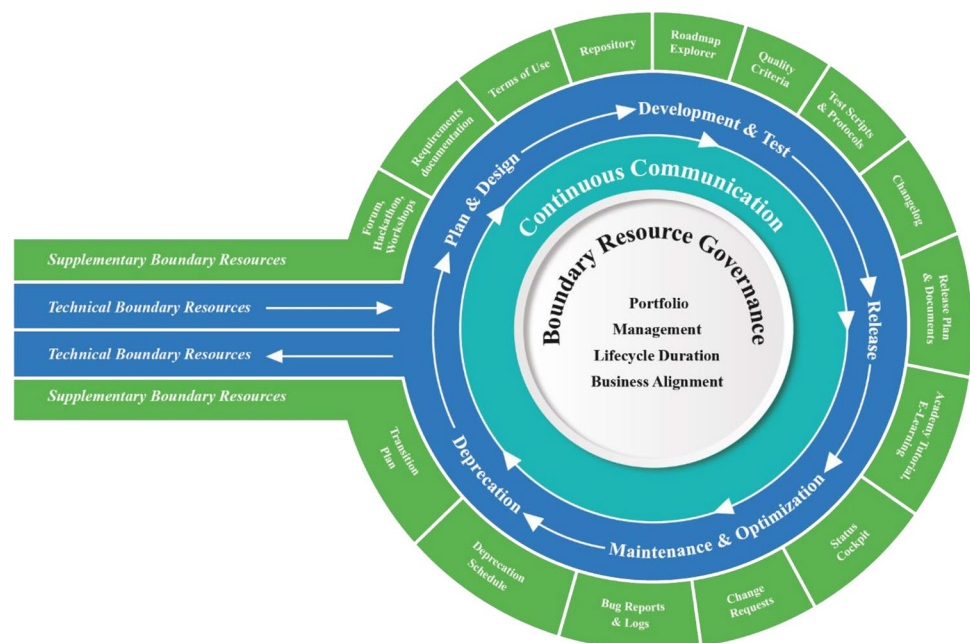
	Boundary resource management lifecycle		
	Theory-based	Enhanced	Final
Input data	26 application management lifecycles	8 innovation platforms in e-commerce	9 interviews with domain experts
Method for data collection and analysis	Systematic literature analysis (vom Brocke et al., 2015) and concept matrix (Webster & Watson, 2002)	Multi-case analysis (Yin, 2018)	Semi-structured interviews (Misoeh, 2015) and qualitative content analysis (Mayring, 2014)
Lifecycle stages	11 (requirements, plan, design, development, test, release, maintenance, optimization, deprecation, communication, governance)	9 (requirements, plan, design, development, test, release, maintenance, optimization, deprecation) 2 cross-stage activities (communication, governance)	5 (plan and design, development and test, release, maintenance and optimization, deprecation) 2 cross-cross stage activities (Boundary resource governance, continuous communication)
Development notes	<ul style="list-style-type: none"> • Focus on technical boundary resources • Boundary resource development and operation 	<ul style="list-style-type: none"> • Supplementary social boundary resources • Mapping of boundary resources • Layered structure 	<ul style="list-style-type: none"> • Aggregation of lifecycle stages • Additional supplementary social boundary resources • Updated mapping of boundary resources

The iterative development of our boundary resource management lifecycle resulted in two intermediary lifecycles and a final boundary resource management lifecycle (Fig. 3). We summarize the development process in Table 3.

Boundary resource management lifecycle

Based on our DSR approach, which is theoretically grounded in the current body of literature on ALM and involves empirical data from eight case analyses triangulated by nine interviews with domain experts, we developed a reference procedure model for the management of boundary resources in innovation ecosystems for the domain of e-commerce.

We take a lifecycle perspective to ensure an adequate management of boundary resources from design to retirement (AXELOS, 2012; ISO, 2017). The boundary resource management lifecycle resembles a reference procedure model for the particular context of e-commerce ecosystems (Jörg Becker & Delfmann, 2007; R. Schütte, 1998). The core of the boundary resource management lifecycle is formed by governance activities, illustrated with a light grey shading, spanning the whole lifecycle, and includes portfolio management, responsibility management, business alignment, and the determination of the lifespan for each boundary resource (Chappell, 2008; Vester, 2018). The core of the boundary resource management lifecycle is surrounded by activities for enabling a bidirectional communication with ecosystem

Fig. 3 Boundary resource management lifecycle

participants, such as third-party developers and service providers, across the entire lifecycle depicted in turquoise. As suggested by ISO (2017), our proposed boundary resource management lifecycle is “organized into stages.” We shaded the lifecycle for managing technical boundary resources in green. The lifecycle for technical boundary resources is divided into five phases, namely plan and design, development and test, release, maintenance and optimization, and deprecation. We also mapped major supplementary boundary resources to each lifecycle stage of the technical boundary resources in the outer circle shaded in blue (Fig. 3). The differentiation of technical and supplementary boundary resources is in line with findings from our case and interview analysis (e.g., Shopify, 2022b, #7, #10). Their intertwined nature is reflected by visually integrating both circles (Fig. 3).

Boundary resource governance

The owner of the focal platform in a platform-centered ecosystem “sets, and often enforces, the governance rules” (Adner, 2017, p. 48). These rules are direct and indirect measures to control innovation ecosystems and their participants (Tiwana, 2014). As part of the ecosystem governance that involves “platform regulation including contractual, technological and information design” (Boudreau & Hagiu, 2009, p. 187), boundary resource governance defines the portfolio of necessary boundary resources in coordination with business requirements, the necessary boundary resources’ lifetime, overall procedures, and responsibilities (Tüzün et al., 2019; Vester, 2018). The portfolio represents the entire set of past, present, and future boundary resources. The coordination of business requirements with the boundary resource portfolio defines the openness of the platform (Boudreau, 2010; Hein et al., 2016). “*Before designing new boundaries, internal and external responsibilities need to be clarified and communicated. Who is responsible for what? Which services do we need to provide as a platform provider? Which application services are open for developers?*”³ (#9).⁴ Boundary resource governance can also promote compatibility of an innovation platform’s boundary resources with other platforms, thus increasing the reach for third-party developers (Spaeth & Niederhöfer, 2022).

The governance of boundary resources is an overarching task for the management of single boundary resources and orchestrates development and maintenance

tasks (Chappell, 2008; Tüzün et al., 2019). Our analyses revealed three types of platform release cycles that also involve the boundary resources. While the majority of analyzed platform cases favor scheduled quarterly releases with fixed boundary resource lifecycles (Adobe, 2022d; OXID, 2021; Shopify, 2022e), unscheduled release cycles with prior announcements are also implemented in practice (#1, #2). Another approach introduced by interviewees #4 and #7 prescribes a continuous evolution of boundary resources without the necessity for deprecation; however, interviewee #4 also highlighted backward compatibility as a major prerequisite. “*We generally operate on a pattern of backward compatibility. Eventually, there’s something that can’t be, but we can really try not to deprecate things*” (#4). “*To avoid versioning, we ensure that our API is backward compatible. This is part of our API contract. This also results in a number of API fields that are unused.*” (#7)

In this regard, breaking and non-breaking changes were introduced by several innovation platforms (BigCommerce, 2022b). While breaking changes usually misplace communication with external applications and result in a new API version, non-breaking changes can be implemented into production systems without any interference for developers. However, third-party developers are encouraged to “write code against our APIs that will not break if an endpoint begins returning additional fields” (BigCommerce, 2022b). Adobe also highlights backward-incompatible changes in a dedicated section of their developer blog (Adobe, 2022a). OXID (2022d) uses a three-tier application landscape consisting of a development, maintenance, and legacy environment. The legacy environment is supported with updates until the next maintenance version is released. The development environment is one version ahead of the maintenance version that is productively used by customers and extension developers. SAP Commerce Cloud makes use of development and quality assurance environments before a boundary resource is made available in productive systems (#5, #9). Four interviewees also reported on a pipeline of staging systems with intermediary quality gates (#1, #5, #6, #8).

Versioning is a common approach that owners of innovation platforms use in order to introduce new and updated APIs while continuing old ones for a predefined transition period (BigCommerce, 2022a; SAP, 2022d; Shopify, 2022e; WooCommerce, 2022c). Versioning can also be used to map possible dependencies between technical boundary resources and external tools (Adobe, 2022e; Shopware, 2022d). Semantic versioning can also give meaning to version numbers by emphasizing major, minor, and patch versions in a single string (Preston-Werner, 2022).

³ This and the following quotes are illustrative extracts taken from the expert interviews conducted.

⁴ The numbers in brackets refer to the expert interview overview in Table 2 and indicate single interviewees.

Continuous communication

Continuous communication deals with the continuous information of developers and additional ecosystem participants, with general information about boundary resources (Priyadarsini et al., 2020). Developers depend on timely information about the status of technical boundary resources, as the extensions they develop make use of application and development services provided via boundary resources that are provided by the innovation platform. Communication in this context means publishing supplementary boundary resources to ecosystem participants and includes information such as news, advantages and disadvantages, projects, roadmaps, new services, and other information (Vester, 2018). This activity is neither tied to a lifecycle phase nor to a single boundary resource release (AXELOS, 2012). Developers and additional ecosystem participants are frequently informed about the status of the boundary resources independently of single lifecycle phases and boundary resource releases. *“One of the biggest challenges is always communicating.”* (#1)

Supplementary social boundary resources are used for the continuous communication layer of our boundary resource management lifecycle (Dal Bianco et al., 2014). Communication should be ahead with regard to technical boundary resources' time of actual release cycles to allow third-party developers to cope with upcoming changes (Schwarz & Legner, 2020). Depending on the type of technical boundary resource and lifecycle stage, it is possible to create different supplementary boundary resources and publish them for communication (Dal Bianco et al., 2014; Hein et al., 2020; Schwarz & Legner, 2020). Besides extensive textual documentation (OXID, 2022a; SAP, 2022d; WooCommerce, 2022f), supplementary boundary resources can also involve developer forums (WooCommerce, 2022e), animated guides, videos (SAP, 2022a), communities and forums (Shopify, 2022d), and hackathons (BigCommerce, 2022g). Boundary resource parameters and documentation are often available via internal or external repositories such as GitHub (BigCommerce, 2022c; Github, 2022a, b, c). Product release notes often contain information on boundary resources for developers (Salesforce, 2022e). *“If you are active in this ecosystem, then you get information on all channels and, additionally, if you are active as a developer on that platform, then there is a special mailing [list]”* (#4). Platform providers can also offer certification programs for partners and developers (WooCommerce, 2022a) and even academies for training developers in the use of boundary resources, including online training courses and blended learning (OXID, 2022b). Smaller platforms also favor direct contact for the communication of changes on boundary resources. *“We also go to our developers in some cases. If we have only a few developers who are now using an old resource, just as*

we want to deprecate it, then before I tell them it was in the release notes, I'd rather go and talk to them” (#4).

Technical boundary resources include development and application boundary resources (Dal Bianco et al., 2014) that are necessary for the development of platform extensions by third-party developers. The management of these boundary resources involves all stages, from requirements management and design to their deprecation (Deuter & Rizzo, 2016; Patni, 2017). In the following sections, we integrate the supplementary boundary resources used in the boundary resource management lifecycle stages.

Plan and design

The plan and design stage begins with the determination of requirements for boundary resources (Hallerstede et al., 2011; Vester, 2018). During our analysis, we identified business requirements and developer demands as major sources of requirements. While the former reflects the innovation platform's strategic focus as part of the governance activity's business alignment, the latter requires a close communication with the developers of platform extensions (Bondel et al., 2021; Hallerstede et al., 2011). Developers can make demands for new boundary resources via workshops, hackathons, feature forums, and direct contact with business process owners (BigCommerce, 2022f; Bondel et al., 2021). The governance decision needs to be reflected in order to balance openness and control in boundary resource design (#5). Demand management deals with the collection, analysis, and processing of requirements demanded by developers. These requirements can be collected via the previous iteration of the lifecycle for an existing boundary resource and proactively collected for a new boundary resource (Mohagheghzadeh & Lindman, 2022). *“During the project work, we start with a discovery research phase that allows us to map out the business requirements, the user needs, and from that, we create platform features and boundary resources”* (#2). The requirement definition involves the formulation of functional and non-functional requirements. While the former describes the behavior of the boundary resource as demarcation objects between the focal platform and third-party extensions, the latter defines the intended performance with regard to service-level agreements (Akana, 2020).

After the requirements have been identified and documented (Bondel et al., 2021), they can be translated into feature specifications, business cases can be derived for them, and dedicated boundary resources fulfilling these requirements can be planned (Deuter & Rizzo, 2016; Hallerstede et al., 2011). The boundary resource specification is created and harmonized with the endpoint structure and behavior (Bondel et al., 2021). For technical boundary resources traversing another iteration of the lifecycle, it is necessary

to determine which parts of the boundary resource must be updated (#3). Technical boundary resources can only be made available for a specific group of developers, for instance, for gold and silver developers. It is important to consider developers' flexibility and offer technical boundary resources for a number of programming languages to allow developers to flexibly use their favorite language when implementing extensions (Salesforce, 2022b). Since boundary resources and the associated terms of use are inseparable, this aspect is a relevant supplementary boundary resource (#2, #9). The focus of supplementary boundary resources, such as guidelines and regulations, is on accumulating and organizing necessary information.

Development and test

The previously designed technical boundary resources are implemented and tested (Bondel et al., 2021), while supplementary boundary resources are finalized in this stage (Chappell, 2008). The process of developing and testing can take an iterative form (#3). Since contemporary IS are complex, software artifact development including boundary resources is conducted in teams requiring shared repositories and code orchestration (#4). *"For development, each boundary resource has its own repository"* (#2). The development of extensions usually takes place in preconfigured platform environments and integrates necessary boundary resources (Salesforce, 2022a). Before their release to production environments, boundary resources can be flagged as beta. *"We often release boundary resources as a beta first. And then it's kind of a beta component for three months, and during beta we're allowed to deprecate and change things"* (#4). The development of technical boundary resources can be steered by a roadmap overview for products (Shopware, 2022c; WooCommerce, 2022b) and by a detailed roadmap explorer for new boundary resources (Salesforce, 2022d; SAP, 2022e) that can be communicated to ecosystem participants. *"We provide a roadmap on our documentation site, where we share information about features we are implementing or planning to implement"* (#2).

Testing of technical boundary resources involves the investigation of functionality, performance, and behavior (Bondel et al., 2021). Automated and standardized tests of boundary resources ensure sophisticated testing for each boundary resource and each version (#1). This testing ensures a higher quality of extensions and stability of technical boundary resources (Priyadarsini et al., 2020, #3, #9). *"Our marketplace template for e-commerce includes a suit for automated tests. Both units and end-to-end tests involving our boundary resources"* (#2). Innovation platforms usually formulate quality criteria for boundary resource implementation and testing (Adobe, 2022c) and provide a rich set of methods and tools for testing extensions

that are also used for testing boundary resources, such as environments for testing API calls and responses (Adobe, 2022f; SAP, 2022c). Testing involves unit tests, integration tests, and dedicated tests for the usage of technical boundary resources (#7). Staging in sandbox environments to test extensions and the correct use of API calls is a common approach for innovation platforms (BigCommerce, 2022e). *"We have a dedicated, free-of-charge staging environment, where third-parties can develop and test their platform extensions manually and automatically"* (#2). Supplementary boundary resources are checked for their harmonization with their technical counterparts and overall communication strategy (#8).

Release

In the release stage, previously developed and tested technical boundary resources are deployed into production environments and can be accessed by developers (Chappell, 2008). This release is accompanied by the publishing of supplementary boundary resources, such as release plans, documents, and changelogs, enabling and supporting the use of technical components. In addition to these written documents, multi-media tutorials and e-learning material can be provided (OXID, 2022a; SAP, 2022d; WooCommerce, 2022f). The platform owner needs to ensure that external developers accept and use the technical boundary resources (Hallerstede et al., 2011). To coordinate release activities and harmonize boundary resource types, our interviewees consider a deployment strategy mandatory. *"You definitively need to implement a deployment for internal and external coordination"* (#5). The release of technical boundary resources is often part of continuous deployment pipelines (Bondel et al., 2021; Priyadarsini et al., 2020). For the release of boundary resources and features, innovation platforms can implement a transportation route from development and quality assurance environments to production systems, that needs to be followed, with dedicated quality gates between these environments (#6, #8). The platform provider can also decide to release more complex boundary resources in several iterations in subsequent releases (Tang et al., 2011). In addition to the release of a new stable version, innovation platform owners may also grant access to pre-release versions to provide developers with the ability to test the developed extensions' compatibility with forthcoming versions of the innovation platform (WooCommerce, 2022d).

Maintenance and optimization

In the maintenance and optimization stage, the boundary resources are operated in production environments (Bondel et al., 2021; Tang et al., 2011). This stage includes

tasks, such as monitoring the status of technical boundary resources and collecting, categorizing, and evaluating developers' feedback (Chung & Kim, 2010). The task is therefore the structured analysis of all feedback (Deuter & Rizzo, 2016). *"Metrics collected automatically during platform use serve as evidence of the design and implementation quality of existing and future boundary resources"* (#7). Our multi-case analysis revealed that, for maintenance tasks, status pages for monitoring the APIs' and developer services' operability (BigCommerce, 2022d; Shopify, 2022f) and sophisticated cockpits for monitoring the status of single APIs that are even customizable for the APIs that developers use (OXID, 2022c; Salesforce, 2022c) help minimize the effort for bug fixing and optimizations (#3, #7). *"Providing a status page for the platform and the boundary resources used is an important trust-building and support-reducing measure"* (#7). Platform providers also provide tools and functions for fixing bugs in using boundary resources (Shopware, 2022a). *"APIs can report their status, utilization, and utilizing ecosystem participants"* (#4). Boundary resource downtime and bugs can be communicated in dedicated forums (WordPress, 2022). The behavior of boundary resources can be reproduced by implementing logging mechanisms (SAP, 2022b). *"We do monitor, for example, the status of our systems and APIs as well. If we see some issues with a technical boundary resource, we have a process in place that we go investigate and make sure we fix it before it becomes a system critical issue impacting operation"* (#1).

While maintenance activities aim at correcting issues of boundary resources in use, optimizations try to avoid problems and improve the performance of boundary resources. Depending on the requested changes, feedback can be used to incrementally optimize the boundary resource, or the feedback can be implemented in a future lifecycle in a new boundary resource version (Hallerstede et al., 2011). Incremental optimizations are deployed into the current version of the boundary resource (Chung & Kim, 2010). Optimizations can affect the innovation platform's openness and control characteristics positively, but also negatively. Therefore, it is part of change management to analyze an optimization's potential benefits and risk for the overall ecosystem. Platform providers introduced a priority concept for boundary resource optimizations, indicating the necessity of updating and the vulnerability to external fraud (Adobe, 2022b). Since optimizations may involve numerous application services, a team of internal platform developers needs to handle optimizations (#1).

Deprecation

In the deprecation stage, the unnecessary, unsafe, and outdated versions of boundary resources are indicated for deprecation and retired (Bondel et al., 2021; Shopify, 2022e).

Deprecation practices vary depending on the type of boundary resource. Deprecating technical boundary resources requires a migration plan to upgrade to a new or updated resource (Shopware, 2022b; Tang et al., 2011). Deprecations are announced in developer forums (Shopify, 2022c) and dedicated boundary resource release notes (Shopify, 2022a). *"We mark deprecated features in the documentation along with links to documentation topics that describe how to achieve the functionality after the deprecation"* (#2). The deprecation stage includes the planning of deprecation schedules, the implementation of transition plans, and the management of adequate communication of the deprecated boundary resource to developers (Shopify, 2022e). *"For new [boundary resource] releases, there are always release notes, and deprecations are announced via various channels"* (#3). Boundary resources of older innovation platform versions are likely to be deprecated after a transition period (Shopify, 2022c). *"Whenever new boundary resources are implemented, what we usually do is start a phase-over process where we start deprecating the old module and move it into the new module in a phased approach, shifting the utilization to the new resource"* (#1). If a boundary resource is indicated as deprecated, it can be unpublished and removed in a following release (Chung & Kim, 2010; Patni, 2017; Shopify, 2022e; Shopware, 2022b). Deprecating technical boundary resources is a more complicated endeavor for smaller platforms, who tend to keep old but still used boundary resources. *"Unfortunately, we also have to keep many old boundary resources because they are still in use. We can't migrate them"* (#1). Keeping old building blocks of a boundary resource can also be used to ensure backward compatibility (#1). If backward compatibility is propagated, boundary resources can only be deprecated if they are no longer used at all (#4).

Discussion

We developed a theoretically grounded and empirically validated lifecycle model for the management of boundary resources in innovation ecosystems. Innovation ecosystems with sophisticated boundary resources and a critical mass of developers can cope with the contemporary challenges of traditional IS development. Boundary resources can be differentiated between technical boundary resources and supplementary boundary resources (Dal Bianco et al., 2014; Ghazawneh & Henfridsson, 2013). We developed the reference procedure model (Jörg Becker & Delfmann, 2007) in a DSR approach consisting of two design cycles. The first design cycle utilized a review of the existing ALM literature as input for deriving suitable lifecycle stages of boundary resource management (vom Brocke et al., 2015). The second design cycle investigated eight case studies of innovation

platforms in e-commerce to derive boundary resource management activities and supplementary boundary resources for each lifecycle stage (Yin, 2018). We evaluated these results with the analysis of interviews with nine domain experts employed at innovation platforms in e-commerce (Misoch, 2015). We developed two intermediary lifecycle models (i.e., theory-based lifecycle, enhanced lifecycle) to come up with the final boundary resource management lifecycle consisting of five lifecycle phases (Fig. 3). With our reference procedure model, we not only depict the lifecycle of boundary resources in innovation ecosystems but also prescribe (i.e., theory for design and action) how owners of innovation platforms should manage these boundary resources (Gregor, 2006). The “choice of lifecycle models for the design process influences what it is that is designed” (Clemmensen et al., 2008, p. 137). Mapping supplementary boundary resources to technical boundary resources’ lifecycle stages gives platform owners guidance on which documents to create and publish (Dal Bianco et al., 2014; Ghazawneh & Henfridsson, 2010). In this regard, third-party developers can be addressed directly.

Critical appraisal

We chose e-commerce as the domain of interest, as innovation ecosystems in e-commerce are very dynamic in nature, with participants (e.g., customers, developers) joining and departing (McIntyre & Srinivasan, 2017; Wulfert et al., 2021). This requires a diverse set of application services to support the variety of business models, such as electronic shops, digital marketplaces and electronic auctions (Alt, 2020; Timmers, 1998; B. Yoo & Jang, 2019), and has a continuous stream of new extensions and updates (Ghazawneh & Henfridsson, 2013; Medjaoui et al., 2021). Since we abstracted from the peculiarities of eight cases and nine additional interviews that include “a range of possible real situations” (Kosiol, 1964, p. 758), it is possible to consider our boundary resource management lifecycle as a reference procedure model (Delfmann, 2006; Fettke & Loos, 2002; R. Schütte, 1998; Vom Brocke, 2015). The boundary resource management lifecycle also contains variants for a number of activities and decisions that can be selected when instantiating and implementing the reference model for a particular company (R. Schütte, 1998; Vom Brocke, 2015). For instance, we reported on a versioning and a backward-compatible approach for establishing an updated version of an already existing technical boundary resource (#4, #7, Adobe, 2022a). Moreover, the reference model is designed in a flexible manner to be changed and extended for the customization implementation at a particular company.

Besides the mapping of technical boundary resources and supplementary social boundary resources emphasized in our boundary resource management cycle, we integrated boundary resource governance and continuous communication as fundamental activities guiding the development and maintenance of boundary resources (Eaton et al., 2015; Ghazawneh & Henfridsson, 2013; Vester, 2018; Wulfert et al., 2022). To highlight the necessity of governance and communication for the management of boundary resources, we applied a layered approach for the lifecycle model. Boundary resource governance forms the inner layer in which fundamental decisions regarding the boundary resources of an innovation platform, such as the boundary lifecycle duration, versioning standards, responsibilities, and the portfolio of boundary resources, are made (Boudreau, 2010; Ghazawneh & Henfridsson, 2013). The continuous communication layer involves activities relevant for the communication with ecosystem participants (e.g., third-party developers) (Vester, 2018). The continuous communication is responsible for the dissemination of the boundary resources across the innovation ecosystem. Moreover, the interviewees emphasized the importance of adequate communication between platform owners and third-party developers regarding the management of boundary resources (#1, #2, #4, #9). Therefore, we integrated a dedicated layer for continuous communication along the lifecycle of a boundary resource (Bondel et al., 2021; Kääriäinen & Välimäki, 2008; Priyadarsini et al., 2020). We amplified the communication aspect with the mapping of dedicated supplementary boundary resources for the support of the communication of technical boundary resources (Dal Bianco et al., 2014). The outer layers are formed by technical and supplementary boundary resources that resemble the demarcation points for the information and involvement of ecosystem participants (Dal Bianco et al., 2014; Wulfert et al., 2022).

Our multi-case and interview analysis revealed a diverging behavior of platform owners about updating technical boundary resources. While the first group of platforms uses strict versioning, the second group propagates a high level of backward compatibility, avoiding new versions. The former approach makes changes and new developments transparent but potentially requires effort in updating extensions to new versions and adapting the use of tools for developers. The latter approach allows for a parallel usage of boundary resources for different developments and reduces possible adaption efforts for developers but may increase potential boundary resource overheads (#2, #4, #7). A possible explanation for the two different approaches can be found in the varying amounts of power the platform owner can wield on ecosystem participants (Annabelle Gawer, 2021;

Kretschmer et al., 2022). Introducing new versions of boundary resources while simultaneously deprecating the old version force developers to use the new version after a defined transition period (Star, 2010). Platforms with lower amounts of power in their innovation ecosystems would need to provide backward compatibility. The amount of power and control over the ecosystem also prescribes whether the platform owner can define boundary resources on its own or tune distribution by ecosystem participants (Eaton et al., 2015). We account for the ecosystem participants' integration in the boundary resource management lifecycle with the continuous communication layer, with possibilities to post requirements, and by enabling a dialog of business process owners or internal platform developers and developers.

For the development of the boundary resource management lifecycle, we differentiated between technical and supplementary boundary resources (Dal Bianco et al., 2014). These two types include the possible boundary resources as described by Karhu et al. (2018), Star (2010), and Wulfert et al. (2022). We argue that supplementary boundary resources are often bound to a lifecycle stage and a specific version of a technical boundary resource. We therefore provided a mapping of major supplementary boundary resources per lifecycle stage of the technical boundary resource. If the technical boundary resource is created, updated, and deprecated, the explaining supplementary boundary resource also needs to be adapted.

Although our boundary resource management lifecycle seems to have an iterative form, single stages can also be organized in an agile manner (Nerur & Balijepally, 2007; Shore et al., 2022). An example of an agile realization can be the development and test stage, with incremental development of the software artifact and increment tests (#2). These increments can be arranged for the completed boundary resource (Priyadarsini et al., 2020).

Scientific and managerial contribution

We contribute to the current research on ALM by providing a reference procedure model for managing boundary resources. The integrated lifecycle management approach for a holistic management of boundary resources accounts for the strategic role of boundary resources in innovation ecosystems (Bondel et al., 2021; Vester, 2018; Y. Yoo et al., 2010), addresses third-party developers, and fosters an ecosystem's generativity. We provide a novel tool for analyzing and prescribing collaboration in innovation ecosystems. We provide further insights on the role of boundary resource governance and continuous communication as important

activities in managing boundary resources. Moreover, our boundary resource management lifecycle emphasizes the focal innovation platform's role for implementing boundary resources while controlling and governing value co-creation. The reference procedure model's development process also highlights the role of innovation platforms in e-commerce. Boundary resources are especially important in electronic markets, enabling digital collaboration and value co-creation. We decoupled technical and supplementary social boundary resources in line with existing research (Dal Bianco et al., 2014; Engert et al., 2022; Ghazawneh & Henfridsson, 2013; Hein et al., 2020; Wulfert et al., 2022), but emphasized their intertwined nature for ecosystem participants over the whole boundary resource management lifecycle. Technical boundary resources require a supplementary counterpart for efficient ecosystem-wide application. Within our lifecycle model, supplementary social boundary resources are mapped to technical boundary resources per lifecycle stage.

While platform providers can customize this reference model for their individual peculiarities, developers can reproduce the lifecycle of technical boundary resources, anticipate the development of future boundary resources, and benefit from associated supplementary boundary resources. Although the potential impact of continuous software engineering for the software industry is recognized in research and practice, especially for cloud environments (Dittrich et al., 2018; Wilson, 2017), the role of technical boundary resources remains unchanged as stable demarcation points between innovation platforms and third-party developers (Baldwin & Woodard, 2009). However, the accelerated evolution of innovation platforms as a core of innovation ecosystems often causes adjustments for previously stable boundary resources. Hence, the boundary resource lifecycle can be used as a toolkit for boundary resource management when it comes to implementing and communicating changes in interfaces in proper documentation. The reference procedure model for managing boundary resources can therefore be used to cope with evolving innovation and cloud environments in which extensions are "released in small increments and constantly modified" (Norbjerg, 2018, p. 72). The reference procedure model can be applied as an important tool for the development of ecosystems surrounding innovation platforms in e-commerce.

Limitations

This study entails some limitations, which provide opportunities for future research. Since the literature on boundary

resource management is sparse, we started our design science approach with an investigation of the ALM literature to identify relevant lifecycle stages. Although technical boundary resources are software artifacts (Dal Bianco et al., 2014), it might be disputable whether all stages are relevant for the management of boundary resources. We therefore conducted an additional design cycle to refine the lifecycle for boundary resource management with insights from practice derived in a multi-case analysis. However, innovation platforms did not provide detailed information on pre-release lifecycle stages that were considered during our analysis as internal. For information on these internal phases, we especially benefitted from the evaluation of the enhanced lifecycle during our interview analysis. For our interview analysis, we were able to obtain interviews with nine domain experts employed at innovation platforms that provide application services for transaction platforms in e-commerce and online retailers (Aulkemeier et al. 2016b; Wulfert & Karger, 2022). Unfortunately, our interviewees demanded that the results of the interview analysis be published anonymously without any reference to the innovation platform. Therefore, we could not draw any direct relations or conclusions from an evaluation of the comparison of case studies and interviews. Instead, we used the interview analysis to evaluate the results of our multi-case analysis.

Our design science research approach involves qualitative research methods for data collection and data analysis. As the results of qualitative research are often criticized as subjective and biased by the investigator (e.g., Beuving & De Vries, 2015; Guba & Lincoln, 1982; Miles et al., 2014; Straub & Gefen, 2004), research rigor with regard to reliability and validity needs to be ensured for qualitative research endeavors (Golafshani, 2003; Miles et al., 2014; Recker, 2013).⁵ To ensure the construct validity, internal validity, external validity, and reliability of our research project, we applied the tactics put forth by Guba (1981) and Yin (2018) for an increased research rigor. We ensured construct validity by establishing a chain of evidence triangulating multiple data sources in the multi-case analysis (Yin, 2018) and our interviewees verified the interview transcript to avoid

misunderstandings and misinterpretations (Shenton, 2004). To increase the internal validity of our research project, we provided transparency on our research procedure, allowed for data triangulation across the applied methods, and depicted linkages among the methods applied. Moreover, we tried to match patterns (i.e., lifecycle phases) derived from existing literature with our case observations (Yin, 2018). Following Hänninen et al. (2018), we used a variety of secondary data sources for our multi-case analysis. We acquired the interviewees based upon a random sampling approach among the population of existing innovation platforms and approached a total of 46 platforms with only nine platforms offering volunteers for interviews. We elaborated on the purpose of the interviews, provided additional information a priori to the interviewees, and offered the possibility for refusing the interview (Guba, 1981; Shenton, 2004). With regard to the number of interviewees, we applied sufficiency and data saturation as indicators (Guest et al., 2006; Hageman & Wutich, 2017; Seidman, 2019). While sufficiency was achieved by contacting a variety of innovation platforms in terms of size, revenue, and provided services, and by reaching interviewees with different positions (Table 2), data saturation was already achieved after the seventh interview with two additional interviews in the pipeline. Data saturation was measured by the degree of new information added and changes to the coding scheme caused by the consecutive interviews (Guest et al., 2006). The degree of additional information was already low when analyzing the seventh interview and high saturation of information on boundary resources in general and boundary resource management stages in particular was achieved, despite the platforms' diverging peculiarities. External validity was ensured by abstracting from the peculiarities of analyzed case companies and interviewee information. Hence, our boundary resource management lifecycle approach might be implemented for a sophisticated boundary management for technical boundary resources and supplementary social boundary resources in other domains of interest. Although we conducted our multi-case analysis with a replication logic (Yin, 2018), the external validity of our study could be further increased by analyzing innovation platforms and consulting interviewees in other domains. Reliability was enriched by providing supplementary information on our research endeavor (e.g., concept matrix, intermediary life-cycle models). By providing transparency on the intermediary life-cycle models, we maintained a consistent chain of evidence throughout our design science research project (Yin, 2018). Among the team of authors, we periodically reflected on the

⁵ Several scholars have proposed guidelines to address the subjective and interpretive nature of qualitative research and increase research rigor (e.g., Beuving & De Vries, 2015; Lincoln & Guba, 1985; Merriam, 1999; Myers & Newman, 2007; Shanks, 2002; G. Walsham, 1995; Geoff Walsham, 2006). These researchers refer to confirmability, dependability, credibility, and transferability as quality criteria (e.g., Miles et al., 2014; Shenton, 2004).

research design, execution, and results and prepared research diaries for the design science research project (Guba, 1981; Yin, 2018).

Research agenda

Drawing on existing calls for research on boundary resources and platform governance (e.g., de Reuver et al., 2018; Hein et al., 2020; Schöler & Petrik, 2021; Y. Yoo et al., 2010) and our research on boundary resource management, we distilled three areas for future research (i.e., innovation ecosystems, boundary resource management, types of boundary resources) and suggested questions guiding future research, namely Table 4.

The first research area is concerned with future research on boundary resources on (innovation) ecosystem level. Despite first attempts to conceptualize the concept of generativity in platform contexts (e.g., Blaschke & Brosius, 2018; Fürstenau et al., 2023; Jarvenpää & Standaert, 2018) and to analyze its effect on third-party developer adoption (Miremadi et al., 2023), the impact of boundary resource provision on ecosystem generativity remains underrepresented (Hein et al., 2020). In this regard, the role of boundary resources in the dichotomy between platform openness and platform control is worthwhile investigating (e.g., Benlian et al., 2015; Spaeth & Niederhöfer, 2022; Y. Yoo et al., 2010). Another possible avenue for future research would be an analysis of the influence of boundary resources on different types of ecosystem participants (Eisenmann et al., 2009). The influence of proper boundary resource implementation

on third-party developers and their dyadic relationship with platform owners would also be worthwhile investigating. In particular, providing transparency about boundary resource development and provisioning might mitigate distributed tuning of boundary resources by other ecosystem participants (Eaton et al., 2015).

The second research area deals with boundary resource management—analyzed in this research paper—in particular. Adding to the boundary resource management lifecycle, future research might derive design guidelines for implementing boundary resources for innovation platform owners and abstract from domain peculiarities of existing research (e.g., Engert et al., 2022; Wulfert et al., 2022). Although we chose e-commerce as the application domain for our research on the management of boundary resources to achieve a high fitness for one domain, the lifecycle might be applicable in other domains (e.g., agriculture). Hence, a possible avenue for future research would be to test the boundary resource management lifecycle in other domains and increase its projectability while retaining a high fitness (Vom Brocke et al., 2020). Hence, future research could discuss the management of boundary resources with additional experts from other domains and third-party developers. Future research could also evaluate the boundary resource management lifecycle's benefits and performance by applying it in selected companies and customizing it to the peculiarities of these cases (Hevner et al., 2004; Vom Brocke, 2015). The performance could be evaluated using appropriate indicators, such as the number of third-party developers, the number of extensions, developer satisfaction, and interface downtime.

Table 4 Future research agenda

Field of action	Exemplary research questions
Innovation ecosystems	What is the impact of boundary resources on platform openness? How can boundary resources be managed to ensure platform control? How do boundary resources influence ecosystem generativity? Which ecosystem participants can engage in an innovation platform's boundary resource management? How can third-party developers tune boundary resources?
Boundary resource management	What are specific design guidelines for implementing boundary resources in innovation platforms? What are the benefits of implementing boundary resource management lifecycle? What principles do exist for boundary resource governance and continuous communication? How is boundary resource management implemented across domains? Which ALM tools support boundary resource management? How can boundary resource management be supported with dedicated software tools?
Types of boundary resources	In which scenarios is backward compatibility for technical boundary resources required? How can a proper mapping between technical boundary resources and supplementary boundary resources be achieved? Which boundary resource archetypes do exist for innovation platforms?

Another possible avenue for future research would be a more detailed analysis of the core phases boundary resource governance and continuous communication. Analogous to ALM, which is typically supported in practice using dedicated tools depending on the lifecycle stage or integrated solutions (Aston, 2022; Jwo et al., 2013), boundary resource management needs to be supported with IS. Future research might investigate existing ALM solutions for the coverage of boundary resources and implement potential extensions for these tools. An important avenue for future research would be the development of a tool supporting the different boundary resource types' entire lifecycle in an integrated manner.

The third research area identified is concerned with different types of boundary resources (e.g., Dal Bianco et al., 2014; Ghazawneh & Henfridsson, 2013). It could be worth investigating for which innovation platforms and technical boundary resources backward compatibility is required and for which new versions discarding old ties are sufficient. A possible explanation for this might be the platform owner's relative power within the surrounding ecosystem (Hein et al., 2016; Hurni et al., 2022; Kretschmer et al., 2022). Moreover, future research could investigate the relationship between technical and supplementary boundary resources in more detail. Existing investigations distinguished these boundary resource types, but did not discuss their relationship (e.g., Dal Bianco et al., 2014; Ghazawneh & Henfridsson, 2013). Moreover, future research could strive for deriving archetypes of boundary resources to enable a more focused management and implementation. We summarize the research areas with potential research questions for future research endeavors in Table 4.

Conclusion

We developed a reference procedure model for the management of the whole lifecycle of boundary resources in innovation ecosystems from the perspective of focal platform owners. The boundary resource management lifecycle is structured in four layers: boundary resource governance,

continuous communication, technical boundary resources, and associated supplementary boundary resources. With this research, we add value to the literature on ALM with a special focus on technical boundary resources and supplementary social boundary resources. The ALM perspective was especially applied for technical boundary resources and extended for this particular type of software artifacts. Supplementary boundary resources are mapped according to these lifecycle stages to support boundary resources' ecosystem-wide application. The lifecycle for the management of technical boundary resources consists of the following stages: plan and design, development and test, release, maintenance and optimization, and deprecation. Major supplementary boundary resources include written documentation, repositories, changelogs, tutorials, status reports, and transition plans. The boundary resource management lifecycle supports owners of innovation platforms in organizing boundary resources and orchestrating third-party developers. Third-party developers can anticipate potential changes in boundary resources made by the platform owner applying our model to boundary resources used. We contribute to the current body of knowledge on ALM by providing a sophisticated reference procedure model for the management of technical and supplementary boundary resources. In the same vein, we enhance existing research with a mapping of supplementary boundary resources to technical boundary resources' lifecycle stages while highlighting their intertwined nature.

Appendix

Appendix. Concept matrix

In the following, we present the concept matrix resulting from our review of the literature on application lifecycle management (Table 5) following the approach by Webster and Watson (2002). We summarized the number of phases and aggravated the denomination of each phase. For each application lifecycle, we analyzed the phases for information on managing boundary resources.

Table 5 Concept matrix of application lifecycle management stages

Reference	Number of Phases	1: Requirements	2: Plan	3: Design	4: Development	5: Test	6: Release	7: Maintenance	8: Optimization	9: Deprecation	10: Communication	11: Governance
Andrikopoulos (2018)	6	•	•	•	•	-	•	•	•	-	-	•
AXELOS (2012)	6	•	•	•	•	•	•	•	•	-	-	-
Bondel et al. (2021)	7	•	•	•	•	•	•	•	-	-	•	-
Chappell (2008)	8	•	•	•	•	•	•	•	•	-	•	•
Chung and Kim (2010)	6	•	•	•	•	•	•	•	•	•	-	-
Cresswell et al. (2013)	4	•	•	•	•	•	•	•	•	-	-	-
Deuter and Rizzo (2016)	8	•	•	•	•	•	•	•	•	-	-	•
Duda et al. (2022)	5	•	•	•	•	•	•	•	•	-	-	-
Ebert (2012)	4	•	•	-	•	•	•	•	•	•	-	•
Hallerstede and Bullinger (2014)	6	•	•	•	•	-	•	•	•	-	-	-
Hallerstede et al. (2011)	6	•	•	•	•	•	•	•	•	-	-	-
Hosono et al. (2011)	8	•	•	•	•	•	•	•	-	-	-	•
Imran et al. (2015)	10	•	•	•	•	•	•	•	-	-	-	•
Kääriäinen (2011)	9	•	•	•	•	•	•	•	-	-	•	•
Kääriäinen and Välimäki (2008)	9	•	-	•	•	•	•	•	-	-	•	•
Kalluri and Gilruth (2004)	9	•	•	•	•	•	•	•	•	•	-	-
Khabou et al. (2019)	5	•	•	•	•	•	•	•	-	-	-	-
Kritikos et al. (2019)	7	•	•	•	•	•	•	•	•	-	-	-
Kritikos and Skrzypek (2018)	7	•	•	•	•	•	•	•	•	-	-	-
Maerien et al. (2015)	4	•	•	•	•	•	•	•	-	•	-	•
Matthes et al. (2009)	9	•	•	•	•	•	•	•	•	-	•	•
Patni (2017)	8	•	•	•	•	•	•	•	•	•	-	-
Priyadarsini et al. (2020)	4	•	•	•	•	•	•	•	•	-	•	-
Rajba (2018)s	4	•	•	•	•	•	•	•	•	•	-	-
Tang et al. (2011)	5	•	•	•	•	•	•	•	•	•	-	-
Tüzün et al. (2019)	3	•	•	•	•	•	•	•	•	-	•	•

Acknowledgements We acknowledge support by the Open Access Publication Fund of the University of Duisburg-Essen.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Adner, R. (2017). Ecosystem as structure: An actionable construct for strategy. *Journal of Management*, 43(1), 39–58. <https://doi.org/10.1177/0149206316678451>
- Adobe. (2022a). *Backward-incompatible changes highlights*. <https://developer.adobe.com/commerce/php/development/backward-incompatible-changes/highlights/>. Accessed 18 Oct 2022.
- Adobe. (2022b). *Code Contributions*. <https://developer.adobe.com/commerce/contributor/guides/code-contributions/>. Accessed 18 Oct 2022.
- Adobe. (2022c). *Definition of done*. <https://developer.adobe.com/commerce/contributor/guides/code-contributions/definition-of-done/#tco-analysis>. Accessed 18 Oct 2022.
- Adobe. (2022d). *Release schedule*. <https://experienceleague.adobe.com/docs/commerce-operations/release/schedule.html?mt=false>. Accessed 18 Oct 2022.
- Adobe. (2022e). *System requirements*. <https://experienceleague.adobe.com/docs/commerce-operations/installation-guide/system-requirements.html?mt=false>. Accessed 18 Oct 2022.
- Adobe. (2022f). *Web API Functional Testing*. <https://developer.adobe.com/commerce/webapi/get-started/web-api-functional-testing/>. Accessed 18 Oct 2022.
- Akana. (2020). *How to Define API Requirements*. <https://www.akana.com/blog/api-requirements-what-consider>. Accessed 26 Oct 2022.
- Alhassan, I., Sammon, D., & Daly, M. (2016). Data governance activities: An analysis of the literature. *Journal of Decision Systems*, 25, 64–75. <https://doi.org/10.1080/12460125.2016.1187397>
- Alsaawi, A. (2014). A critical review of qualitative interviews. *European Journal of Business and Social Sciences*, 3(4), 149–156. <https://doi.org/10.1093/applin/amq043>
- Alt, R. (2020). Electronic markets on business model development. *Electronic Markets*, 30(3), 405–411. <https://doi.org/10.1007/s12525-020-00438-z>
- Alves, C., Oliveira, J., & Jansen, S. (2017). Software ecosystems governance-A systematic literature review and research agenda. *Proceedings of the 19th International Conference on Enterprise Information Systems (ICEIS 2017)* (pp. 26–29). Porto, Portugal.
- Amazon. (2022). *Amazon Annual Report 2021*. Annual Report. https://s2.q4cdn.com/299287126/files/doc_financials/2022/ar/Amazon-2021-Annual-Report.pdf. Accessed 18 Jan 2023.
- Andrikopoulos, V. (2018). Engineering cloud-based applications: Towards an application lifecycle. In Z. Á. Mann & V. Stolz (Eds.), *Advances in Service-Oriented and Cloud Computing* (Vol. 824, pp. 57–72). Springer International Publishing. <https://doi.org/10.1007/978-3-319-79090-9>
- Armstrong, M. (2006). Competition in two-sided markets. *RAND Journal of Economics*, 37(3), 668–691. <https://doi.org/10.1111/j.1756-2171.2006.tb00037.x>
- Aston, B. (2022). *10 Best ALM Tools For 2022*. Best tools. <https://thedigitalprojectmanager.com/tools/alm-tools/>. Accessed 25 Oct 2022.
- Aulkemeier, F., Paramartha, M. A., Iacob, M. E., & van Hillegersberg, J. (2016a). A pluggable service platform architecture for e-commerce. *Information Systems and E-Business Management*, 14(3), 469–489. <https://doi.org/10.1007/s10257-015-0291-6>
- Aulkemeier, F., Schramm, M., Iacob, M. E., & van Hillegersberg, J. (2016b). A service-oriented e-commerce reference architecture. *Journal of Theoretical and Applied Electronic Commerce Research*, 11(1), 26–45. <https://doi.org/10.4067/S0718-18762016000100003>
- Autio, E. (2021). Orchestrating ecosystems: a multi-layered framework. *Innovation: Organization and Management*, 00(00), 1–14. <https://doi.org/10.1080/14479338.2021.1919120>
- Aversa, R., Panza, N., & Tasquier, L. (2015). An agent-based platform for cloud applications performance monitoring. *2015 Ninth International Conference on Complex, Intelligent, and Software Intensive Systems*, 535–540. <https://doi.org/10.1109/CISIS.2015.79>
- AXELOS. (2012). *ITIL Service Operation*. AXELOS Publishing.
- Babb, J. S., Nørberg, J., Yates, D. J., & Waguespack, L. J. (2017). The empire strikes back: The end of agile as we know it? *Proceedings of the 40th Information Systems Research Seminar in Scandinavia* (pp. 44–59). Halden, Norway.
- Baldwin, C. Y., & Woodard, J. C. (2009). The architecture of platforms: a unified view. In A. Gawer (Ed.), *Platforms, Markets and Innovation* (pp. 19–44). Edward Elgar. <https://doi.org/10.4337/9781849803311>
- Bandara, W., Furtmueller, E., Gorbacheva, E., Miskon, S., & Beekhuizen, J. (2015). Achieving rigor in literature reviews: Insights from qualitative data analysis and tool-support. *Communications of the Association for Information Systems*, 37, 154–204. <https://doi.org/10.17705/1cais.03708>
- Baskerville, R., Kaul, M., Pries-Heje, J., & Storey, V. (2019). Inducing creativity in design science research. *Proceedings of the 14th International Conference on Design Science Research in Information Systems and Technology (DESIST2019)*, 3–17. https://doi.org/10.1007/978-3-030-19504-5_1
- Becker, J., & Delfmann, P. (2007). *Reference Modeling. Efficient Information Systems Design Through Reuse of Information Models*. Physica-Verlag.
- Benfenatki, H., Ferreira Da Silva, C., Kemp, G., Benharkat, A.-N., Ghodous, P., & Maamar, Z. (2017). MADONA: A method for automated provisioning of cloud-based component-oriented business applications. *Service Oriented Computing and Applications*, 11(1), 87–100. <https://doi.org/10.1007/s11761-016-0199-0>
- Benlian, A., Hilkert, D., & Hess, T. (2015). How open is this platform? The meaning and measurement of platform openness from the complementors' perspective. *Journal of Information Technology*, 30(3), 209–228. <https://doi.org/10.1057/jit.2015.6>
- Beuving, J., & De Vries, G. (2015). *Doing Qualitative Research: The Craft of Naturalistic Inquiry*. Amsterdam University Press.
- Beverungen, D., Müller, O., Matzner, M., Mendling, J., & vom Brocke, J. (2019). Conceptualizing smart service systems. *Electronic Markets*, 29(1), 7–18. <https://doi.org/10.1007/s12525-017-0270-5>
- BigCommerce. (2022a). *About Our APIs*. <https://developer.bigcommerce.com/docs/3eea0e31b4c6a-about-our-ap-is>. Accessed 18 Oct 2022.

- BigCommerce. (2022b). *API Best Practices*. <https://developer.bigcommerce.com/docs/a0dd0f08933ce4-api-best-practices>. Accessed 18 Oct 2022.
- BigCommerce. (2022c). *BigCommerce Examples & Starters*. <https://developer.bigcommerce.com/dev-library/>. Accessed 18 Oct 2022.
- BigCommerce. (2022d). *BigCommerce Status*. <https://status.bigcommerce.com/>. Accessed 18 Oct 2022.
- BigCommerce. (2022e). *Create a Sandbox Store*. <https://developer.bigcommerce.com/docs/a8d9c927e48b5-create-a-sandbox-store>. Accessed 18 Oct 2022.
- BigCommerce. (2022f). *Feature Request*. https://support.bigcommerce.com/s/topic/0TO130000001RKPGA2/feature-request?language=en_US. Accessed 18 Oct 2022.
- BigCommerce. (2022g). *Knowledge Base*. https://support.bigcommerce.com/s/knowledge?language=en_US. Accessed 18 Oct 2022.
- Blaschke, M. R., & Brosius, M. (2018). Digital platforms: Balancing control and generativity. *Proceedings of the 39th International Conference on Information Systems (ICIS2018)* (pp. 1–9). San Francisco, USA.
- Blaschke, M. R., Haki, K., Aier, S., & Winter, R. (2018). Capabilities of digital platform survival: Insights from a business-to-business digital platform. *Proceedings of the 39th International Conference on Information Systems (ICIS2018)* (pp. 1–17). San Francisco, USA.
- Blaschke, M. R., Haki, K., Aier, S., & Winter, R. (2019a). Taxonomy of digital platforms: A platform architecture perspective. *Proceedings of the 14th International Conference on Wirtschaftsinformatik*. Siegen, Germany.
- Blaschke, M. R., Riss, U., Haki, K., & Aier, S. (2019b). Design principles for digital value co-creation networks: A service-dominant logic perspective. *Electronic Markets*, 29(3), 443–472. <https://doi.org/10.1007/s12525-019-00356-9>
- Bondel, G., Landgraf, A., & Matthes, F. (2021). API management patterns for public, partner, and group web API initiatives with a focus on collaboration. *26th European Conference on Pattern Languages of Programs*, 1–17. <https://doi.org/10.1145/3489449.3490012>
- Böttcher, T., Rickling, L., Gmelch, K., Weking, J., & Krcmar, H. (2021). Towards the digital self-renewal of retail: The generic ecosystem of the retail industry. *Proceedings of the 16th Internationale Tagung Wirtschaftsinformatik (WI2021)* (pp. 1–9). Essen, Germany.
- Boudreau, K. (2010). Open platform strategies and innovation: Granting access vs. devolving control. *Management Science*, 56(10), 1849–1872. <https://doi.org/10.1287/mnsc.1100.1215>
- Boudreau, K., & Hagiu, A. (2009). Platform rules: Multi-sided platforms as regulators. In A. Gawer (Ed.), *Platforms, Markets and Innovation* (pp. 163–191). Edward Elgar. <https://doi.org/10.4337/9781849803311>
- Builtwith. (2022). *eCommerce Usage Distribution in the Top 1 Million Sites*. <https://trends.builtwith.com/shop>. Accessed 22 Oct 2022.
- Carlile, P. R. (2002). A pragmatic view of knowledge and boundaries: Boundary objects in new product development. *Organization Science*, 13(4), 442–455. <https://doi.org/10.1287/orsc.13.4.442.2953>
- Chanda, S., & Foggon, D. (2013). Application lifecycle management. In *Beginning ASP.NET 4.5 Databases* (pp. 235–249). Apress. https://doi.org/10.1007/978-1-4302-4381-6_13
- Chandra Kruse, L., & Seidel, S. (2017). Tensions in design principle formulation and reuse. In A. Maedche, J. Vom Brocke, & A. R. Hevner (Eds.), *Proceedings of the 12th International Conference on Design Science Research in Information Systems and Technology (DESRIST 2017)* (pp. 180–188). Karlsruhe, Germany.
- Chappell, D. (2008). *What is Application Lifecycle Management?*. <https://download.microsoft.com/download/9/5/F/95FB4775-F769-477E-B187-6737DC09BC9D/What.is.ALM%2D%2DChappell.pdf>. Accessed 25 Oct 2022
- Chen, D., & Zhao, H. (2012). Data security and privacy protection issues in cloud computing. *2012 International Conference on Computer Science and Electronics Engineering*, 647–651. <https://doi.org/10.1109/ICCSEE.2012.193>
- Choi, S., Whinston, A. B., & Stahl, D. (1997). *The Economics of Electronic Commerce*. Macmillan Computer Publishing.
- Chung, K.-S., & Kim, S. (2010). A study on the application lifecycle management over SOA based application hosting platform. *Proceedings of the 12th International Conference on Advanced Communication Technology (ICACT2010)* (pp. 930–933). Phoenix Park, South Korea.
- Clemmensen, T., Norbjerg, J., & Joshi, A. (2008). Information systems – Why it is a design science. *Proceedings, Computers and Philosophy an International Conference* (pp. 126–139). Laval, France.
- Cram, W. A., Templier, M., & Paré, G. (2020). (Re)considering the concept of literature review reproducibility. *Journal of the Association for Information Systems*, 21(5), 1103–1114. <https://doi.org/10.17705/1jais.00630>
- Cresswell, K. M., Bates, D. W., & Sheikh, A. (2013). Ten key considerations for the successful implementation and adoption of large-scale health information technology. *Journal of the American Medical Informatics Association : JAMIA*, 20(e1), e9–e13. <https://doi.org/10.1136/amiajnl-2013-001684>
- Dal Bianco, V., Myllärniemi, V., Komssi, M., & Raatikainen, M. (2014). The role of platform boundary resources in software ecosystems: A case study. *Proceedings of the 11th Working IEEE/IFIP Conference on Software Architecture (WICSA2014)* (pp. 11–20). Sydney, Australia.
- Datanyze. (2021). *E-Commerce Platforms Market Share*. <https://www.datanyze.com/market-share/e-commerce-platforms%2D%2D6>. Accessed 22 Oct 2022.
- de Reuver, M., Sørensen, C., & Basole, R. C. (2018). The digital platform: A research agenda. *Journal of Information Technology*, 33(2), 124–135. <https://doi.org/10.1057/s41265-016-0033-3>
- Delfmann, P. (2006). *Adaptive Referenzmodellierung. Methodische Konzepte zur Konstruktion und Anwendung wiederverwendungsorientierter Informationsmodelle*. Logos Verlag.
- Demchenko, Y., Turkmen, F., Slawik, M., & Laat, C. D. (2017). Defining intercloud security framework and architecture components for multi-cloud data intensive applications. *17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID2017)* (pp. 945–952). <https://doi.org/10.1109/CCGRID.2017.144>
- Deuter, A., & Rizzo, S. (2016). A critical view on PLM/ALM convergence in practice and research. *Procedia Technology*, 26, 405–412. <https://doi.org/10.1016/j.protcy.2016.08.052>
- Diederich, S., Brendel, A. B., & Kolbe, L. M. (2020). Designing anthropomorphic enterprise conversational agents. *Business & Information Systems Engineering*, 62(3), 193–209. <https://doi.org/10.1007/s12599-020-00639-y>
- Dittrich, Y., Nørbjerg, J., Tell, P., & Bendix, L. (2018). Researching cooperation and communication in continuous software engineering. *Proceedings of the 11th International Workshop on Cooperative and Human Aspects of Software Engineering*, 87–90. <https://doi.org/10.1145/3195836.3195856>
- Drechsler, A., & Hevner, A. R. (2018). Utilizing, producing, and contributing design knowledge in DSR projects. In S. Chatterjee, K. Dutta, & R. P. Sundarraj (Eds.), *Proceedings of the 13th International Conference on Design Science Research in Information Systems and Technology* (pp. 82–97). https://doi.org/10.1007/978-3-319-91800-6_6
- Duda, J., Oleszek, S., & Santarek, K. (2022). Product lifecycle management (PLM) in the context of industry 4.0. In J. Trojanowska,

- A. Kujawińska, J. Machado, & I. Pavlenko (Eds.), *Advances in Manufacturing III* (pp. 171–185). Springer International Publishing. <https://doi.org/10.1007/978-3-030-99310-8>
- Eaton, B., Elaluf-Calderwood, S., Sorensen, C., & Yoo, Y. (2015). Distributed tuning of boundary resources: The case of Apple's iOS service system. *MIS Quarterly: Management Information Systems*, 39(1), 217–243.
- Ebert, C. (2012). *Global Software and IT. A Guide to Distributed Development, Projects, and Outsourcing*. John Wiley & Sons.
- Eisenhardt, K. M. (1989). Building theories from case study research. *The Academy of Management Review*, 14(4), 532–550. <https://doi.org/10.5465/amr.1989.4308385>
- Eisenmann, T., Parker, G., & Van Alstyne, M. (2009). Opening platforms: How, when and why? In A. Gawer (Ed.), *Platforms, Markets and Innovation* (pp. 131–162). Edward Elgar. <https://doi.org/10.4337/9781849803311>
- Engert, M., Evers, J., Hein, A., & Krcmar, H. (2022). The engagement of complementors and the role of platform boundary resources in e-commerce platform ecosystems. *Information Systems Frontiers*, 0123456789. <https://doi.org/10.1007/s10796-021-10236-3>
- Estublier, J. (2000). Software configuration management: A roadmap. In C. Ghezzi, M. Jazayeri, & A. L. Wolf (Eds.), *Proceedings of the 22nd International Conference on Software Engineering (ICSE2000)* (pp. 279–289). Limerick, Ireland
- Fettke, P., & Loos, P. (2002). Der Referenzmodellkatalog als Instrument des Wissensmanagements: Methodik und Anwendung. In J. Becker & R. Knackstedt (Eds.), *Wissensmanagement mit Referenzmodellen. Konzepte für die Anwendungssystem- und Organisationsgestaltung* (pp. 3–24). Springer.
- Fischbach, M., Puschmann, T., & Alt, R. (2013). Catchword: Service lifecycle management. *Business & Information Systems Engineering*, 5(1), 45–49. <https://doi.org/10.1007/s12599-012-0241-5>
- Fürstenau, D., Baiyere, A., Schewina, K., Schulte-althoff, M., Rothe, H., Daniel, F., Baiyere, A., Schewina, K., & Schulte-althoff, M. (2023). Extended generativity theory on digital platforms. *Information System Research*, March. <https://doi.org/10.1287/isre.2023.1209>
- Gawer, A. (2021). Digital platforms and ecosystems: Remarks on the dominant organizational forms of the digital age. *Innovation: Organization & Management*, 00(00), 1–15. <https://doi.org/10.1080/14479338.2021.1965888>
- Ghazawneh, A. (2012). Towards a boundary resources theory of software platforms. In *JIBS disseration series* (Issue 85). ARK Tryckaren AB. Jönköping, Sweden. <http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A567769&dsid=1262>
- Ghazawneh, A., & Henfridsson, O. (2010). Governing third-party development through platform boundary resources. *Proceedings of the 31st International Conference on Information Systems (ICIS2010)* (pp. 1–17). Saint Louis, Missouri, USA.
- Ghazawneh, A., & Henfridsson, O. (2013). Balancing platform control and external contribution in third-party development: The boundary resources model. *Information Systems Journal*, 23(2), 173–192. <https://doi.org/10.1111/j.1365-2575.2012.00406.x>
- Github. (2022a). *BigCommerce*. <https://github.com/bigcommerce>. Accessed 18 Oct 2022.
- Github. (2022b). *OXID eSales AG*. <https://github.com/OXID-eSales>. Accessed 21 Oct 2022.
- Github. (2022c). *shopware/platform*. <https://github.com/shopware/platform>. Accessed 21 Oct 2022.
- Github. (2023). *Selling Partner API Documentation*. <https://github.com/amzn/selling-partner-api-docs?Id=NSGoogle>. Accessed 18 Jan 2023.
- Golafshani, N. (2003). Understanding reliability and validity in qualitative research. *The Qualitative Report*, 8(4), 597–606. <https://doi.org/10.46743/2160-3715/2003.1870>
- Gregor, S. (2006). The nature of theory in information systems. *MIS Quarterly: Management Information Systems*, 30(3), 611–642. <https://doi.org/10.2307/25148742>
- Gregor, S., & Jones, D. (2007). The anatomy of a design theory. *Journal of the Association for Information Systems*, 8(5), 312–335. <https://doi.org/10.17705/1jais.00129>
- Gregor, S., Müller, O., & Seidel, S. (2013). Reflection, abstraction, and theorizing in design and development research. *Proceedings of the 21st European Conference on Information Systems (ECIS2013)* (pp. 1–12). Utrecht, The Netherlands.
- Guba, E. G. (1981). Criteria for assessing the trustworthiness of naturalistic inquiries. *Educational Communication and Technology*, 29(2), 75–91.
- Guba, E. G., & Lincoln, Y. S. (1982). Epistemological and methodological bases of naturalistic inquiry. *ECTJ*, 30(4), 233–252. <https://doi.org/10.1007/BF02765185>
- Guest, G., Bunce, A., & Johnson, L. (2006). How many interviews are enough?: An experiment with data saturation and variability. *Field Methods*, 18(1), 59–82. <https://doi.org/10.1177/1525822X05279903>
- Hagaman, A. K., & Wutich, A. (2017). How many interviews are enough to identify metathemes in multisited and cross-cultural research? Another perspective on Guest, Bunce, and Johnson's (2006) landmark study. *Field Methods*, 29(1), 23–41. <https://doi.org/10.1177/1525822X16640447>
- Hallerstede, S. H., & Bullinger, A. C. (2014). Managing the lifecycle of online innovation contests - A case study on an innovation intermediary's approach. *International Journal of Technology Marketing*, 9(2), 125. <https://doi.org/10.1504/IJTMKT.2014.060077>
- Hallerstede, S. H., Danzinger, F., Bullinger, A. C., & Möslin, K. M. (2011). Akzeptanzorientiertes application lifecycle management. *HMD Praxis Der Wirtschaftsinformatik*, 48(2), 30–40. <https://doi.org/10.1007/BF03340565>
- Hänninen, M., Smedlund, A., & Mitronen, L. (2018). Digitalization in retailing: multi-sided platforms as drivers of industry transformation. *Baltic Journal of Management*, 13(2), 152–168. <https://doi.org/10.1108/BJM-04-2017-0109>
- Hein, A., Schrieck, M., Wiesche, M., & Krcmar, H. (2016). Multiple-case analysis on governance mechanisms of multi-sided platforms. In V. Nissen, D. Stelzer, S. Straßburger, & D. Fischer (Eds.), *Multikonferenz Wirtschaftsinformatik (MKWI)* (pp. 1613–1624). Ilmenau, Germany.
- Hein, A., Weking, J., Schrieck, M., Wiesche, M., Böhm, M., & Krcmar, H. (2019). Value co-creation practices in business-to-business platform ecosystems. *Electronic Markets*, 29(3), 503–518. <https://doi.org/10.1007/s12525-019-00337-y>
- Hein, A., Schrieck, M., Riasanow, T., Setzke, D. S., Wiesche, M., Böhm, M., & Krcmar, H. (2020). Digital platform ecosystems. *Electronic Markets*, 30(1), 87–98. <https://doi.org/10.1007/s12525-019-00377-4>
- Hervner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105. <https://doi.org/10.2307/25148625>
- Hervner, A. R. (2007). A three cycle view of design science research. *Scandinavian Journal of Information Systems*, 19(2), 87–92. <https://aisel.aisnet.org/sjis/vol19/iss2/4>
- Hinz, O., Otter, T., & Skiera, B. (2020). Estimating network effects in two-sided markets. *Journal of Management Information Systems*, 37(1), 12–38. <https://doi.org/10.1080/07421222.2019.1705509>
- Hosono, S., He, J., Liu, X., Li, L., Huang, H., & Yoshino, S. (2011). Fast development platforms and methods for cloud applications. *IEEE Asia-Pacific Services Computing Conference, 2011*, 94–101. <https://doi.org/10.1109/APSCC.2011.75>

- Hurni, T., Huber, T. L., & Dibbern, J. (2022). Power dynamics in software platform ecosystems. *Information Systems Journal*, 32(2), 310–343. <https://doi.org/10.1111/isj.12356>
- Iansiti, M., & Levien, R. (2004). *The Keystone Advantage - What the New Dynamics of Business Ecosystems Mean for Strategy, Innovation, and Sustainability*. Harvard Business School Publishing Corporation. Boston, Mass.
- Iivari, J. (2015). Distinguishing and contrasting two strategies for design science research. *European Journal of Information Systems*, 24(1), 107–115. <https://doi.org/10.1057/ejis.2013.35>
- Iivari, J., Hansen, R. P., & M., & Haj-Bolouri, A. (2021). A proposal for minimum reusability evaluation of design principles. *European Journal of Information Systems*, 30(3), 1–18. <https://doi.org/10.1080/0960085X.2020.1793697>
- Imran, S., Buchheit, M., Hollunder, B., & Schreier, U. (2015). Tool chains in agile ALM environments: A short introduction. In I. Ciuciu, H. Panetto, C. Debruyne, A. Aubry, P. Bollen, R. Valencia-García, A. Mishra, A. Fensel, & F. Ferri (Eds.), *On the Move to Meaningful Internet Systems: OTM 2015 Workshops* (Vol. 9416, pp. 371–380). Springer International Publishing. <https://doi.org/10.1007/978-3-319-26138-6>
- ISO. (2017). *ISO 12207:2017 - Software Lifecycle Processes*. International Organization for Standardization (ISO). <https://www.iso.org/standard/63712.html>. Accessed 16 Oct 2022.
- Jansen, S., Brinkkemper, S., & Cusumano, M. A. (2013). *Software Ecosystems. Analyzing and Managing Networks in the Software Industry*. Edward Elgar.
- Jansen, S., & van Capelleveen, G. (2013). Quality review and approval methods for extensions in software ecosystems. In S. Jansen, S. Brinkkemper, & M. A. Cusumano (Eds.), *Software Ecosystems: Analyzing and Managing Business Networks in the Software Industry* (pp. 187–217). Elgar.
- Jarvenpaa, S. L., & Standaert, W. (2018). Digital probes as opening possibilities of generativity. *Journal of the Association for Information Systems*, 19(10), 982–1000. <https://doi.org/10.17705/1jais.00516>
- Jwo, J.-S., Hsu, T.-S., & Cheng, Y. C. (2013). Jumpstarting application lifecycle management: A new approach with tool support. *Journal of Information Science and Engineering*, 29, 475–492.
- Kääriäinen, J. (2011). *Towards an Application Lifecycle Management Framework*. VTT Publications.
- Kääriäinen, J., & Välimäki, A. (2008). Impact of application lifecycle management — A case study. In K. Mertins, R. Ruggaber, K. Popplewell, & X. Xu (Eds.), *Enterprise Interoperability III* (pp. 55–67). Springer.
- Kääriäinen, J., & Välimäki, A. (2009). Applying application lifecycle management for the development of complex systems: Experiences from the automation industry. In R. V. O'Connor, N. Baddoo, J. Cuadrado Gallego, R. Rejas Muslera, K. Smolander, & R. Messnarz (Eds.), *Software Process Improvement* (pp. 149–160). Springer.
- Kabac, M., Consel, C., & Volansch, N. (2016). Leveraging declarations over the lifecycle of large-scale sensor applications. *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, 211–219. <https://doi.org/10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0051>
- Kalluri, S., & Gilruth, P. (2004). Full lifecycle development of transitioning remote sensing research to operational applications. *IEEE International IEEE International IEEE International Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings., 2004*, 4822–4824. <https://doi.org/10.1109/IGARSS.2004.1370241>
- Karhu, K., Gustafsson, R., & Lyytinen, K. (2018). Exploiting and defending open digital platforms with boundary resources: Android's five platform forks. *Information Systems Research*, 29(2), 479–497. <https://doi.org/10.1287/isre.2018.0786>
- Khabou, N., Bouassida Rodriguez, I., & Jmaiel, M. (2019). An overview of a novel analysis approach for enhancing context awareness in smart environments. *Information and Software Technology*, 111, 131–143. <https://doi.org/10.1016/j.infsof.2018.11.001>
- Khatry, V., & Brown, C. V. (2010). Designing data governance. *Communications of the ACM*, 53(1), 148–152. <https://doi.org/10.1145/1629175.1629210>
- Kosiol, E. (1964). Betriebswirtschaftslehre und Unternehmensforschung. Eine Untersuchung ihrer Standorte und Beziehungen auf wissenschaftstheoretischer Grundlagen. *Zeitschrift Für Betriebswirtschaft*, 34(12), 743–762.
- Kretschmer, T., Leiponen, A., Schilling, M., & Vasudeva, G. (2022). Platform ecosystems as meta-organizations: Implications for platform strategies. *Strategic Management Journal*, 43(3), 405–424. <https://doi.org/10.1002/smj.3250>
- Krippendorff, K. (2013). *Content Analysis: An introduction to its methodology* (3rd ed.). SAGE.
- Kritikos, K., Magoutis, K., Papoutsakis, M., & Ioannidis, S. (2019). A survey on vulnerability assessment tools and databases for cloud-based web applications. *Array*, 3–4, 100011. <https://doi.org/10.1016/j.array.2019.100011>
- Kritikos, K., & Skrzypek, P. (2018). A review of serverless frameworks. *2018 IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC Companion)*, 161–168. <https://doi.org/10.1109/UCC-Companion.2018.00051>
- Kuckartz, U., & Rädiker, S. (2022). *Qualitative Inhaltsanalyse. Methoden, Praxis, Computerunterstützung* (5., überar.).
- Lacheiner, H., & Ramler, R. (2011). Application lifecycle management as infrastructure for software process improvement and evolution: Experience and insights from industry. *2011 37th EURO-MICRO Conference on Software Engineering and Advanced Applications*, 286–293. <https://doi.org/10.1109/SEAA.2011.51>
- Laudon, K. C., & Traver, C. G. (2020). *E-Commerce 2020-2021: business. technology. society. (16th ed., Vol. 16)*. Pearson Education.
- Lee, J. S., Pries-Heje, J., & Baskerville, R. (2011). Theorizing in design science research. In H. Jain, A. P. Sinha, & P. Vitharana (Eds.), *Proceedings of the International Conference on Design Science Research in Information Systems (DESIRIST2011)* (pp. 1–16). Springer Berlin Heidelberg.
- Lincoln, Y. S., & Guba, E. G. (1985). *Naturalistic Inquiry*. Sage Publications.
- Maerien, J., Michiels, S., Hughes, D., Huygens, C., & Joosen, W. (2015). SecLooCI: A comprehensive security middleware architecture for shared wireless sensor networks. *Ad Hoc Networks*, 25, 141–169. <https://doi.org/10.1016/j.adhoc.2014.09.018>
- Matthes, F., Neubert, C., & Steinhoff, A. (2009). Federated application lifecycle management based on an open web architecture. *Workshop Design for Future - Langlebige Softwaresysteme, GI-Arbeitskreis Langlebige Softwaresysteme (L2S2)* (pp. 1–12). Karlsruhe, Germany.
- Mayring, P. (2014). *Qualitative Content Analysis: Theoretical Foundation, Basic Procedures and Software Solution*. Beltz. <https://doi.org/10.4135/9781446282243.n12>
- McIntyre, D. P., & Srinivasan, A. (2017). Networks, platforms, and strategy: Emerging views and next steps. *Strategic Management Journal*, 38(1), 141–160. <https://doi.org/10.1002/smj.2596>
- Medjaoui, M., Wilde, E., Mitra, R., & Amundsen, M. (2021). *Continuous API Management: Making the Right Decisions in an Evolving Landscape* (2nd ed.). O'Reilly.
- Merrick, E. (1999). An exploration of quality in qualitative research: Are “reliability” and “validity” relevant? In *Using Qualitative*

- Methods in Psychology* (pp. 25–36). SAGE Publications, Inc.. <https://doi.org/10.4135/978145225487.n3>
- Merton, R. K., & Kendall, P. L. (1946). The focused interview. *American Journal of Soc*, 51(6), 541–557.
- Miles, M. B., Huberman, A. M., & Saldana, J. (2014). *Qualitative Data Analysis: A Methods Sourcebook* (3rd ed.). Sage <http://journal.um-surabaya.ac.id/index.php/JKM/article/view/2203>
- Miremadi, I., Khoshbakhsh, M., & Saeedian, M. M. (2023). Fostering generativity in platform ecosystems: How open innovation and complexity interact to influence platform adoption. *Research Policy*, 52(6), 104781. <https://doi.org/10.1016/j.respol.2023.104781>
- Misoch, S. (2015). *Qualitative Interviews*. De Gruyter Oldenbourg.
- Mohagheghzadeh, A., & Lindman, J. (2022). Shaping boundary resources of a digital platform: Alliance co- development perspective. *PACIS Proceedings*, 1–13. <https://aisel.aisnet.org/pacis2022/120%0AThis>
- Myers, M. D., & Newman, M. (2007). The qualitative interview in IS research: Examining the craft. *Information and Organization*, 17(1), 2–26. <https://doi.org/10.1016/j.infoandorg.2006.11.001>
- Nath, R., Akmanligil, M., Hjelm, K., Sakaguchi, T., & Schultz, M. (1998). Electronic commerce and the internet: issues, problems, and perspectives. *International Journal of Information Management*, 18(2), 91–101. [https://doi.org/10.1016/S0268-4012\(97\)00051-0](https://doi.org/10.1016/S0268-4012(97)00051-0)
- Nerur, S., & Balijepally, V. (2007). Theoretical reflections on agile development methodologies. *Communications of the ACM*, 50(3), 79–83. <https://doi.org/10.1145/1226736.1226739>
- Niederman, F., & March, S. (2019). The “theoretical lens” concept: We all know what it means, but do we all know the same thing? *Communications of the Association for Information Systems*, 44(1), 1–33. <https://doi.org/10.17705/1CAIS.04401>
- Norbjerg, J. (2018). Design. Building the solution or framing a problem? Reflections on design science. *Scandinavian Journal of Information Systems*, 30(2), 69–74.
- Nunkoosing, K. (2005). The problems with interviews. *Qualitative Health Research*, 15(5), 698–706. <https://doi.org/10.1177/1049732304273903>
- OXID. (2021). *Roadmap*. <https://www.oxid-esales.com/blog/roadmap-oxid-eshop-2021/>. Accessed 11 July 2021.
- OXID. (2022a). *Development*. <https://docs.oxid-esales.com/developer/en/latest/development/index.html>. Accessed 21 Oct 2022.
- OXID. (2022b). *Know-how from the OXID Academy*. <https://www.oxid-esales.com/en/resources/academy/>. Accessed 21 Oct 2022.
- OXID. (2022c). *OXID eshop bugtrack*. https://bugs.oxid-esales.com/view_all_bug_page.php. Accessed 21 Oct 2022
- OXID. (2022d). *Release Plan*. <https://docs.oxid-esales.com/en/releases/release-plan.html>. Accessed 21 Oct 2022.
- Parker, G., Van Alstyne, M., & Choundary, S. (2016). *Platform revolution: how networked markets are transforming the economy and how to make them work for you*. W. W. Norton & Company.
- Patni, S. (2017). API management and API client. In *Pro RESTful APIs* (pp. 97–106). Apress. https://doi.org/10.1007/978-1-4842-2665-0_7
- Petri, D., & Herzwurm, G. (2020). Boundary resources for IIoT platforms - A complementor satisfaction study. *Proceedings of the 41st International Conference on Information Systems (ICIS 2020)* (pp. 1–17) Hyderabad, India.
- Pirklbauer, G., Ramler, R., & Zeilinger, R. (2009). An integration-oriented model for application lifecycle management. *Proceedings of the 11th International Conference on Enterprise Information Systems (ICEIS2009)*, 399–402. <https://doi.org/10.5220/0002007203990402>
- Pon, B., Seppälä, T., & Kenney, M. (2014). Android and the demise of operating system-based power: Firm strategy and platform control in the post-PC world. *Telecommunications Policy*, 38(11), 979–991. <https://doi.org/10.1016/j.telpol.2014.05.001>
- Preston-Werner, T. (2022). *Semantic Versioning 2.0.0*. <https://semver.org/#semantic-versioning-200>. Accessed 18 Oct 2022.
- Priyadarsini, K., Raj, E. F. I., Begum, A. Y., & Shanmugasundaram, V. (2020). Comparing DevOps procedures from the context of a systems engineer. *Materials Today: Proceedings*. <https://doi.org/10.1016/j.matpr.2020.09.624>
- Rajba, P. (2018). Challenges and mitigation approaches for getting secured applications in an enterprise company. *Proceedings of the 13th International Conference on Availability, Reliability and Security*. <https://doi.org/10.1145/3230833.3233276>
- Recker, J. (2013). *Scientific Research in Information Systems: A Beginner's Guide*. Springer. <https://doi.org/10.1007/978-3-642-30048-6>
- Rosemann, M., & Vessey, I. (2008). Toward improving the relevance of information systems research to practice: The role of applicability checks. *MIS Quarterly*, 32(1), 7–22. <https://doi.org/10.2307/25148826>
- Rossberg, J. (2014). *Beginning Application Lifecycle Management*. Apress.
- Salesforce. (2022a). *Build apps and automate with low-code magic*. <https://www.salesforce.com/products/platform/overview/?bc=OTH>. Accessed 21 Oct 2022.
- Salesforce. (2022b). *Introduction to Marketing Cloud Platform SDKs*. <https://developer.salesforce.com/docs/marketing/marketing-cloud/guide/sdks-overview.html>. Accessed 21 Oct 2022.
- Salesforce. (2022c). *Metadata Coverage*. <https://developer.salesforce.com/docs/metadata-coverage/56>. Accessed 21 Oct 2022.
- Salesforce. (2022d). *Salesforce Roadmap Explorer*. <https://architect.salesforce.com/roadmaps/roadmap-explorer>. Accessed 21 Oct 2022.
- Salesforce. (2022e). *Salesforce Winter '23 Release Notes*. https://help.salesforce.com/s/articleView?id=release-notes.salesforce_release_notes.htm&type=5&release=240. Accessed 21 Oct 2022.
- Santos, F. M., & Eisenhardt, K. M. (2005). Organizational boundaries and theories of organization. *Organization Science*, 16(5), 491–508. <https://doi.org/10.1287/orsc.1050.0152>
- SAP. (2022a). *openSAP Microlearning: SAP Commerce Cloud*. <https://microlearning.opensap.com/category/SAP+Commerce+Cloud/178315851>. Accessed 21 Oct 2022.
- SAP. (2022b). *Platform, Services, and Utilities: Logging*. https://help.sap.com/docs/SAP_COMMERCE/d0224eca81e249cb821f2cdf45a82ace/8c07853c866910148a00baf81e1669e.html?locale=en-US. Accessed 21 Oct 2022.
- SAP. (2022c). *Platform, Services, and Utilities: Testing*. https://help.sap.com/docs/SAP_COMMERCE/d0224eca81e249cb821f2cdf45a82ace/8c6f059d866910149b6aede7ab7a0f36.html?locale=en-US. Accessed 21 Oct 2022.
- SAP. (2022d). *SAP API Business Hub*. <https://api.sap.com/search?searchterm=commerce.cloud&tab=APIs&top=12>. Accessed 21 Oct 2022.
- SAP. (2022e). *SAP Roadmap Explorer*. <https://www.sap.com/germany/products/roadmaps.html?gl=https%3A%2F%2Froadmaps.sap.com%2Findex.html%23%2F>. Accessed 21 Oct 2022.
- Schilling, M. A. (2000). Toward a general modular systems theory and its application to interfirm product modularity. *Academy of Management Review*, 25(2), 312–334. <https://doi.org/10.5465/AMR.2000.3312918>
- Schreier, M. (2014). Qualitative content analysis. In U. Flick (Ed.), *The Sage handbook of qualitative data analysis* (pp. 170–183). Sage Publications.
- Schuh, G., & Rudolf, S. (2015). *Marktspiegel Business Software PLM/ PDM 2015/2016* (3rd ed.). Trovarit AG.

- Schüler, F., & Petrik, D. (2021). Objectives of platform research: A co-citation and systematic literature review analysis. In M. Seiter, L. Grünert, & A. J. Steur (Eds.), *Management Digitaler Plattformen* (pp. 1–33). Springer Fachmedien. https://doi.org/10.1007/978-3-658-31118-6_1
- Schütte, R. (1998). *Grundsätze ordnungsmäßiger Referenzmodellierung. Konstruktion konfigurations- und anpassungsorientierter Modelle. Neue betriebswirtschaftliche Forschung* (Bd. 233). Gabler.
- Schütte, R., & Wulfert, T. (2022). Digital platforms and trading companies: The evolution of traditional business ecosystems to integrated digital business ecosystems. In S. Baumann (Ed.), *Handbook on Digital Business Ecosystems: Technologies, Markets, Business Models, Management, and Societal Challenges* (pp. 212–231). Edward Elgar. <https://doi.org/10.4337/9781839107191.00022>
- Schwarz, J. S., & Legner, C. (2020). Business model tools at the boundary: Exploring communities of practice and knowledge boundaries in business model innovation. *Electronic Markets*, 30(3), 421–445. <https://doi.org/10.1007/s12525-019-00379-2>
- Seidman, I. (2019). *Interviewing as qualitative research: A guide for researchers in education and the social sciences* (5th ed.). Teachers college press.
- Shanks, G. (2002). Guidelines for conducting positivist case study research in information systems. *Australasian Journal of Information Systems*, 10(1), 76–85. <https://doi.org/10.3127/ajis.v10i1.448>
- Shapiro, C., & Varian, H. R. L. B.-S. (1998). *Information rules: a strategic guide to the network economy*. Harvard Business Press.
- Shenton, A. K. (2004). Strategies for ensuring trustworthiness in qualitative research projects. *Education for Information*, 22(2), 63–75. <https://doi.org/10.3233/EFI-2004-22201>
- Shopify. (2022a). *API version release notes*. <https://shopify.dev/api/release-notes>. Accessed 21 Oct 2022.
- Shopify. (2022b). *App Design Guidelines*. <https://shopify.dev/apps/design-guidelines>. Accessed 21 Oct 2022.
- Shopify. (2022c). *Fulfillment API Deprecation*. <https://community.shopify.com/c/fulfillment-api-deprecation/bd-p/fulfillment-api>. Accessed 21 Oct 2022.
- Shopify. (2022d). *Partners and Developers*. <https://community.shopify.com/c/partners-and-developers/ct-p/appdev?shpxid=c60642d1-83D0-4E6D-FA2C-72EDF37CA21A>. Accessed 21 Oct 2022.
- Shopify. (2022e). *Shopify API versioning*. <https://shopify.dev/api/usage/versioning>. Accessed 21 Oct 2022.
- Shopify. (2022f). *Shopify Status*. <https://www.shopifystatus.com/>. Accessed 21 Oct 2022.
- Shopware. (2022a). *Debugging Shopware*. <https://developers.shopware.com/developers-guide/debugging/>. Accessed 21 Oct 2022.
- Shopware. (2022b). *Shopware 5 upgrade guide*. <https://developers.shopware.com/developers-guide/shopware-5-upgrade-guide-for-developers/>. Accessed 21 Oct 2022.
- Shopware. (2022c). *Shopware Roadmap*. <https://www.shopware.com/en/roadmap/>. Accessed 21 Oct 2022.
- Shopware. (2022d). *Versioning & Dependencies*. <https://developer.shopware.com/docs/guides/hosting/installation-updates/composer>. Accessed 21 Oct 2022.
- Shore, J., Larsen, D., Klitgaard, G., & Warden, S. (2022). *The Art of Agile Development* (2nd ed.). O'Reilly.
- Spaeth, S., & Niederhöfer, S. (2022). Compatibility promotion between platforms: The role of open technology standards and giant platforms. *Electronic Markets*, 32(4), 1891–1915. <https://doi.org/10.1007/s12525-022-00590-8>
- Star, S. L. (2010). This is not a boundary object: Reflections on the origin of a concept. *Science Technology and Human Values*, 35(5), 601–617. <https://doi.org/10.1177/0162243910377624>
- Star, S. L., & Griesemer, J. R. (1989). Institutional ecology, “translations” and boundary objects amateurs and professionals in Berkeley’s museum of vertebrate zoology. *Social Studies of Science*, 19, 387–420.
- Stark, J. (2011). *Product Lifecycle Management. 21st Century Paradigm for Product Realisation*. Springer. <https://doi.org/10.1007/978-0-85729-546-0>
- Stark, J. (2015). *Product Lifecycle Management: The Devil is in the Details* (2nd ed.). Springer. <https://doi.org/10.1007/978-3-319-17440-2>
- Staykova, K. S., & Damsgaard, J. (2015). A typology of multi-sided platforms: The core and the periphery. *Proceedings of the Twenty-Third European Conference on Information Systems (ECIS2015)* (pp. 1–17). Münster, Germany.
- StrategyR. (2022). *Application Lifecycle Management (ALM) World Market Report*. World Market Report <https://www.strategyr.com/market-report-application-lifecycle-management-alm-forecasts-global-industry-analysts-inc.asp>
- Straub, D., & Gefen, D. (2004) Validation guidelines for IS positivist research. *Communications of the Association for Information Systems*, 13(June 2014). <https://doi.org/10.17705/1cais.01324>
- Takeda, H., Veerkamp, P., Tomiyama, T., & Yoshikawa, H. (1990). Modeling design processes. *AI Magazine*, 11(4), 37–48. <https://doi.org/10.1609/aimag.v11i4.855>
- Tang, K., Zhang, J. M., & Feng, C. H. (2011) Application centric lifecycle framework in cloud. *2011 IEEE 8th International Conference on E-Business Engineering*, 329–334. <https://doi.org/10.1109/ICEBE.2011.32>
- Timmers, P. (1998). Business models for electronic markets. *Electronic Markets*, 8(2), 3–8.
- Tiwana, A. (2014). *Platform Ecosystems. Aligning Architecture, Governance, and Strategy*. Morgan Kaufman.
- Tiwana, A., Konsynski, B., & Bush, A. A. (2010). Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research*, 21(4), 675–687.
- Tüzün, E., Tekinerdogan, B., Macit, Y., & İnce, K. (2019). Adopting integrated application lifecycle management within a large-scale software company: An action research approach. *Journal of Systems and Software*, 149, 63–82. <https://doi.org/10.1016/j.jss.2018.11.021>
- Vaishnavi, V., Kuechler, B., & Petter, S. (2019) *Design science research in information systems*. <http://www.desrist.org/design-research-in-information-systems/>. Accessed 15 Oct 2021.
- Van Aken, J. E. (2004). Management research based on the paradigm of the design sciences: The quest for field-tested and grounded technological rules. *Journal of Management Studies*, 41(2), 219–246. <https://doi.org/10.1111/j.1467-6486.2004.00430.x>
- Vester, J. (2018). *RESTful API Lifecycle Management*. <https://dzone.com/refcardz/restful-api-lifecycle-management>. Accessed 18 Oct 2022.
- Vom Brocke, J. (2015). Referenzmodellierung. Gestaltung und Verteilung von Konstruktionsprozessen. In J. Becker, H. L. Grob, S. Klein, H. Kuchen, U. Müller-Funk, & G. Vossen (Eds.), *Advances in Information Systems and Management Science: Vol. 2. Auflage*. Logos Verlag.
- vom Brocke, J., Simons, A., Riemer, K., Niehaves, B., Plattfaut, R., & Cleven, A. (2015). Standing on the Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research. *Communications of the Association for Information Systems*, 37, 205–224. <https://doi.org/10.17705/1CAIS.03709>
- Vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Cleven, A., Brocke, J. Von, & Reimer, K. (2009). Reconstructing the giant: On the importance of rigour in documenting the literature search process. *Proceedings of the 17th European Conference on Information Systems (ECIS '17)* (pp. 2206–2217). Verona, Italy.

- Vom Brocke, J., Winter, R., Hevner, A., & Maedche, A. (2020). Special issue editorial – Accumulation and evolution of design knowledge in design science research: A journey through time and space. *Journal of the Association for Information Systems*, 21(3), 520–544. <https://doi.org/10.17705/1jais.00611>
- Walsham, G. (1995). Interpretive case studies in IS research: nature and method. *European Journal of Information Systems*, 4(2), 74–81. <https://doi.org/10.1057/ejis.1995.9>
- Walsham, G. (2006). Doing interpretive research. *European Journal of Information Systems*, 15(3), 320–330. <https://doi.org/10.1057/palgrave.ejis.3000589>
- Wareham, J., Fox, P. B., & Giner, J. L. C. (2014). Technology ecosystem governance. *Organization Science*, 25(4), 1195–1215. <https://doi.org/10.1287/orsc.2014.0895>
- Webster, J., & Watson, R. T. (2002). Analyzing the past to prepare for the future: Writing a literature review. *MIS Quarterly*, 26(2), xiii–xxiii.
- Weir, L. (2019). *Enterprise API Management: Design and deliver valuable business APIs*. Packt Publishing.
- Wilson, N. (2017). *Modernizing Application Development Primer*. <https://www.gartner.com/en/documents/3571618>. Accessed 20 Oct 2022.
- WooCommerce. (2022a). *Become a certified WooExpert*. <https://woocommerce.com/partner-program/>. Accessed 21 Oct 2022.
- WooCommerce. (2022b). *Store Editing Roadmap Update – Q3 2022*. <https://developer.woocommerce.com/category/roadmap-insights/>. Accessed 21 Oct 2022.
- WooCommerce. (2022c). *Woo Rest API*. <https://woocommerce.github.io/woocommerce-rest-api-docs/#introduction>. Accessed 21 Oct 2022.
- WooCommerce. (2022d). *WooCommerce Beta Tester*. <https://wordpress.org/plugins/woocommerce-beta-tester/>. Accessed 21 Oct 2022.
- WooCommerce. (2022e). *WooCommerce Developer Blog*. <https://developer.woocommerce.com/blog/>. Accessed 21 Oct 2022.
- WooCommerce. (2022f). *WooCommerce Documentation*. <https://woocommerce.com/documentation/>. Accessed 21 Oct 2022.
- WordPress. (2022). *Support WooCommerce Plugin*. <https://wordpress.org/support/plugin/woocommerce/>. Accessed 21 October 2022.
- Wulfert, T., & Karger, E. (2022). Shaping digital platforms in e-commerce: Developing an architecture framework. *Proceedings of the 26th Pacific Asia Conference on Information Systems (PACIS2022)* (pp. 1–17). Virtual Conference.
- Wulfert, T., Seufert, S., & Leyens, C. (2021). Developing multi-sided markets in dynamic electronic commerce ecosystems - Towards a taxonomy of digital marketplaces. *Proceedings of the 54th Hawaii International Conference on System Sciences (HICSS 2021)* (pp. 6133–6142). Maui, Hawaii.
- Wulfert, T., Woroch, R., Strobel, G., Seufert, S., & Möller, F. (2022). Developing design principles to standardize e-commerce ecosystems: A systematic literature review and multi-case study of boundary resources. *Electronic Markets*, 32(4), 1813–1842. <https://doi.org/10.1007/s12525-022-00558-8>
- Yi, X., & Ming, X. (2011). Study on competitive advantage and construction strategy of e-commerce ecosystem. *Proceedings of the 4th International Conference on Business Intelligence and Financial Engineering*, 207–210. <https://doi.org/10.1109/BIFE.2011.111>
- Yin, R. K. (1981). The case study as a serious research strategy. *Science Communication*, 3(1), 97–114.
- Yin, R. K. (2018). *Case study research: design and methods (6th ed.)*. Sage Publications.
- Yoo, B., & Jang, M. (2019). A bibliographic survey of business models, service relationships, and technology in electronic commerce. *Electronic Commerce Research and Applications*, 33, 100818. <https://doi.org/10.1016/j.elerap.2018.11.005>
- Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research*, 21(4), 724–735. <https://doi.org/10.1287/isre.1100.0322>
- Yu, D.-F., Chang, C.-Y., Jiau, H. C., & Ssu, K. F. (2017). Which API lifecycle model is the best for API removal management? *Proceedings of the 12th International Conference on Software Engineering Advances (ICSEA2017)* (pp. 219–224). Athens, Greece.
- Zeng, Y., Cheng, Y., Xie, G., & Wang, R. (2021). Design of mobile application lifecycle security management platform. *2021 International Conference on Computer Network, Electronic and Automation (ICCNEA)*, 26–30. <https://doi.org/10.1109/ICCNEA53019.2021.00017>
- Zulian, A., & Bouza, A. (2021). *API Product Management: Product Strategy and Execution for the Digital Economy*. leanpub.
- Zutshi, A., & Grilo, A. (2019). The Emergence of digital platforms: A conceptual platform architecture and impact on industrial engineering. *Computers and Industrial Engineering*, 136(July), 546–555. <https://doi.org/10.1016/j.cie.2019.07.027>
- Zwass, V. (1996). Electronic commerce: Structures and issues. *International Journal of Electronic Commerce*, 1(1), 3–23. <https://doi.org/10.1080/10864415.1996.11518273>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.