Friedrich-Alexander-Universität
Faculty of Engineering

Dissertation

2025

Elçin Yenişen Yavuz

Creating Company-Led Open Source Consortia

Creating Company-Led Open Source Consortia

Erstellung unternehmensgeführter Open-Source-Konsortien

Der technischen Fakultät
der Friedrich-Alexander-Universität
Erlangen-Nürnberg

zur

Erlangung des Doktorgrades
Doktor-Ingenieur (Dr.-Ing.)

vorgelegt von

ELÇİN YENİŞEN YAVUZ
aus Izmit

Als Dissertation genehmigt
von der Technischen Fakultät
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der mündlichen Prüfung:	9. Mai 2025
Gutachter:	Prof. Dr. Dirk Riehle
	Prof. Dr. Gregorio Robles
	Prof. Dr. Sven Laumer

Abstract

This thesis focuses on company-led open source (OS) consortia. Unlike community-led open source communities, company-led OS consortia are led by companies or organizations. Driver members of these consortia may include information technology firms or end-user companies from industries beyond the software sector.

The objectives of this thesis are to identify the characteristics of company-led OS consortia, examine the motives of organizations for joining such consortia, and outline practices for successful collaboration. To analyze these aspects, we distinguish between two types of company-led OS consortia: user-led open source consortia and vendor-led open source consortia.

This thesis is based on the synthesis of four studies conducted by the author in collaboration with co-authors and published between 2022 and 2025. These studies include a systematic literature review on user-led OS consortia; a single-case case study on user-led OS consortia, focusing on the openMDM consortium; a single-case case study on vendor-led OS consortia, focusing on the LF Edge consortium, and a multiple-case case study on both user-led and vendor-led OS consortia.

The key contributions of this thesis include establishing a terminology for company-led OS consortia, defining the main characteristics of user-led and vendor-led OS consortia, identifying the motivations behind company involvement in these consortia, and determining 90 practices across 26 different contexts to support the success of company-led OS consortia.

Zusammenfassung

Diese Dissertation konzentriert sich auf unternehmensgeführte Open-Source-(OS)-Konsortien. Im Gegensatz zu gemeinschaftsgeführten Open-Source-Communities werden unternehmensgeführte OS Konsortien von Unternehmen oder Organisationen geleitet. Die führenden Mitglieder dieser Konsortien können Unternehmen aus der Informationstechnologiebranche oder Endnutzerunternehmen aus anderen Wirtschaftszweigen sein.

Die Ziele dieser Dissertation sind die Identifizierung der Merkmale unternehmensgeführter OS-Konsortien, die Untersuchung der Beweggründe von Organisationen für den Beitritt zu solchen Konsortien sowie die Darstellung von Erfolgsmethoden für eine erfolgreiche Zusammenarbeit. Zur Analyse dieser Aspekte unterscheiden wir zwischen zwei Arten von unternehmensgeführten OS-Konsortien: anwendergeführten Open-Source-Konsortien und anbietergeführten Open-Source-Konsortien.

Diese Dissertation basiert auf der Synthese von vier Studien, die von der Autorin in Zusammenarbeit mit Koautoren durchgeführt und zwischen 2022 und 2025 veröffentlicht wurden. Dazu gehören eine systematische Literaturübersicht über anwendergeführte OS-Konsortien, eine Einzelfallstudie zu anwendergeführten OS-Konsortien mit Fokus auf das openMDM-Konsortium, eine Einzelfallstudie zu anbietergeführten OS-Konsortien mit Fokus auf das LF Edge-Konsortium sowie eine Mehrfallstudie zu sowohl anwender- als auch anbietergeführten OS-Konsortien.

Die wesentlichen Beiträge dieser Dissertation umfassen die Etablierung einer Terminologie für unternehmensgeführte OS-Konsortien, die Definition der Hauptmerkmale von anwender- und anbietergeführten OS-Konsortien, die Identifikation der Beweggründe für die Beteiligung von Unternehmen an diesen Konsortien sowie die Bestimmung von 90 Erfolgsmethoden in 26 verschiedenen Kontexten zur Unterstützung des Erfolgs unternehmensgeführter OS-Konsortien.

Contents

List of Figures

List of Tables

List of Acronyms

ASWF	The Academy Software Foundation
BSD	Berkeley Software Distribution License
EF	The Eclipse Foundation
EHR	Electronic Health Record
IP	Intellectual Property
GPL	General Public License
KFS	Kuali Financial Systems
LF	The Linux Foundation
LMS	Learning Management System
OS	Open Source
OSF	Open Source Foundations
OSP	Open Source Portfolio
OSS	Open source software
SLR	Systematic Literature Review

Acknowledgments

Throughout my journey to complete this work, I was supported by many people, to whom I wish to express my deepest gratitude.

I would like to express my gratitude to my supervisor, Prof. Dirk Riehle, for providing me with the opportunity to work on this research topic. His support, guidance, and ideas have helped me pursue this research every step of the way.

I would like to extend my sincere thanks to Ann Barcomb for her supervision and friendship, to my colleagues and friends on the Open Source Research Group for all the feedback they provided to my research, and for their friendly support.

I would like to thank Andreas Kaufmann, Georg Schwarz, Julia Mucha, and Stefan Buchner for their invaluable friendship and enduring support.

I thank my life-long friends, Ersoy Akın and Barbaros Gülşah, for their constant support, always being there to listen, and motivating me whenever I needed it.

I am beyond grateful to my father, Mesut Yenişen, and my mother, Hanife Yenişen, for supporting all my decisions in life.

I sincerely thank my husband, Özgür Yavuz, for standing by my side through every moment, celebrating my joys, lifting my spirits, and offering boundless support.

I am forever grateful to my baby son, who gave me hope and faith at the most unexpected time.

Related Publications

In this dissertation, the third-person plural ("we") is used throughout. This choice reflects the cumulative nature of the work, as all contributing papers were co-authored with other researchers. To acknowledge the valuable contributions of my co-authors and enhance the readability of the thesis, I will consistently use "we" instead of "I."

This cumulative dissertation comprises four research papers, for which I served as the lead author. In each of the four studies, I undertook the role of main author, making the major contributions to the overall results. The following studies are included in this thesis:

- Paper 1 (P1): Yenişen Yavuz, E., & Riehle, D. (2025) Why and How Do Organizations Create User-Led Open Source Consortia? A Systematic Literature Review. In Information and Software Technology, 107681.

- Paper 2 (P2): Yenişen Yavuz, E., Barcomb, A., & Riehle, D. (2022). Problems, solutions, and success factors in the openMDM user-led open source consortium. Communications of the Association for Information Systems, 51(1), 13.

- Paper 3 (P3): Yenişen Yavuz, E., Riehle, D., & Mehrotra, A. (2025). Why do companies create and how do they succeed with a vendor-led open source foundation. Empirical Software Engineering, 30(1), 1-49.

- Paper 4 (P4): Yenişen Yavuz, E., Shrivastava, A., Riehle, D., & Putz, F. (2025). Governance Practices for Open Source Foundations in the Healthcare Sector. In: Papatheocharous, E., Farshidi, S., Jansen, S., Hyrynsalmi, S. (eds) Software Business. ICSOB 2024. LNBIP, vol 539. Springer, Cham.

1

Introduction

Open Source Software (OSS) has a long-standing history in the software industry. Since the emergence of the 'open source' concept, the OSS development approach has driven innovation in both software development and business domains (Riehle, 2019). From a business perspective, OSS is regarded as a prime example of open innovation due to its collaborative nature and openness (West & Gallagher, 2004).

The origins of the OSS movement date back to 1985, when Richard Stallman introduced the "free software" movement as a reaction to the development of proprietary software packages. Stallman introduced the 'copyleft' concept and the 'General Public License (GPL),' which grants users basic rights over software code, such as the right to use, modify, and distribute it, ensuring that these freedoms remain available to all future users. Since the term 'free software' created skepticism, Bruce Perens and Eric Raymond introduced the term 'open source' in 1998 to describe the approach of 'free software development' (von Hippel & von Krogh, 2003).

OSS projects initially emerged within developer communities, followed by the establishment of open source (OS) foundations to support and govern them. In 1999, the Apache Software Foundation was founded to ensure the sustainability of the Apache Web (HTTP) Server project. In 2000, the Linux Foundation was established to support the growing Linux kernel community. Similarly, in 2004, the Eclipse Foundation was established to support the Eclipse IDE project (Hunter & Walli, 2013). The growing popularity of OSS projects captured the attention of software companies over time, leading them to actively participate in these projects (Fitzgerald, 2006). A common strategy that corporate organizations adopted is open-sourcing their in-house developed code and building a community around it (West & O'Mahony, 2005; Harutyunyan et al., 2020). Another strategy is assigning employed developers to work on community-led OSS projects, which include a diverse community of developers from other companies and volunteers. Community-led OSS projects accept individual contributors as members, rather than companies or corporate entities. These projects operate under a meritocratic governance model, where contributors earn governance roles through consistent and high-quality contributions (Riehle & Berschneider, 2012; Weikert et al., 2019). Although corporate entities are not formally recognized as members, they can support hosted OSS projects by funding individual contributors, offering infrastructure resources, or sponsoring project-related events (Shaikh & Cornford, 2010).

The Apache Foundation is an example of a community-led open-source foundation that hosts and supports community-led OSS projects. While companies can sponsor projects within the Apache Foundation, they cannot become project members as corporate entities.

An emerging strategy is collaborating with sector counterparts to create value together. In these collaborations, companies sign contractual agreements and operate under established governance rules. They become members of the collaboratively established open source consortia. Unlike community-led projects, they do not rely on volunteer developers; instead, they pool their resources for OSS development. We refer to these models as 'company-led open source consortia.' This model differs from community-led OS consortia, where leading members are individuals, whereas in company-led OS consortia, the leading members are companies. It is adopted by both software vendors and end-user companies operating beyond the software industry.

The focus of this thesis is company-led open source consortia. We aim to identify the characteristics of company-led OS consortia, motives of organizations to involve in these types of consortia, and good practices to follow for successful collaboration in company-led OS consortia. To understand the characteristics and member's motives, we distinguish the two different types of such consortia. These are: user-led open source consortia, and vendor-led open source consortia.

We refer to consortia consisting of end-user organizations from non-software industries as leading members who collaboratively work on OSS development projects as user-led open source consortia. These consortia collaborate with the goal of developing open-source software applications to use in their internal processes. End-user organizations steer OSS development by providing requirements and financial incentives. Software vendors also participate in these consortia but primarily act as development partners, implementing specifications and developing the software. Two examples of such consortia are the Open Logistics Foundation and the openMDM consortium.

We refer to consortia consisting of information technology (IT) companies as leading members that collaboratively work on OSS development projects as vendor-led open source consortia. These consortia collaborate with the goal of developing undifferentiated open-source software components (Schaarschmidt et al., 2011; Riehle & Berschneider, 2012). Two examples of this type of consortium are the OpenInfra Foundation and the LF Edge Foundation.

Figure 1.1 illustrates the differences between community-led, vendor-led, and user-led OS consortia, focusing on their leading members and leading members' primary interests.

In this thesis, we use the terms 'foundation' and 'consortium' interchangeably. Riehle & Berschneider (2012) distinguish between these terms based on their goals—whether they serve their members or the public—, and their legal jurisdiction, which varies by the country in which they are established.

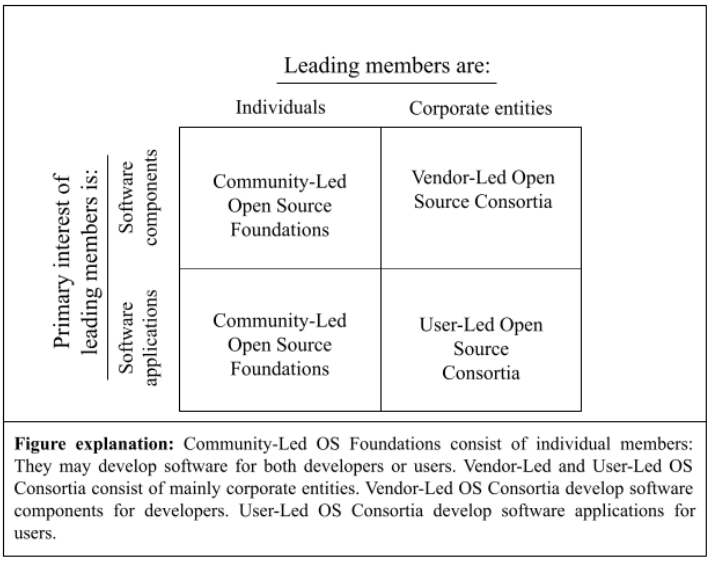

Figure 1.1: Classification of Open Source Consortia

In company-led OS consortia, members primarily collaborated for their own benefits rather than for the public interest. Therefore, we refer to this approach as “consortia”. However, the terminology used to describe these consortia differs depending on the country in which they are established. Rather than excluding cases based on legal distinctions, we focus on their structure and objectives in relation to software development. For this reason, we use both terms to describe our model.

2

Aim of the Thesis

This section outlines the aim and research questions of this thesis, and provides an overview of its structure.

2.1 Goals and Research Question

User involvement in OSS development projects has been common practice since the early days of OSS development movement. The literature has examined this involvement from various perspectives. However, these examined 'users' were primarily volunteer developers who contribute to the software they use and have played a leading role in OSS development for many years.

User organizations' involvement in OSS development began in the higher education sector in the early 2000s. The first examples of user-led OS consortia projects, such as Kuali and Sakai, were initiated by universities in the United States to address the need for customized software solutions. The success of these projects led to a rise in similar initiatives, not only within the education sector but also across other industries, including automotive, energy, entertainment, finance, healthcare, transportation, and logistics.

Although interest in this approach has increased, research on user-led OS consortia remains limited. Most of the publications focus on the projects originated in the education industry and consist primarily of experience papers rather than research articles. We observed a lack of systematic investigation in this area, as well as the absence of a comprehensive explanation of this model's structure across different industries.

On the other hand, vendor involvement in the OSS development movement has been examined from various perspectives. However, existing studies primarily focus on this topic from the perspective of developers or individual companies. We identified a lack of research on vendor-led OSS consortia, particularly regarding governance and sustainability aspects. Additionally, we observed a lack of established terminology in the literature to define the vendor-led OS consortia approach.

Furthermore, for both user-led and vendor-led OS consortia, a gap remains in the literature regarding practices for successful collaborations.

This thesis aims to address these gaps based on the findings of four research studies (P1, P2, P3, P4) that we published between 2022 and 2025.

To address these gaps, our goal is to identify the characteristics of company-led open-source (OS) consortia and the motivations of companies to create or participate in these consortia. To explore these characteristics and motivations, user-led OS consortia and vendor-led OS consortia topics separately.

Furthermore, we aim to identify the practices for the successful establishment and governance of company-led open-source consortia.

The research questions of this thesis are:

RQ1: What are the characteristics of company-led open source consortia?

RQ1.1: What are the characteristics of user-led open source consortia?

RQ1.2: What are the characteristics of vendor-led open source consortia?

RQ2: What are the motivations of companies to create or involve in company-led OS consortia?

RQ2.1: What are the motivations of companies to create or involve in user-led open source consortia?

RQ2.2: What are the motivations of companies to create or involve in vendor-led open source consortia?

RQ3: What are the good practices for successful company-led open source consortia?

2.2 Structure of the Thesis

This thesis follows a cumulative format, consisting of four peer-reviewed and published research articles (P1, P2, P3, P4) that collectively address the overarching research questions. Table 3.1 presents an overview of the four studies that form the basis of this thesis. It highlights the research and data analysis methods followed in each study and provides a brief summary of results' content.

Chapter 3 provides an overview of the state of the art in company-led open source consortia literature with a focus on user-led open source consortia and vendor-led open source consortia. Furthermore, Chapter 3 presents the success factors for inter-company collaborations as outlined in the literature.

Chapter 4 introduces the methods followed to conduct studies forming this research. In P1, we performed a systematic literature review (SLR) research. We followed an SLR method focusing on Kitchenham's guidelines (Kitchenham, 2004; Kitchenham & Brereton, 2013). After defining the related papers about our research focus, we qualitatively analyzed the content of the selected articles. For this purpose, we followed the thematic analysis guidelines of Braun & Clarke (2006). This process is explained in Sections 4.1 and 4.3.

In P2, P3, and P4, we performed case study research by following guidelines of Yin (2018) and Eisenhardt (1989). The details of the case study research approach is explained in subsection 4.2. In P2 and P3, we followed the coding paradigm of the grounded theory research method by following Strauss & Corbin (1990). This approach is explained in Section 4.4. For the P4, we followed thematic analysis (Braun & Clarke, 2006) which is explained in 4.3.

Chapter 5 reports the answers to the research questions in this thesis by synthesizing the key findings of our research papers. These findings serve as the basis of this cumulative dissertation.

P1 addresses the research questions RQ1.1, and RQ2.1 by presenting the characteristics of user-led OS consortia and motivations of user organizations involvement. The detailed results are presented in Chapter 5.1 and 5.2.

P2 addresses the research question RQ3 by highlighting the problems encountered in a user-led OS consortium, and the solutions applied to solve these problems. The detailed results are presented in Chapter 5.3.

P3 addresses the research questions RQ1.2, RQ2.2, and RQ3 by defining the companies' motivations to involve in vendor-led OS consortia, the problems encountered in a vendor-led OS consortium, and governance practices applied to solve these problems. The results are presented in Chapter 5.2 and 5.3.

P4 addresses both the RQ2 and RQ3 by comparing and contrasting user-led and vendor-led OS consortia. Key findings are presented in Chapter 5.2 and 5.3.

Chapter 6 discusses the findings of this thesis.

Chapter 7 concludes this thesis by emphasizing insights valuable to practitioners, and summarizing our key findings.

Table 2.1: Overview of Methodology and Content of Peer-Reviewed Articles Presented in This Thesis

Study ID	Research Method	Data Analysis Method	Content
P1	Systematic Literature Review	Thematic Analysis	• The defining characteristics of user-led open source consortia • Companies motivations to engage with user-led OS consortia • Roles and actors in user-led OS consortia environment
P2	Case Study Research	Grounded Theory Approach	• Problems of user-led OS consortia • Practices to solve current and potential problems in user-led OS consortia
P3	Case Study Research	Grounded Theory Approach	• Companies motivations to engage with vendor-led OS consortia • Problems of vendor-led OS consortia • Practices to solve current and potential problems in vendor-led OS consortia
P4	Case Study Research	Thematic Analysis	• Differences and similarities between vendor-led and user-led OS consortia • Governance practices to solve current and potential problems in user-led and vendor-led OS consortia

3

State of the Art

This section reviews the literature on user-led open source consortia and vendor-led open source consortia, and presents the success factors for company collaborations as outlined in the literature.

3.1 State of the Art of User-led Open Source Consortia Literature

The first examples of user-led open source consortia emerged in higher education in the United States starting in the 2000s. Early projects, such as Sakai Learning Management Systems (LMS) and Kuali Financial Systems (KFS), appeared as alternatives to the traditional build-or-buy decisions faced by universities (Wheeler, 2007a). Around these projects, the first user-led open source consortia evolved. Since then, various user-led open-source consortia projects have emerged in higher education as well as in other sectors.

This university-led collaborative open-source software (OSS) development approach is referred to as 'directed open source' by Courant & Griffith (2006) and as 'community-source software development' by Wheeler (2007a) and Liu et al. (2007).

Studies exploring the structure of this model, especially within the context of higher education, are predominantly experiential or opinion-based papers written by university members involved in user-led OS consortia projects (Yenişen Yavuz & Riehle, 2025). Research on user-led OS consortia outside the higher education industry started emerging from 2013 onward (Yenişen Yavuz & Riehle, 2025).

Section 2.1.1 presents user-led open-source consortia and projects in higher education as discussed in the literature. Section 2.1.2 focuses on user-led open-source consortia and projects beyond higher education. Section 2.1.3 reviews the literature on the motives behind organizations' involvement in user-led open-source consortia. Section 2.1.4 examines the literature on problems and solutions in user-led open-source consortia.

3.1.1 Literature on User-Led Open Source Consortia in Higher Education

Most of the studies emerged around the projects hosted by two umbrella foundations in the higher education sector: The Kuali Foundation and the Apereo Foundation.

The Kuali Foundation was established in 2004 to ensure the financial sustainability of the Kuali Financial Systems (KFS) project (Foutty, 2010). The KFS project was initiated by Indiana University and the University of Hawaii with the goal of developing software to support their internal finance management processes. As other universities with similar needs joined, they formed a consortium around the project (Liu et al., 2012). Following the success of the KFS project, the foundation expanded its range of projects to address various needs of universities. Some of these projects include Kuali Coeus (a research administration system), Kuali Student (a student information system), and Kuali Rice (middleware applications) (Liu et al., 2020). The Kuali Foundation and its hosted projects have been the subject of numerous research publications, including Bulushi (2019), Liu et al. (2012), Liu et al. (2014a), Liu et al. (2020), and Liu et al. (2021).

The Apereo Foundation was established in 2012. The projects it hosts include Sakai LMS, Open Source Portfolio (OSP) and OpenCast.

The ***Sakai*** LMS project was initiated by members of four universities—the University of Michigan, Indiana University, Massachusetts Institute of Technology, and Stanford University—along with two research projects, uPortal and the Open Knowledge Initiative (OKI), in the USA. At that time, these universities were independently developing their own in-house learning management systems. They decided to pool their resources and collaboratively develop a system that would meet their needs. The project was officially launched in January 2004 (Severance, 2011; Wheeler, 2007b).

In its first two years, the project was externally funded. After the funding period ended, Sakai members launched an initiative to ensure the project's sustainability both financially and functionally. This initiative began as the 'Sakai Educational Partners Program,' later evolved into the Sakai Foundation, and ultimately became the Apereo Foundation (Severance, 2011). Along with its governance structure and its establishment process is detailed by the project's founders in Severance (2011), and Wheeler (2007b).

Open Source Portfolio (OSP) is an online e-portfolio for teaching, assessment, and accreditation. Nidy Kwok (2005) shared their experiences from their involvement in the development process of OSP.

Opencast (Matterhorn) is an open source video recording and management system designed for lecture use. openCast consortium was initiated in 2008 with the goal of leveraging a collaborative effort to reduce the cost of developing and hosting a scalable podcasting infrastructure that was flexible enough to meet the needs of various universities (Hancock, 2010). Hancock (2010) and Ketterl et al. (2010) described the establishment process of the Opencast community and provided detailed information about the project.

e-Presence was a web-based streaming tool for large-scale broadcast of events over the Internet. Initially developed as an in-house project by the University of Toronto, ePresence became an open-source project in 2005 when the university decided to release it and form a consortium around it. The primary motivation behind this decision was to offer users the flexibility to customize the system according to their specific needs (Rankin & Baecker; 2007). In their paper, Rankin & Baecker (2007) shared their experiences on how the project was open-sourced.

3.1.2 Literature on User-Led Open Source Consortia Beyond Higher Education

Beyond higher education, studies on user-led open source consortia focus on cases from the energy, entertainment, finance, governance, and library industries.

An investigated example from the energy sector is the ***openKonsequenz*** consortium. It was initiated by energy providers in Germany in 2013. The goal of the consortium is collaborative software development for energy grid operation management (Goering et al., 2017). Schwab et al. (2020) investigated the ecosystem of the openKonsequenz. They defined that openKonsequenz consists of three membership types: (1) energy company providers as the funders and drivers of the development direction, (2) software vendors who contribute to the software development process, (3) consultants and research groups. Goering et al. (2017) focused on the reference architecture of the openKonsequenz platform.

The Academy Software Foundation (ASWF) is an example from the entertainment industry. It was initiated in 2018 by motion picture and visual effects studios in the United States (Heckenberg et al., 2019). Heckenberg et al. (2019) provide information about the structure and goals of ASWF's Technical Advisory Board. OpenColorIO is one of the projects hosted by ASWF. Walker et al. (2020) discuss the origins of the OpenColorIO project, as being initiated by Sony Pictures Imageworks, and the reasons for the project's involvement in ASWF. The main reason was the decline in the community engagement on the project, and its need for support. After joining the ASWF, the project experienced a revival, since the members in the ASWF provided financial and intellectual support for the project's long-term sustainability (Walker et al., 2020).

openMAMA is a consortium in the finance industry. It was initiated by financial institutions in the United States in 2010. The main focus of the openMAMA is providing a data transfer platform in the finance industry (Germonprez et al., 2013, Levy & Germonprez, 2015).

FOLIO and Hyku for Consortia are examples from the library sector. ***FOLIO*** was initiated in 2010 with the name of Kuali OLE as a library system project of the Kuali Foundation (Winkler, 2018). Since the Kuali members changed their focus from providing open source projects, to being a co-profit company, in 2016 OLE project left the consortium and initiated its own foundation as the Open Library Foundation (OLF) (Winkler, 2018). Winkler (2018) shared his experience during this transition period. ***Hyku for Consortia*** is a collaborative initiative aimed at building an open-source institutional repository for libraries. The project was initiated by the Pennsylvania Academic Library Consortium (PALC) and the Private Academic Library Network of Indiana (PALNI) (Morris and Leonard, 2020). Morris and Leonard (2020) shared their experiences regarding the creation and management process of this consortium.

Oskari and X-Road are publicly funded governance projects. ***Oskari*** is a geospatial software project supported by the National Land Survey of Finland (NLSF) and a community of organizations called the Joint Development Group (Henttonen et al., 2017). Henttonen et al. (2017) investigated the Oskari consortium to offer a framework for the lifecycle management practices of government-driven OSS projects. With this framework, they aimed to provide guidance to OSS consortia of public sector organizations.

X-Road is a data exchange platform, initiated in 2017. It is developed by the Nordic Institute for Interoperability Solutions (NIIS) consortium which is established by governmental organizations in Estonia and Finland (Robles et al., 2019). Robles et al. (2019) examined the organizational structure, contributors, and stakeholder-perceived challenges of the X-Road project.

3.1.3 Motivations of Members' Involvement in User-Led Open Source Consortia

In the literature, we identified three studies focusing on the motivations behind organizations' involvement in user-led OS consortia. These studies were conducted by Liu et al. and focused on the Kuali Foundation.

Liu et al. (2014a) highlighted the motivations of user organizations to participate in these collaborations, including reducing development costs, decreasing dependence on vendors, expanding options for system customization, and enhancing staff development through gaining system expertise and social interaction within the community.

According to Liu et al. (2017), decision-makers in institutions consider the characteristics of consortia and opportunities in the industry when deciding whether to join. For instance, they take into account established norms, governance structures, and monitoring mechanisms of consortia, member organizations in the consortia, the possibility of receiving external funding for projects, vendor behaviors in the industry, and the usability of information technology in the related industry. When developers from member organizations work on projects, they gain expertise, allowing these organizations to deploy the system faster and at a lower cost than institutions that were not involved. Learning opportunities considered as another motive for joining user-led OS consortia (Liu et al., 2017). Moreover, organizations' decisions about involvement in user-led OS consortia are also affected by their own size, financial power, and IT capabilities (Liu et al., 2014b).

3.1.4 Problems and Solutions in User-Led Open Source Consortia

In the literature, we identified six studies that discuss the challenges faced by user-led open-source consortia. Five of these studies focused on software development projects in the higher education industry (Kuali Financial System, Kuali Rice, Sakai LMS, Kuali Foundation and projects, and ePresence), while one examined a government-led project (X-Road).

Liu et al. (2010) focused on the development aspect of the ***Kuali Financial Systems*** project. The challenges they highlighted include finding and retaining skilled developers, and ensuring the sustainability of projects. The proposed solution for these problems was outsourcing developers instead of working with employees in the member universities (Liu et al., 2010).

The diverse expectations and needs of member organizations created another challenge. By focusing on the ***Kuali Rice*** and ***Sakai LMS*** projects, Liu et al. (2012) proposed a solution to this problem by ensuring technological flexibility and enabling customization.

Ensuring the continuity of user-led OS collaborations is another topic discussed in the literature by Liu et al. (2020). Liu et al. (2020) examined this issue in the context of the ***Kuali community***, including both its projects and foundation. The challenges they highlighted include governance of the community, roles of commercial affiliates, providing a friendly atmosphere, sharing knowledge across projects, and coordination of projects. The proposed solution for these problems is ensuring a modular organizational design in the collaboration (Liu et al., 2020; Liu et al., 2021).

The ***ePresence*** project encountered several challenges, such as creating high-quality software with distributed development teams, maintaining an engaged community, selecting an appropriate license, and establishing a revenue model (Baecker, 2005). To address the issue of generating revenue from the OSS product, the consortium initially implemented a "dual license" approach. However, this strategy did not lead to success. A key problem with the dual-license model was the need to manage two separate yet interconnected software packages, which led the consortium to duplicate efforts during each release cycle. This not only consumed considerable time but also caused usability issues (Rankin & Baecker, 2007). Ultimately, the consortium streamlined the licensing approach by offering ePresence under a single open-source license, the BSD license (Rankin & Baecker, 2007).

The ***X-Road*** project is led by a consortium consisting of governmental organizations. Observed challenges include the complexity of the onboarding process for new developers, vendors' lack of knowledge about the project and its technologies, the absence of contributions from private sector companies using X-Road. Furthermore, the project faced challenges in gaining new members at a slow pace due to bureaucratic processes in governmental organizations (Robles et al., 2019).

3.2 State of the Art of Vendor-Led Open Source Consortia Literature

Vendor engagement within OSS communities and projects has been examined in the literature from various perspectives. One key topic is ***software companies' engagement strategies*** with open source communities and the benefits these strategies offer to companies.

Grand et al. (2004) investigated the motivations and methods behind software and IT companies' engagement in OSS communities. They identified four levels of involvement, along with the benefits of each: (1) being a user of OSS, (2) using OSS as a complementary asset, (3) contributing to OSS projects and developing their own software based on OSS code, and (4) adopting OSS as a compatible business model, such as offering services for OSS.

West & Gallagher (2006) discussed company strategies for engaging in OSS projects and the challenges they face. The strategies they highlighted include pooling research and development efforts, for example, to compete against a rival product; spinning out proprietary code as open source for public use; selling complementary products or services for OSS; and donating certain software components to encourage users to develop their own modifications.

Shaikh & Cornford (2010) explored the engagement dynamics of companies in community-led OS projects. Their research was based on interviews with employees from two global technology companies, each of which dedicates more than 1,000 developers to OSS projects. They identified challenges in the requirements process, key factors to consider when assessing the total cost of adoption, and effective strategies for collaborating with OSS communities.

Riehle (2010) discussed the involvement of software vendors in OSS communities from an economic perspective.

Joo et al. (2012) examined the involvement of commercial software companies in OSS projects by assigning their developers to contribute. Their study focused on OSS projects influenced by IBM, Oracle, and Google.

Schaarschmidt & Stol (2018) examined companies' involvement in OSS communities from the perspective of developers.

Another topic is the ***open sourcing of proprietary code*** and the development of a community around it. Agerfalk & Fitzgerald (2008) focused on open sourcing proprietary code to OSS communities. They identified the obligations of companies and communities to build a sustainable ecosystem. Linåker & Regnell (2020) focused on the vendors' strategies on open sourcing their proprietary codes, by highlighting the advantages and disadvantages of this approach for companies.

Recently, research on ***multi-vendor OSS projects*** has gained attention.

Schaarschmidt et al. (2011) highlighted different governance approaches in single-vendor and multi-vendor OSS projects, with a focus on the Eclipse Foundation.

Teixeira & Lin (2014) introduced the term 'open-coopetition' to describe the collaboration between competing companies in OSS projects. Teixeira et al. (2016) analyzed the collaboration structure within the OpenStack ecosystem, focusing on the interactions between developers from rival companies collaborating on the same OSS projects.

Weikert et al. (2019) examined the conflict of interest problem and mechanisms to mitigate it in open source foundations, focusing on the Apache CloudStack, Cloud Foundry, Eclipse, and OpenStack foundations.

Zhang et al. (2020, 2022a, 2022b) focused on companies' involvement and collaboration dynamics within the OpenStack ecosystem. They investigated companies' engagement strategies and collaboration patterns (Zhang et al., 2020), the extent of company contributions to the development of OpenStack, company contribution models, and the impact of company diversity on the ecosystem (Zhang et al., 2021), the effects of company dominance in OSS ecosystems (Zhang et al., 2022b), and the consequences of companies withdrawing their employees from projects.

In Sections 2.2.1 and 2.2.2, we present a synthesis of these studies, focusing on the motivations behind vendor involvement in OSS projects and the problems and solutions vendors experience in OSS projects, respectively.

3.2.1 Motivations of Vendor Involvement in Open Source Software Projects

We present the findings of the literature synthesis on vendor motives for participating in OSS projects, categorized into four key dimensions: (1) revenue, (2) competition, (3) productivity and innovation, and (4) reputation.

Revenue

One of the key motives for vendors to open-source their software is the potential to ***generate revenue*** through complementary products and services associated with it (Grand et al. 2004; West & Gallagher, 2006; Schaarschmidt et al., 2011; Teixeira et al., 2016; Zhang et al., 2020; Zhang et al., 2021). Furthermore, by open-sourcing their software, companies aim to expand the software's adoption and user base, ultimately securing a dominant position in the market (Grand et al. 2004; West & Gallagher, 2006; Joo et al., 2012).

Another motive is ***reducing software development and maintenance costs***. By engaging in OSS projects, vendors collaborate with community members to share resources and expertise, ultimately lowering software development expenses (Shaikh & Cornford, 2009; Riehle, 2010; Teixeira et al., 2016). Moreover, by using open-source code, companies can develop complementary products, which can, in turn, increase the number of suppliers and service providers for these open-source software solutions (Ägerfalk & Fitzgerald, 2008; Linåker & Regnell, 2020). An expanded supplier market enables companies to lower their maintenance costs (Linåker & Regnell, 2020).

Competition

A further motive for companies to contribute to OSS projects is to ***compete with dominant competitors***. One approach is open-sourcing their product's source code, attracting developers, users, and other companies to contribute, adopt, and innovate (West & Gallagher, 2006; Joo et al., 2012). Another strategy involves collaborating with other vendors to develop a substitute product that challenges market leaders (Teixeira et al., 2016; Weikert et al., 2019).

Vendors can engage in OSS projects and communities to ***establish standards*** in the industry. For instance, vendors apply this approach by donating their source code to OSS communities or joining already established communities to influence on the development direction (West & Gallagher, 2006; Riehle, 2010; Schaarschmidt et al., 2011; Teixeira et al., 2016; Linåker & Regnell, 2020). By establishing standards, companies can gain first-mover advantages and strengthen their influence on a community or industry (Linåker & Regnell, 2020).

Productivity and Innovation

Accessing external resources is another key motive for companies to engage in OSS projects. This can take the form of leveraging contributions from external developers within OSS communities (Grand et al., 2004; West & Gallagher, 2006; Ägerfalk & Fitzgerald, 2008; Shaikh & Cornford, 2009) or, in the case of multi-vendor OSS projects, acquiring knowledge and expertise from other companies (Schaarschmidt, 2011; Teixeira et al., 2016; Zhang et al., 2020).

Furthermore, OSS engagement can ***accelerate innovation*** in the field. Collaborating with OSS communities enables companies to gather user feedback, explore innovative ideas, and receive support for testing and quality assurance (Grand et al., 2004; Iivari et al., 2008; Shaikh & Cornford, 2009; Teixeira et al., 2016; Linåker & Regnell, 2020). By reducing software development time and expenses, organizations can allocate internal resources to differentiating activities, such as user interface design (Iivari et al., 2008; Linåker & Regnell, 2020). Moreover, data and insights gathered from OSS communities help companies align with customer expectations and contribute to the creation of new products and services (Grand et al. 2004; Linåker & Regnell, 2020).

Reputation

Engagement with OSS communities positively impacts ***employee satisfaction*** and enhances a company's ***credibility*** in the eyes of both customers and potential employees (Grand et al. 2004; West & Gallagher, 2006; Ägerfalk & Fitzgerald, 2008; Shaikh & Cornford, 2009; Linåker & Regnell, 2020).

Involvement in multi-vendor projects creates a positive impression among other vendors, as it showcases a thriving community and a ***high-quality product*** (Shaikh & Cornford, 2009; Teixeira et al., 2016).

3.2.2 Problems of Vendor Involvement in Open Source Software Projects

Based on the literature we reviewed, we identified four key problems vendors encounter in OSS projects: ***domination of a single company***, ***conflict of interest***, ***losing competitive advantage***, and risk of ***developer attrition***.

The ***domination of a single company*** can lead to two potential issues in an OSS project. First, if a vendor company attempts to influence the development of an OSS project by imposing its own agenda without considering the expectations of other community members, it can negatively impact the community and trigger resistance within it. This approach may also threaten the project's continuity (Grand et al. 2004; Ägerfalk & Fitzgerald, 2008; Shaikh and Cornford, 2009; Schaarschmidt et al., 2011; Schaarschmidt & Stol, 2018). A balance must be maintained between the dominant company's priorities and the community's expectations (Ägerfalk & Fitzgerald, 2008). Furthermore, if a single company dominates the software development process but later reduces its involvement, this could endanger the project's sustainability (Zhang et al., 2022b).

Conflict of interest can arise in multi-vendor OSS projects when stakeholders have misaligned agendas and attempt to prioritize their own interests over the collective goals of the consortium. (Weikert et. al., 2019; Linåker & Regnell, 2020; Zhang et al., 2022a). This issue presents a risk for companies whose value propositions are closely tied to the OSS they are co-developing (Linåker & Regnell, 2020).

Weikert et al. (2019) propose a conflict prevention mechanism to mitigate the risk of conflicts of interest in these types of collaborations. They identify five key categories. The first category involves screening the member acceptance process, which includes assessing the motivations of potential members before admitting them and ensuring their alignment with the common interests of the existing community. The second category emphasizes the importance of established governance structures and rules. The third category focuses on promoting public communication and monitoring members' behavior. The fourth category highlights the common interests of the community. The final category stresses the need for openness, transparency, and the establishment of shared values.

One concern that companies face is the risk of ***losing their competitive advantage*** when contributing to OSS development (Grand et al., 2004; Linåker & Regnell, 2020).

Another problem for companies in multi-vendor OSS projects is ***risk of developer attrition***. Developers involved in OSS projects may choose to join other companies within the same ecosystem or start their own ventures (Teixeira et al., 2016; Schaarschmidt & Stol, 2018; Zhang et al., 2022a).

3.3 State of the Art of Success Factors in Company Collaboration Literature

Company-led OS consortia involve the collaboration of multiple companies. It is common for member companies in these consortia to operate within the same industry and compete with one another. However, they collectively contribute to the development of undifferentiated OSS projects.

Coopetition is a strategy that combines cooperation and competition (Brandenburger & Nalebuff, 1996), which is evident in the structure of company-led OS consortia. To understand the dynamics of successful collaboration within such consortia, we reviewed the literature on inter-company collaboration and coopetition. Specifically, we synthesized findings from seven literature reviews, focusing on the success factors of collaboration, coopetition, and strategic partnerships in Information Technology.

In this section, we present the common factors identified in at least three of these studies.

Collaborating with ***partners that have complementary expertise and strengths*** enhances the likelihood of a successful collaboration from the outset (Bruce et al., 1995; Hoffmann & Schlosser, 2001; Chin et al., 2008; Rikkiev & Mäkinen, 2009; Petter et al., 2014).

Ensuring ***mutual understanding and building trust*** are considered among the most important factors for a successful collaboration (Mattessich & Monsey, 1992; Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001; Chin et al., 2008; Rikkiev & Mäkinen, 2009; Petter et al., 2014).

Establishing ***common goals*** and defining ***concrete objectives*** agreed upon by all parties enhance the likelihood of a successful collaboration (Mattessich & Monsey, 1992; Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001; Chin et al., 2008; Rikkiev & Mäkinen, 2009; Petter et al., 2014).

Establishing ***clearly defined rules*** and outlining responsibilities and procedures at the initial stage of a collaboration contributes to mitigating conflicts in later phases (Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001).

Ensuring ***equality in resource sharing and decision-making*** within the community fosters stronger commitment to the collaboration (Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001, Rikkiev & Mäkinen, 2009; Petter et al., 2014). ***Periodic reviews*** enhance collaboration success by facilitating information sharing, enabling effective performance monitoring, minimizing conflicts, and allowing timely adjustments to the collaboration strategy (Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001; Chin et al., 2008).

Top management commitment enhances collaboration success, as their support ensures the allocation of sufficient resources (Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001, Rikkiev & Mäkinen, 2009).

The presence of ***mentors*** (or collaboration champions) who provide guidance and help navigate challenges strengthens the success potential of a collaboration (Bruce et al., 1995; Rai et al., 1996; Rikkiev & Mäkinen, 2009).

4

Methodology

To address our research questions, we followed a systematic literature review method in one study and case study methods among three studies. We qualitatively analysed the collected data by following thematic analysis and coding paradigm of grounded theory. In this chapter, we explain the details of each method in relation to our studies.

4.1 Research Method: Systematic Literature Review

A systematic literature review (SLR) is a form of secondary study, focusing on "identifying, evaluating and interpreting all available research relevant to a particular research question, or topic area, or phenomenon of interest" (Kitchenham, 2004).

The potential contributions of SLRs can be classified as "backward-oriented", involving the synthesis of existing knowledge or the aggregation of evidence from prior studies, and "forward-oriented", focusing on theory development or identifying gaps for future research (Diaz et al., 2024). Our research (P1) addresses both orientations. The first objective is to synthesize existing knowledge by reviewing literature on user-led OS consortia, while the second objective aims to develop a theory grounded in the information gathered.

We employed Kitchenham (2004) and Kitchenham & Brereton (2013) methodology in conducting our SLR research (P1) on user-led OS consortia. Kitchenham (2004) recommends conducting an SLR in three phases: planning the review, conducting the review, and reporting the results.

In the planning phase, we searched for existing SLRs on user-led OS consortia but did not identify any. As part of this phase, we also developed a literature review protocol that outlined our research objectives, the rationale behind the review, the search strategy, paper selection criteria, and data extraction methods. We followed this protocol. In the second phase, we conducted the review and documented our findings. Finally, in the reporting phase, we present the review process and its outcomes in our peer-reviewed paper (P1).

We provide the details of our methodology in the original paper in Appendix A. In this section, we provide a summary of the "conducting the review" phase.

4.1.1 Search Strategy

In the search strategy step, we defined the keywords, specified the timeframe, and selected the digital libraries for the search. We developed four sets of keyword lists, as shown in Table 4.1. These keywords were selected to compile a list of relevant papers published between 2000 and 2023. Our search was conducted across the electronic databases of Google Scholar, Web of Science, ACM Digital Library, IEEE Xplore, and Scopus.

4.1.2 Selection Process

To optimize our search process, we defined inclusion and exclusion criteria that are presented in Table 4.2 and developed a data extraction table to record and track all obtained results. The data extraction table is included in the External Appendix * of P1.

*Yenişen Yavuz, E., & Riehle, D. (2025). External appendixes: Why and How Do Organizations Create User-Led Open Source Consortia? A Systematic Literature Review.

Table 4.1: Sets of Keyword Lists

Sets	Definition	Keywords
Set 1	Terms which are used in the literature to define user-led OS consortia	"community source", "directed open source", "user-led open source consortia", and "user-led open source foundations"
Set 2	Terms which are used to define organizational involvement in open-source software development	"collaborative open-source software development", "intercompany OSS development", and "sponsored OSS development"
Set 3	Terms which present umbrella foundations	"Eclipse Foundation", and "Apereo Foundation"
Set 4	Terms which present known examples of user-led OS consortia or their projects	"Kuali", "Sakai", "openKonsequenz", "openMAMA", "Academy Software Foundation", "Nordic Institute for Interoperability Solutions" and "samvera"

We performed the selection process in three steps. We eliminated the results that do not meet the inclusion criteria, or that meet the exclusion criteria. In Figure 4.1, we present an overview of the paper search and selection steps, and results.

4.1.3 Quality Assessment

To assess the quality of the selected 54 papers, we examined the clarity of result reporting, the rigor of the studies, and the credibility of the findings. We adapted quality criteria from Dybå et al. (2007) and Kitchenham & Brereton (2013) to develop a quality assessment model

We began by understanding the overall structure of the papers, starting with questions to define research types and methods. We followed Kitchenham & Brereton (2013) by including research, discussion, and experience papers to capture both academic and practical perspectives. Different evaluation criteria were applied based on paper categories.

Our second set of questions focused on reporting quality, the third on rigor and trustworthiness, and the fourth on the credibility of findings. While Dybå et al. (2007) and Kitchen-

Table 4.2: Inclusion and Exclusion Criteria Applied in P1

Type	Criteria
Inclusion criteria	Publication year of the studies should be between 2000 and 2023
Inclusion criteria	Focus of the study should be about a user-led open source consortium, or a project, or overall model itself
Inclusion criteria	Type of the study should be either: empirical research papers; or discussion / opinion papers; or experience sharing papers of authors who are/were participants of any user-led OS consortia or project
Exclusion criteria	Studies which are not written in English
Exclusion criteria	Search results that are duplicates
Exclusion criteria	Search results that are not concurrent manuscripts such as conference agendas, journal announcements, interview scripts, lecture notes, presentations or editorials
Exclusion criteria	Student thesis (bachelor theses, master's theses, dissertations)

ham & Brereton (2013) primarily assessed quantitative studies, we added criteria for qualitative studies. To minimize subjective interpretation, we excluded assessments of method appropriateness and study relevance.

Regarding the results of our quality assessment model, we excluded the papers which were showing the following criteria:

- If any paper does not provide final results (findings) or provide results partly, it will be excluded
- If a research paper does not provide detailed information about sample AND data collection AND data analysis, it will be excluded
- If a research paper doesn't define their research method AND we couldn't define it either, it is excluded

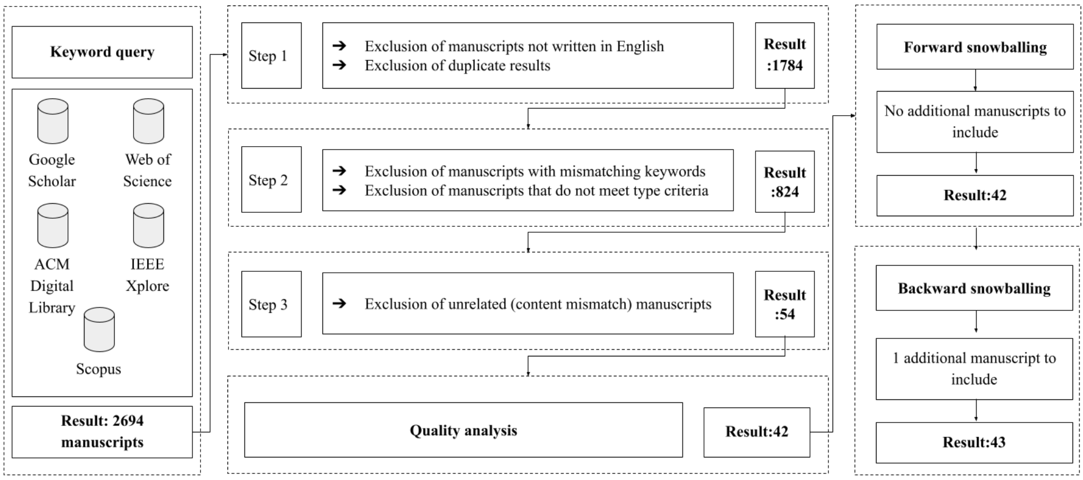

Figure 4.1: Search and Selection Process of the SLR

- If a research paper has a published extended version with almost the same content, the first version of the paper will be excluded
- If an experience sharing paper does not have an author involved in the sample project they discussed, it will be excluded

As a result, we excluded 12 papers in this step.

4.1.4 Snowballing

As the third step of our data collection process, we conducted forward (Felizardo et al., 2016) and backward snowballing (Wohlin, 2014) on the 42 papers that passed the quality assessment.

For the forward snowballing process, Google Scholar was used to gather papers citing the collected works. All papers were searched using the "cited by" function, except for Chesbrough & Appleyard (2007), which had 2364 citations. To refine this, Publish or Perish was used with keywords "open innovation and strategy" and "self-service." This process yielded 617 papers after duplicates were removed.

Exclusion criteria (duplicates, non-English papers, inaccessible papers, post-2023 publications, and irrelevant types) reduced this to 221 papers. After reviewing abstracts and full texts, three potentially relevant papers were identified, but none met the quality standards for inclusion.

For backward snowballing, references from the included papers were collected, resulting in 1223 references from 33 papers (9 papers lacked references). After exclusions, 740 papers remained for review. From this, one relevant paper was identified and added to the results.

4.2 Research Method: Case Study Research

We conducted two exploratory single-case case studies (P2 and P3), and one multiple-case case study (P4) following the guidelines of Yin (2018) and Eisenhardt (1989).

Our objective was to explore the motivations of organizations within user-led and vendor-led OS consortia, as well as to identify the current and potential problems faced by these consortia, along with the solutions and practices implemented to address them. To accomplish this, we examined real-world cases in their natural contexts, taking into account actual events and circumstances.

4.2.1 Case Selection

In our single-case case study research (P2 and P3), we employed purposeful sampling (Patton, 1990) by determining the relevant dimensions. For P2, we focused on the industry, maturity and activity level of the potential samples. We chose openMDM, a user-led OS consortium from the automotive industry as our sample. For P3, we focused on the size, focus, activity, and maturity level of the potential samples. We chose LF Edge, a vendor-led OS consortium focusing on development of an edge computing software stack.

In our multiple-case case study research (P4), we employed polar sampling to select two representative cases (Eisenhardt, 1989). Our first consideration was the type of foundation, choosing one vendor-led and one user-led OS consortia. For this research, we focused on the healthcare sector. Additional selection criteria included the consortia's size, scope, geographical activity area, focus, and maturity level. We chose openEHR as a sample of vendor-led OS consortium, and RACOON as a sample of user-led OS consortium.

Details of the case selection process for P2, P3, and P4 are presented in the original papers, in Appendix B, C, and D, respectively.

4.2.2 Background of Cases

openMDM is a user-led open-source consortium operating in the automotive industry. It is hosted by the Eclipse Foundation and officially recognized as an Eclipse Working Group. It was founded in 2014 by automobile companies Audi, BMW, and Daimler, along with service providers HighQSoft, Gigatronik, Canoo Engineering, Science+Computing, and Peak Solutions. Its primary goal is to develop and promote open-source tools for measurement data management based on ASAM ODS standards.

LF Edge is a vendor-led open-source consortia under the Linux Foundation that supports multiple Internet of Things (IoT) and edge device projects. The Linux Foundation provides LF Edge with administrative, technical, and legal services, such as establishing governance models, managing development infrastructure (e.g., code repositories), and supporting operations like event organization and community building. LF Edge was founded in 2019 with the participation of 60 companies. Its founding members include both emerging startups and well-established leaders in the software industry. LF Edge hosts a variety of projects (as of June 2023, 11 projects) with distinct use cases.

openEHR is a non-profit organization that offers technical specifications for Electronic Health Record (EHR) platforms and provides clinical models tailored to specific domains for defining content. Established in 2003, the openEHR Foundation is a globally recognized vendor-led open-source organization. Its Governing Board includes representatives from diverse membership categories, such as organizations, industry partners, and professionals, with a majority from vendor companies. While founded in England, openEHR operates internationally, including in countries like Australia and Canada. Its mission is to standardize Electronic Health Record (EHR) data, fostering interoperability and efficient healthcare data management.

The **RACOON** consortium is a user-led consortium, which was established in 2020 by the university clinics in Germany. The goal of the consortium is to promote collaboration among university clinics to enhance medical care for COVID-19 and cardiac diseases. RACOON is a user-led open-source consortium operating solely in Germany, with its governing board consisting of representatives from user organizations, specifically university clinics. Its main objective is to create a national system for multicenter analysis of radiological data.

4.2.3 Data Collection

As recommended by Yin (2018) and Guion (2011), we aimed data triangulation in all of three studies by using multiple sources of evidence. We used three main data sources: key informant interviews, meeting minutes of governing bodies, and official documents shared by the consortia.

Interviews with key informants provided insights into the consortia, revealing members' expectations and experiences. Meeting minutes provided an impartial perspective on the concerns and discussions taking place within the consortia. Official documents outlined the criteria for membership and project admission, along with details about the consortia' programs.

We performed semi-structured interviews with key informants. Semi-structured interviews involve open-ended, in-depth questions. While the interviewer prepares the questions beforehand, the sequence or content can be adjusted during the conversation. This flexibility enables the interview to progress naturally in a dialogue-like manner (Bryman, 2016). Before performing each of the interviews, we developed interview protocols that included an initial set of questions, the research topic, the case name, and the type of interview. We started each interview with introductory and transition questions to learn more about the interviewees and their organization. Following these initial questions, we proceeded with core questions focused on our research topics. The interviews concluded by giving the interviewees an opportunity to share any additional thoughts or comments. The sequence and phrasing of the questions were adapted as needed during the interviews.

Table 4.3 presents the summary of data collection details.

Table 4.3: Data Collection Details

Data Collection	Details for P2	Details for P3	Details for P4
Selected cases	openMDM - A user-led OS consortium (with a supporting case: Sakai)	LF Edge - A vendor-led OS consortium	openEHR - A vendor-led OS consortium RACOON - A user-led OS consortium
Data sources: Interviews	2 interviews with key informants of openMDM, 2 interviews with key informants of Sakai	4 interviews with key informants	7 interviews with key informants
Data sources: Documents	42 documents on OpenMDM, 36 documents on Sakai	128 documents	8 documents
Data collection and analysis period	November 2018 to March 2021	January 2023 to February 2024	May 2023 to August 2024

4.3 Qualitative Data Analysis: Thematic Analysis

To address our research questions in P1 and P4, we conducted qualitative data analysis following the six steps of the thematic analysis procedure proposed by Braun & Clarke (2006).

These steps are: (1) getting familiar with the data, (2) generating initial codes, (3) creating candidate themes, (4) reviewing themes, (5) defining and naming themes, and (6) producing a report.

In the first step of our thematic analysis for systematic literature review (P1), we familiarized ourselves with the data by taking notes on the content of each paper during the literature selection process. For the case study research (P4), we had interview transcripts. We reviewed these scripts to gain an overall understanding. In the second step, we started generating initial codes by using qualitative data analysis tools. For P1, we used MaxQDA and for P4, we used QDAcity[†].

In the third step, we refined our coding scheme by consolidating and clustering initial codes into sub-themes and main themes. We created a category for codes that did not fit into any of the themes or did not relate to our research questions, with intent to revisit them again.

[†] QDAcity.com

In the fourth step, we revisited each of the documents we were analyzing, carefully examining our codes and their associations with the emerging themes. We continuously revised and updated the codes and themes as necessary, eliminating any codes that did not fit into a category or were unrelated to our research questions.

In the fifth step, we had developed a clear set of themes and codes. We established precise definitions for the themes and incorporated the most relevant quotes corresponding to each code into the codebook.

In the last step as report production of our analysis, we presented our findings in the research papers, P1 and P4, which are presented in Appendix A and D, respectively.

4.4 Qualitative Data Analysis: Coding Paradigm of Grounded Theory (Open, Axial, Selective Coding)

The qualitative data analysis of our two research papers (P2 and P3) is based on the three phases of coding within **grounded theory** (GT), open, axial and selective coding (Strauss & Corbin, 1990).

During the **open coding** phase, we identified key events and actions to understand the consortia relevant to our study. In P2, our case was openMDM, and in P3, our case was LF Edge. While our focus was guided by research questions, we remained open to emerging relevant aspects. We began by creating conceptual labels and ensured that newly identified theoretical constructs were compared with previously gathered and analyzed data.

During the **axial coding** stage, we compared and grouped the events and actions identified in the open coding phase into subcategories and overarching categories. This process provided deeper insights that helped address our research questions. Relationships between concepts and categories were documented in code memos, which played a crucial role in our analysis.

During the **selective coding** stage, core categories were defined based on the research questions rather than emerging solely from the data, differing from a strict GT approach. All identified constructs were assessed for their relevance to these core categories and adjusted or removed as needed. The final core categories focused on (1) the consortium structure, challenges within the consortium, applied solutions, and success factors in the openMDM case, and (2) organizational engagement reasons, encountered problems, applied solutions, and practices in the LF Edge case. Categories from the axial coding phase represented the strategies, events, actions, and conditions linked to these core categories. This final categorization was essential in shaping the research outcomes and aligning the analysis with our research questions.

In both papers (P2 and P3), we employed data triangulation (Guion, 2011) by drawing on multiple data sources, including interviews, textual meeting minutes, meeting presentations, and official and corporate documents, to develop our theory. Throughout the analysis process, we created codebooks and continuously refined them as new codes emerged. These codebooks are included as supplementary documents in the appendix of each paper. Additionally, we ensured prolonged engagement in the research process.

5

Results

This section reports the results of the research questions addressed in this thesis. We present the answers of research questions by synthesising results of the four main research articles that form the foundation of this cumulative dissertation. In our studies, we assigned unique identifiers to our data sources and used these identifiers as sources of evidence in our results. To differentiate the identifiers for each research study, we prefixed them with labels P1, P2, P3, and P4, corresponding to the related study.

5.1 What are the Characteristics of Company-Led Open Source Consortia?

In this section, we present the characteristics of company-led open source consortia. We focused on the distinction of two consortia types: user-led open source consortia and vendor-led open source consortia. To identify the characteristics of user-led OS consortia, we performed an SLR and focused on the similarities of different user-led OS consortia. Based on our findings, we build our theory. The insights shared in section 5.1.1 were originally presented in:

P1: Yenişen Yavuz, E., Riehle, D. (2025) Why and How Do Organizations Create User-Led Open Source Consortia? A Systematic Literature Review. In Information and Software Technology, 107681.

To identify the characteristics of vendor-led OS consortia, we used the findings from our single-case case study on a vendor-led OS consortium-LF Edge-and our multiple-case case study on the similarities and differences between vendor-led and user-led OS consortia, which has the samples of openEHR (as a vendor-led OS consortium) and RACOON (as a user-led OS consortium). The insights shared in section 5.1.2 were originally presented in:

P3: Yenişen Yavuz, E., Riehle, D., Mehrotra, A. (2025). Why do companies create and how do they succeed with a vendor-led open source foundation. Empirical Software Engineering, 30(1), 1-49.

P4: Yenişen Yavuz, E., Shrivastava, A., Riehle, D., Putz, F. (2025). Governance Practices for Open Source Foundations in the Healthcare Sector. In: Papatheocharous, E., Farshidi, S., Jansen, S., Hyrynsalmi, S. (eds) Software Business. ICSOB 2024. LNBIP, vol 539. Springer, Cham.

The original papers are presented in Appendix A, C, and D.

5.1.1 The Defining Characteristics of User-Led Open Source Consortia

In P1, we conducted an SLR and examined 14 user-led OS consortia cases explored across 43 papers. Through our thematic analysis, we identified key characteristics of user-led OS focusing on consortium structure, membership structure, development process and the output. This section highlights the results of these research.

Consortium Structure

User-led OS consortia are composed of **multiple partner organizations** (P1-S6, P1-S11, P1-S17, P1-S18, P1-S19, P1-S25, P1-S26, P1-S32, P1-S35, P1-S37, P1-S39, P1-S40, P1-S42, P1-

S43). These organizations collaborate by aligning their efforts toward **a common, shared goal** and maintaining **a unified vision** (P1-S1, P1-S12, P1-S19, P1-S23, P1-S24, P1-S26, P1-S34, P1-S37, P1-S40, P1-S42, P1-S43). To support the long-term viability of projects, partners establish **non-profit legal entities**, such as consortia, foundations, or initiatives. The legal structure of these entities varies by country of origin, but their governance frameworks are largely consistent. They define the parameters and rules for collaborative work. A fundamental aspect of forming a consortium is the use of **formal contractual agreements** (P1-S14, P1-S25, P1-S28, P1-S32, P1-S41, P1-S43). These agreements, often structured as consortium charters, specify membership structures, as well as the **roles and responsibilities of member organizations**. Signing these agreements is mandatory for all partners to become part of the consortium (P1-S14, P1-S18, P1-S22, P1-S39, P1-S43).

User-led OS consortia are **initiated, financed, and steered by user organizations** whose core focus is not software development. Nevertheless, these organizations participate in software development activities to enhance and streamline their internal operations (P1-S2, P1-S5, P1-S6, P1-S8, P1-S11, P1-S16, P1-S18, P1-S19, P1-S25, P1-S34, P1-S38, P1-S42, P1-S43).

Membership Structure and Key Actors

In user-led OS consortia, different types of organizations in different roles are involved. As key actors, we defined five categories. These are: driver members, development partners (vendors), adopters (user members), non-profit members, and a legal entity such as a foundation.

Driver members, also referred to as driver organizations, are primarily **user organizations** that require software tailored to meet their specific internal business needs (P1-S8). The end users of this software are typically stakeholders within these organizations, who are general computer users rather than software developers (P1-S26). These driver members contribute to user-led OS consortia by **providing financial support** and/or **allocating staff resources** to software projects (P1-S18, P1-S21, P1-S22, P1-S33, P1-S34, P1-S35, P1-S42, P1-S43). They play a key role in **defining technical requirements** (P1-S11, P1-S15) and shaping **the direc-**

tion of software development (P1-S11, P1-S21, P1-S19, P1-S14, P1-S28, P1-S33).

Development partners, also referred to as development members, are **software vendors** or software suppliers actively participating in consortia (P1-S4, P1-S21, P1-S31, P1-S42, P1-S43). They contribute to the software development process by **implementing the requirements** defined by driver members (P1-S11, P1-S16, P1-S18, P1-S35, P1-S39, P1-S40). With their specialized expertise in software development, these partners help streamline the development process (P1-S3).

Development partners engage with user-led OS consortia by either paying a membership fee or contributing manpower to support development efforts. In return, they aim to offer **complementary fee-based services** to user organizations for software implementation (P1-S3, P1-S8, P1-S21, P1-S22, P1-S28, P1-S31, P1-S40). Participation in a user-led OS consortium also enhances their reputation as **trusted vendors** and allows them to **expand the reach of their technology** in the market (P1-S34).

Adopters, or user members, are user organizations that utilize the software developed by the consortium but **do not play an active role** in shaping the development process (P1-S11, P1-S34, P1-S35, P1-S43). While they sign contractual agreements like other members, their financial contributions are typically smaller compared to driver members. Their involvement often includes **providing feedback**, reporting bugs, or contributing additional functionalities developed to meet their own needs (P1-S11, P1-S35). The participation of these user organizations helps **increase the adoption** of the platform and supports the consortium in establishing industry standards (P1-S34).

Non-profit members, such as research organizations or universities, collaborate with the consortium without being user organizations themselves (P1-S11, P1-S43). Their involvement allows them to contribute to the project while also gaining access to valuable research data (P1-S34).

Foundations are legal entities that provide independent and neutral platforms for member organizations to collaborate and engage (P1-S13, P1-S16, P1-S39). As neutral platforms, foundations **safeguard the rights of members** and **manage the consortium's intellectual property** (IP) by holding its ownership (P1-S13, P1-S16, P1-S28, P1-S39, P1-S41). Members do not hold any special rights to the ownership of the software (P1-S40).

Members of user-led OS collaborations either establish their own foundations or become part of an existing umbrella foundation. To establish or join foundations, members are required to pay a membership fee, which is centrally collected and allocated to cover project expenses. This structure enables foundations to **ensure the financial sustainability** of user-led OS consortia by effectively pooling financial and staff resources (P1-S4, P1-S8, P1-S15, P1-S35).

In particular, umbrella foundations provide **governance, technical, and quality assurance support** to consortia. In terms of governance, they offer a clear framework for participation and contribution, and facilitate collaboration among member organizations (P1-S4, P1-S13, P1-S16). On the technical side, umbrella foundations supply technology platforms and shared development and testing infrastructures, enabling collaborators to work collectively (P1-S13, P1-S16, P1-S37). Additionally, foundations promote transparency in the requirements process, uphold quality assurance standards and help enhance both the quality and quantity of contributions to the project (P1-S13, P1-S35).

Foundations also play a vital role in **community development and management** by enhancing recognition and visibility. To attract prospective members, they organize conferences and meetings and provide marketing support (P1-S16, P1-S35, P1-S39). Foundations assist in selecting suitable members for the consortium, ensuring alignment with its culture and values (P1-S35).

They contribute to structuring the community and fostering a network of expertise by connecting developers and user organizations (P1-S4, P1-S8). By coordinating work, projects, and member activities, foundations help maintain cohesion within the community (P1-S4, P1-S8, P1-S35, P1-S37, P1-S39).

Development Process

User-led OS consortia operate as virtual organizations (P1-S14, P1-S17, P1-S18, P1-S19, P1-S20, P1-S22, P1-S24, P1-S28) and adopt distributed software development methodologies (P1-S8, P1-S18, P1-S28, P1-S33, P1-S39).

User organizations take the lead in the software development process. They are responsible for defining requirements and steering the overall development direction (P1-S2, P1-S6, P1-S11, P1-S14, P1-S16, P1-S19, P1-S21, P1-S24, P1-S25, P1-S31, P1-S38, P1-S43). While external funding may occasionally be an option, the development process is predominantly **financed by the member organizations** themselves (P1-S8, P1-S14, P1-S15, P1-S25, P1-S26, P1-S31, P1-S34, P1-S40). Member organizations collaborate on OSS development by **pooling their resources** (P1-S5, P1-S6, P1-S12, P1-S14, P1-S17, P1-S18, P1-S19, P1-S20, P1-S21, P1-S22, P1-S25, P1-S27, P1-S34, P1-S39, P1-S43). In user-led OS consortia, the majority of project participants are **employees of member organizations** (P1-S11, P1-S16, P1-S17, P1-S18, P1-S19, P1-S20, P1-S22, P1-S24, P1-S25, P1-S26, P1-S28, P1-S33, P1-S35, P1-S39, P1-S42, P1-S43). These employees may work for user organizations, development partners within the consortium, or both (P1-S11, P1-S16, P1-S18, P1-S25, P1-S26, P1-S35, P1-S39, P1-S40, P1-S42, P1-S43). Additionally, in some cases, project management and coordination are handled by paid staff employed by the legal entities, such as foundations (P1-S14, P1-S35, P1-S42, P1-S43).

Output

The outputs of user-led OS consortia projects are open-source software. However, many of these projects begin with an initial phase of closed-code development. User-led OS consortia can emerge from pre-existing closed-source software, as seen in the cases of openMAMA, openMDM, and openColorIO (P1-S16, P1-S37, P1-S43). Alternatively, partner organizations may collaborate to initially develop the software in a closed environment, as in the Sakai project (P1-S36). Once the software reaches a sufficient level of maturity, it is released as open-source software for use and contributions from other organizations (P1-S24, P1-S28, P1-S32, P1-S39).

Unlike software vendor companies, user organizations focus on creating **tailored enterprise applications** designed to address specific functionality gaps in their industries, rather than producing generic software (P1-S18). The resulting **software does not provide a competitive advantage or differentiation** for the user member organizations (P1-S3, P1-S6, P1-S9, P1-S16, P1-S21, P1-S34).

Open-source software (OSS) provides users with flexibility for **customization** (P1-S18, P1-S31, P1-S35), allowing user organizations to modify its functionality to meet their specific requirements. While the software is open to contributions from both individuals and organizations (P1-S16, P1-S24, P1-S31, P1-S32, P1-S37), most **contributions come from organizations** rather than individual volunteers (P1-S16, P1-S24, P1-S26, P1-S39).

5.1.2 The Defining Characteristics of Vendor-Led Open Source Consortia

In P3, we conducted a case study focusing on the vendor-led OS consortium of LF Edge. In order to understand the structure of the consortium, we investigated its membership structure, governance model, and software development process. In P4, we analyzed both a vendor-led and a user-led OS consortium to identify differences and similarities. This section presents the insights gained from these two research projects.

Consortium Structure

Vendor-led OS foundations primarily consist of **competing companies** that collaborate on undifferentiated software and **generic features** (P3-I1, P3-I3, P3-I4). It is essential that member organizations have **shared goals** and **a unified vision** (P3-I1, P3-I2, P3-I3, P3-I4).

Members operate within **established boundary rules**, which help mitigate conflicts and prevent unnecessary discussions within the foundation (P3-I2, P3-I4). For example, in the case of LF Edge, the foundation's charter document defines the membership rules and responsibilities (P3-LF-0). Each candidate must acknowledge their commitment to adhering to these rules and guidelines in order to become a member of the foundation (P3-I2, P3-I4).

In both the LF Edge and openEHR cases, the **governance structure consists of multiple layers**. The first layer is the governance board (as seen in the LF Edge) or the steering board (as seen in the openEHR). These are the main decision-making bodies and consist of members from different membership categories (P3-LF-0, P4-I4). Members vote on resource allocation and priority setting (P3-LF-0, P4-I1, P4-I3).

The second layer in the openEHR is the program boards with their independent governance mechanisms. For instance, as of 2023, in openEHR, there were four programs with governing boards: Specifications, Clinical Modeling, Software, and Education (P4-I4, P4-W1). Each program board represents a different area of expertise, sets priorities, and defines requirements for its own program (P4-I2).

In LF Edge, the second layer comprises the Outreach Committee, Legal Committee, Budget Committee, and Technical Advisory Council (TAC), all of which support the governance board. TAC acts as a link between the governing board and individual projects. TAC meetings provide a platform for members to discuss foundation-wide matters and project-specific topics (P3-L5). The third layer consists of project-specific technical steering committees. Each project operates independently, managing its own technical steering committee (P3-L2). They also have dedicated charters and the flexibility to modify their guidelines as necessary (P3-I2). Technical disputes within projects are addressed by their respective technical steering committees (P3-I4).

Membership Structure

LF Edge classifies its members regarding their financial contributions to the consortium. These categories are: (1) Premium, (2) General, and (3) Associate membership. Governance influence within the foundation is determined by membership level, with premium members having the highest level of influence (P3-LF-0).

Companies in the premium category have more influence on the governance direction of the consortia. Since these companies are primarily software vendors, we define these members as "driver members".

Driver members are **software vendors**. The specifications and the development direction of the software components are defined by these members. Vendors, which financially contribute more compared to other members, have more members in the governance board and power in the governance process of the consortium (P3-LF-0).

In the openEHR case, industry partners are software vendors that develop products or services based on openEHR specifications (P4-I3). They contribute financial and human resources to the ecosystem (P4-I1, P4-W4).

Organizational partners are **end-user companies or customers**. They offer insights into user expectations (P4-I1, P4-I3, P4-W4). In the case of openEHR, these partners are healthcare institutions. Conversely, LF Edge faces challenges due to the lack of end-user companies and their insights within the consortium (P3-I3).

Non-profit members, referred to as associate members in the LF Edge case, include universities, foundations, and non-profit organizations. While they are involved in the foundation, they do not provide financial contributions and have no rights to influence the consortium's governance (P3-LF-0).

Volunteers are **individuals** who contribute to projects on a voluntary basis. Members in vendor-led OS consortia seek to engage a diverse range of contributors to incorporate different perspectives (P3-I1, P3-I2, P3-B4). For example, in openEHR, there are two types of volunteer members: professional members, who provide consultancy and training services related to openEHR specifications, and individual members, who contribute to the development of these specifications (P4-I1, P4-I3). In the case of LF Edge, consortium members attract developers by offering various incentives, such as internship opportunities, to encourage participation in their projects (P3-I2, P3-L50, P3-L56, P3-LM-0).

Foundation is the **legal entity** responsible for owning and safeguarding the intellectual property of consortia (P4-I1, P4-W1). In the case of openEHR, two legal entities represent the consortium. In addition to the openEHR Foundation, a separate company, the openEHR Community Interest Company (CIC), has been established to ensure the foundation's financial continuity (P4-W3). Meanwhile, openEHR International manages the day-to-day operations within the openEHR community (P4-I1, P4-W1).

In the case of LF Edge, the LF Edge Foundation offers **governance and marketing support** to the projects it hosts, however, does not directly participate in the development process, nor does it influence the features and functionalities of the developed open-source software (P3-I4). Each project functions independently, guided by its own technical steering committees, objectives, and roadmaps (P3-L2, P3-L41).

Partnering with an umbrella foundation (in the LF Edge case, it is the Linux Foundation) that hosts multiple foundations helps companies by **simplifying the formation of new partnerships** and streamlining operations, enabling them to benefit from the experiences of other groups (P3-I2).

Development Process

Development of data models in openEHR is based on the efforts of both employees of member software vendors and individual volunteers (P4-I3). Developers contributing to OSS projects within the LF Edge ecosystem come from diverse backgrounds, such as company employees, independent volunteers, and students (P3-I2).

The opportunity of **collaborating with a diversity of organizations** and developers is one of the reasons companies are involved in vendor-led OS foundations (P3-I1, P3-I2, P3-I4). In this way, they gain insights and feedback from different perspectives, reach more adopters, and further reduce their development costs (P3-I1, P3-I2).

Output

Vendor-led OS foundations encompass competing companies as members to collaborate on **developing undifferentiated software and generic features**. Each company then builds its own unique proprietary components on top of the collaboratively developed OSS to support their individual business needs (P3-I1, P3-I3, P3-I4).

For instance, in the LF Edge case, vendors collaborate to facilitate harmonization across Edge projects by working on generic features (P3-L8, P3-L27). In the openEHR case, the focus is to develop data models and specifications (P4-I1). In openEHR, members collaboratively work on the data application layer, while industrial partners individually work on the application layer to offer projects compatible with openEHR specifications (P4-I1, P4-I3). Industry partners are not required to open source application-layer projects; they may choose to release them as OSS or proprietary software (P4-I1).

5.2 What are the Motivations of Companies to Create or Involve in Company-Led Open Source Consortia?

We address this question by synthesizing the findings we identified in our three studies: P1, P3, P4.

To identify the motivations of organizations to create or involve in user-led OS consortia, we conducted thematic analysis on the 43 papers we collected for our SLR research (P1). Furthermore, we synthesized our findings with the RACOON case which we investigated in our multiple-case case study research (P4).

The findings shared in section 5.2.1 was originally presented in:

P1: Yenişen Yavuz, E., Riehle, D. (2025) Why and How Do Organizations Create User-Led Open Source Consortia? A Systematic Literature Review. In Information and Software Technology, 107681.

P4: Yenişen Yavuz, E., Shrivastava, A., Riehle, D., Putz, F. (2025). Governance Practices for

Open Source Foundations in the Healthcare Sector. In: Papatheocharous, E., Farshidi, S., Jansen, S., Hyrynsalmi, S. (eds) Software Business. ICSOB 2024. LNBIP, vol 539. Springer, Cham.

To identify the motivations of vendors to create or involve in vendor-led OS consortia, we used findings from our single-case case study (P3) on a vendor-led OS consortium, LF Edge, and our multiple-case case study (P4) examining the similarities and differences between vendor-led and user-led OS consortia, which includes openEHR as a sample of a vendor-led OS consortium. The findings shared in section 5.1.2 was originally presented in:

P3: Yenişen Yavuz, E., Riehle, D., Mehrotra, A. (2025). Why do companies create and how do they succeed with a vendor-led open source foundation. Empirical Software Engineering, 30(1), 1-49.

PP4: Yenişen Yavuz, E., Shrivastava, A., Riehle, D., Putz, F. (2025). Governance Practices for Open Source Foundations in the Healthcare Sector. In: Papatheocharous, E., Farshidi, S., Jansen, S., Hyrynsalmi, S. (eds) Software Business. ICSOB 2024. LNBIP, vol 539. Springer, Cham.

The original papers are included in Appendix A, C, and D.

5.2.1 Motivations of Companies to Create or Involve in User-Led Open Source Consortia

Based on our literature review, we found that organizations primarily consider control, cost, sustainability, and productivity and innovation dimensions when deciding whether to develop or purchase software. In this section, we elaborate the factors related to each dimension.

Control Dimension

In user-led OS consortia, user organizations operate within the same industries and collaborate to address **shared needs and software functionality requirements** in their field (P1-

S2, P1-S6, P1-S12, P1-S15, P1-S25, P1-S34). Driver members, primarily composed of user organizations, have the authority to **define software functionality requirements**, set task priorities, and **steer the development direction** (P1-S2, P1-S14, P1-S17, P1-S19, P1-S20, P1-S21, P1-S25, P1-S28, P1-S33, P1-S43). With control over the development process, user organizations can prioritize their specific needs and drive the implementation of necessary functionalities (P1-S3, P1-S11, P1-S21, P1-S31, P1-S37).

Software developed by user-led OS consortia is open source and distributed under various OSS licenses selected by each consortium. This approach promotes **software adoption** among organizations with similar needs (P1-S2, P1-S31, P1-S35). As the number of users increases, the likelihood of **establishing industry standards** also grows (P1-S11, P1-S43).

Cost Dimension

User-led OS consortia enable organizations to **pool their** technical, human, and financial **resources** (P1-S6, P1-S12, P1-S14, P1-S17, P1-S20, P1-S25, P1-S39, P1-S43). This collaborative model helps organizations significantly **reduce software development costs** compared to acquiring proprietary solutions or building software internally (P1-S3, P1-S6, P1-S8, P1-S12, P1-S14, P1-S16, P1-S18, P1-S19, P1-S20, P1-S21, P1-S22, P1-S25, P1-S34, P1-S41). Furthermore, it **reduces ongoing maintenance and operational costs** (P1-S11, P1-S21, P1-S25, P1-S34).

As the final product is released as open-source software, some user-led OS consortia projects have the opportunity to secure **external funding** from various funding organizations (P1-S3, P1-S12, P1-S14, P1-S18, P1-S20, P1-S21, P1-S25, P1-S26, P1-S41, P1-S42).

In the higher education sector, some user-led OS consortia projects received early-stage funding from the Andrew W. Mellon Foundation (e.g., Sakai, Kuali) (P1-S18, P1-S20, P1-S41). To sustain development efforts, project participants later established legal entities, such as foundations, and began collecting membership fees from participating organizations. A similar funding model applies to the RACOON case. RACOON is funded by the Federal

Ministry of Education and Research Germany for a period of three years and, as of 2023, does not require the collection of membership fees (P4-J1).

Sustainability Dimension

User-led OS consortia projects have a **lower dependence on vendors** compared to proprietary software systems, as they are led and funded by user organizations (P1-S2, P1-S6, P1-S11, P1-S34). Providing the project outcomes as open-source software **expands the market to more vendors** (P1-S11, P1-S26, P1-S34, P1-S38) and **enhances the quality of support services** due to increased competition (P1-S2).

In user-led OS consortia, the software is not owned by a vendor or any single driver organization. Instead, **intellectual property (IP)** ownership typically resides with legal entities, such as foundations, that represent the consortia (P1-S28, P1-S39, P1-S40). These entities serve as **neutral** forums, ensuring project independence and reliability.

Another key factor contributing to the sustainability of user-led OS consortia is the **commitment of member organizations** to the consortia and their projects. Organizations enter formal agreements to become members, pledging to participate in the project for a defined period. They also commit to regularly investing resources, whether through human capital or financial contributions, thereby strengthening the long-term sustainability of the projects (P1-S6, P1-S14, P1-S21, P1-S24, P1-S43).

Productivity and Innovation Dimension

Since member organizations in user-led OS consortia strive for the same functionality in the final product, they adhere to a stronger product vision (P1-S12). This enables them to focus on the continuous **enhancement of software functionality and quality** (P1-S6, P1-S11, P1-S16, P1-S24, P1-S38, P1-S43).

Member organizations and their employees actively **share knowledge and experience** related to the projects (P1-S14, P1-S25, P1-S33). This collaborative approach **fosters innova-**

tion within projects (P1-S2, P1-S3, P1-S6, P1-S14, P1-S16, P1-S17, P1-S19, P1-S22, P1-S31, P1-S37, P1-S38, P1-S39). Additionally, collaboration supports the **development of expertise** and contributes to the professional development of staff (P1-S8, P1-S33, P1-S39).

For instance, in the RACOON case, the primary focus is on developing a platform that facilitates data sharing and analysis among radiology departments in university hospitals. The goal is to enhance the understanding of COVID-19 and improve patient care (P4-J1, P4-J2). Beyond COVID-19, the scope of data sharing has expanded to include medical imaging data related to cancer and cardiac conditions. One significant use case involves utilizing the collected data to train artificial intelligence and machine learning models, improving pattern recognition in Computed Tomography (CT) scans (P4-J2, P4-V4). Collaboration has helped expand research opportunities and accelerate innovation in the field.

Since the work result is OSS, user-led OS consortia benefit from the **contributions of the community** (P1-S1, P1-S6, P1-S37, P1-S39, P1-S35). These contributions can be in the form of innovative ideas (P1-S2, P1-S9, P1-S14, P1-S39), expert or technical issues (P1-S37, P1-S39), collaboration on future research and development (P1-S1), improvement suggestions and bug fixes (P1-S1, P1-S35).

In the RACOON case, the platform facilitates communication and strengthens networking among university hospitals. Researchers can suggest investigation topics and, once approved, use the platform for their studies (P4-J2). Its transparency policy allows interested parties to take part in proposed research projects and attract contributions to projects (P4-J1). Overall, the platform strengthens collaboration among university clinics, fostering a more integrated research environment (P4-J1, P4-J2, P4-J3).

Working on OSS development has a positive influence on the **developer motivation** in user-led OS consortia projects (P1-33, P1-34, P1-S35). For instance, Samuel et al. (2022) highlights that working collaboratively with other organizations on the Kuali Rice project motivates developers to help each other more, as developers want both their organizations and themselves to gain a good reputation for their expertise in software development and contributions to the project (P1-S33).

Table 5.1 summarizes our findings.

Table 5.1: Motives of User Organizations for Participating in User-Led Open Source Consortia

Dimension	Motive	Sources of evidence (Literature ID)
Control	Steer the development direction and control the functionality	P1-S2, P1-S14, P1-S17, P1-S19, P1-S20, P1-S21, P1-S25, P1-S28, P1-S33, P1-S43
Control	Fulfillment of user requirements in the field	P1-S2, P1-S6, P1-S12, P1-S15, P1-S25, P1-S34
Control	Increased adoption of the software among organizations with similar needs	P1-S2, P1-S31, P1-S35
Control	Establishing industry standards	P1-S11, P1-S43
Cost	Pooling resources	S6, S12, S14, S17, S20, S25, S39, S43
Cost	Reducing software development costs	P1-S3, P1-S6, P1-S8, P1-S12, P1-S14, P1-S16, P1-S18, P1-S19, P1-S20, P1-S21, P1-S22, P1-S25, P1-S34, P1-S41
Cost	Reducing ongoing maintenance costs	P1-S11, P1-S21, P1-S25, P1-S34

Dimension	Motive	Sources of evidence (Literature ID)
Cost	External funding opportunity	P1-S3, P1-S12, P1-S14, P1-S18, P1-S20, P1-S21, P1-S25, P1-S26, P1-S41, P1-S42, P4-J1
Sustainability	Reducing vendor dependency	P1-S2, P1-S6, P1-S11, P1-S34
Sustainability	Extended market of commercial affiliates/vendors	P1-S11, P1-S26, P1-S34, P1-S38
Sustainability	Increase in the quality of support	P1-S2
Sustainability	Neutral IP ownership	P1-S28, P1-S39, P1-S40
Sustainability	Commitment of member organizations	P1-S6, P1-S14, P1-S21, P1-S24, P1-S43
Productivity & Innovation	Enhancement of software functionality and quality	P1-S6, P1-S11, P1-S12, P1-S16, P1-S24, P1-S38, P1-S43
Productivity & Innovation	Sharing knowledge and experience	P1-S14, P1-S25, P1-S33, P4-J1, P4-J2
Productivity & Innovation	Fostering innovation	P1-S2, P1-S3, P1-S6, P1-S14, P1-S16, P1-S17, P1-S19, P1-S22, P1-S31, P1-S37, P1-S38, P1-S39, P4-J1, P4-J2, P4-V4
Productivity & Innovation	Supporting staff development	P1-S8, P1-S33, P1-S39

Dimension	Motive	Sources of evidence (Literature ID)
Productivity & Innovation	Continuous improvement via community contributions	P1-S1, P1-S2, P1-S6, P1-S9, P1-S14, P1-S35, P1-S37, P1-S39, P4-J1, P4-J2, P4-J3
Productivity & Innovation	Increase in developer motivation	P1-33, P1-34, P1-S35

5.2.2 Motivations of Companies to Create or Involve in Vendor-Led Open Source Consortia

Based on our research on LF Edge consortium and openEHR, we identified 18 motives across four dimensions for vendor companies' engagement in vendor-led open-source consortia. These dimensions are: revenue, competition, productivity and innovation, and reputation.

The revenue dimension includes motives related to the financial gains of companies. Motives in the competition category focus on how companies' engagement influences their competitive strength and relationships. The productivity and innovation dimension encompasses motives related to improvements in work processes and outcomes achieved through collaborative efforts. Finally, the reputation dimension focuses on motives that provide companies with opportunities to gain recognition from others. Table 5.2 summarizes our findings.

Revenue Dimension

In LF Edge, vendors collaborate on building open-source software (OSS) projects by focusing on shared, non-competitive, and general-purpose features (P3-I1, P3-I3). By engaging with external developers, companies can **reduce their development costs** and **benefit from collaborative efforts** (P3-I1).

Individually, companies create unique proprietary components tailored to their business needs on top of these shared OSS developments. This strategy enables vendors to **generate**

revenue from products that incorporate OSS elements (P3-I1, P3-I3, P3-I4).

Additionally, collectively establishing open standards with other organizations helps **prevent vendor lock-in** (P3-I1, P3-I2, P3-B4). This approach enhances flexibility for companies when selecting maintainers for their products. Vendors also seek economic benefits, such as **reduced development and maintenance costs** (P3-I1, P3- I2).

Preventing vendor lock-in is also a motive observed in the openEHR case. Industry partners develop application layers based on openEHR specifications, promoting vendor neutrality. This approach enables organizations to choose vendors based on their specific needs (P4-I1, P4-I2). These partners also provide commercial services leveraging openEHR specifications, helping organizations **save costs** by utilizing existing data models (P4-I1).

Competition Dimension

Companies focus on **establishing open standards** (P3-I2, P3-B3) and seek to leverage their contributions to OSS projects to **influence the market** and establish themselves as industry leaders (P3-I1, P3-I4, P3-L43).

Another driving factor for participation in vendor-led OS foundations is **peer pressure** (P3-I1, P3-I3, P3-I4). When companies observe their competitors actively engaging in OSS projects and consortia, they feel compelled to join these initiatives to maintain their competitive advantage.

Productivity and Innovation Dimension

One key reason companies join OS foundations is the opportunity to **collaborate with a wide diversity of organizations** (P3-I1, P3-I2, P3-I4). This collaboration allows them to **gain valuable insights from diverse perspectives** (P3-I1, P3-I2) and expand their **reach to a larger base of adopters** (P3-I2). Additionally, the diversity of member organizations helps prevent the consortium from being dominated by one or two powerful entities (P3-I2).

Vendor-led OS foundations bring together competitive companies that work collectively on non-differentiated software and generic features (P3-I1, P3-I3, P3-I4). It is crucial for member organizations to share common goals and a unified vision. By aligning their objectives, they can focus on **addressing shared common problems** and collaborating with other members to **develop effective solutions** (P3-I1, P3-I2, P3-I3, P3-I4). These companies can also adopt common frameworks to ensure interoperability and streamline testing processes (P3-I1). For example, a common challenge in the healthcare sector is the diversity of data structures. Two main motives for organizations to engage with openEHR are the need for **standardized data structures** and the goal of **ensuring semantic interoperability** across healthcare systems (P4-I1, P4-I2, P4-I3, P4-I4). Through joint problem-solving efforts and the use of open standards, companies can **save time and resources** while **increasing the pace of innovation** across the industry (P3-I1, P3-I2, P3-I3, P3-L43).

Participation in vendor-led OSS projects provides companies with **access to a global pool of talent** (P3-I1, P3-I2, P3-B4). Contributors to OSS projects come from diverse backgrounds, including employees from various organizations, volunteer developers, and students (P3-I2). This diversity fosters broader feedback and support from a larger community (P3-I1, P3-I2). Moreover, the open nature of the source code allows for easier detection of potential malicious code, enhancing the **security of the code and projects** (P3-I1).

Additionally, collaborating with an established umbrella foundation that hosts multiple consortia is appealing to companies. It simplifies the process of **establishing new partnerships** and streamlines operations by allowing them to benefit from the experiences of other groups (P3-I2).

Reputation Dimension

By becoming members of an OS consortium, companies can leverage the consortium's **outreach opportunities** (P3-I1, P3-I4). For example, they can present their projects at events and conferences (P3-I4). Startup companies, in particular, can **gain recognition** by joining a consortium, as it enables them to attract the attention of other member organizations (P3-I4).

These interactions facilitate the formation of collaborative relationships in conjunction with open-source projects (P3-I4).

Investment security is another key motivator, especially for public institutions. In the case of openEHR specifications, which are open-source, organizations can continue development even if openEHR ceases to operate, ensuring the protection of their investments (P4-I2, P4-I4). From an ethical perspective, especially when public funds are involved, investing in open-source solutions benefits the broader community (P4-I4).

Table 5.2: Motives of Vendors for Participating in Vendor-Led Open Source Consortia

Dimension	**Motive**	**Sources of evidence**	**References in literature that support our findings, if any**
Revenue	Generating revenue from the product built using the OSS components	P3-I1, P3-I3, P3-I4	Zhang et al., 2020
Revenue	Preventing vendor lock-in	P3-I1, P3-I2, P3-B4, P4-I1, P4-I2	Ägerfalk & Fitzgerald, 2008; Linåker & Regnell, 2020
Revenue	Reducing development and maintenance costs	P3-I1, P3-I3, P4-I1	Shaikh and Cornford, 2009; Riehle, 2010; Teixeira et al., 2016
Competition	Establishing open standards	P3-I2, P3-B3	West & Gallagher, 2006; Riehle, 2010; Schaarschmidt et al., 2011; Teixeira et al., 2016; Linåker & Regnell, 2020

Dimension	Motive	Sources of evidence	References in literature that support our findings, if any
Competition	Influencing the market and becoming the leader	P3-I1, P3-I4, P3-L43	Linåker & Regnell, 2020
Competition	Experiencing peer pressure	P3-I1, P3-I3, P3-I4	West & Gallagher, 2006
Productivity & Innovation	Collaborating with a diversity of organizations	P3-I1, P3-I2, P3-I4	
Productivity & Innovation	Gaining insights from diverse viewpoints (from members)	P3-I1, P3-I2	Shaikh and Cornford, 2009; Teixeira et al., 2016; Linåker & Regnell, 2020
Productivity & Innovation	Reaching a diversity of adopters	P3-I2	Grand et al. 2004; West & Gallagher, 2006; Joo et al., 2012
Productivity & Innovation	Solving common problems by focusing on generic features	P3-I1, P3-I2, P3-I3, P3-I4, P4-I1, P4-I2, P4-I3, P4-I4	West & Gallagher, 2006; Schaarschmidt et al., 2011; Teixeira et al., 2016
Productivity & Innovation	Increasing the pace of innovation in the industry	P3-I1, P3-I2, P3-I3, P3-L43	Linåker & Regnell, 2020

Dimension	Motive	Sources of evidence	References in literature that support our findings, if any
Productivity & Innovation	Reaching talent globally	P3-I1, P3-I2, P3-B4	Grand et al. 2004; West & Gallagher, 2006; Ägerfalk & Fitzgerald, 2008; Linåker & Regnell, 2020
Productivity & Innovation	Receiving feedback and support of a large community	P3-I2	Grand et al. 2004; Iivari et al., 2008; Shaikh and Cornford, 2009; Teixeira et al., 2016; Linåker & Regnell, 2020
Productivity & Innovation	Having more secure code	P3-I1	
Productivity & Innovation	Reduced complexity of establishing partnerships	P3-I2	Hunter & Walli, 2013; Eckert et al., 2019; Izquierdo und Cabot, 2020; Yenişen Yavuz et al., 2022
Productivity & Innovation	Learning from other groups' experiences	P3-I2	Izquierdo und Cabot, 2020
Reputation	Increasing outreach opportunities	P3-I1, P3-I4	
Reputation	Gaining recognition among other companies	P3-I4	

5.3 What are the Good Practices for Successful Company-Led Open Source Consortia?

In order to identify the practices that contribute to the success of company-led open source consortia, we examined the problems they face and the solutions used to overcome them. Furthermore, we identified the good practices that these consortia apply for successful collaboration.

In this section, we synthesize the findings from the three case studies (P2, P3, and P4) we conducted to address this research question. These studies are presented in Appendices B, C, and D, respectively.

In P2, we investigated the problems, applied solutions, and success factors in a user-led open-source (OS) consortium, **openMDM**. In P3, we examined the problems, applied solutions, and success factors in a vendor-led OS consortium, **LF Edge**. We synthesized these findings with the results from P4, which focuses on the **RACOON** consortium as a user-led OS consortium and the **openEHR** consortium as a vendor-led OS consortium.

We identified **90** practices in **26** contexts that emerged in the cases of the openMDM, the LF Edge, openEHR, and/or RACOON. Contexts consist of problems and applied solutions (practices).

We categorized these problems and solutions in three dimensions. These dimensions are: governance, efficiency and productivity, and sustainability. We explain the implemented (or proposed) solutions to address both experienced and potential problems.

We assigned identifiers (ID) to each of these contexts, problems, and solutions. These IDs are presented with the abbreviations such as C1.P and C1.S, in which "C" stands for Context, "P" stands for Problem, and "S" stands for Solution.

We define each context among their dimensions in this section. Table 5.3, 5.4, and 5.5 present the list of practices regarding their dimensions.

5.3.1 Governance Dimension

Context C1. Mitigating Management Conflicts

Problem C1.P: Company-led OS consortia consist of multiple organizations with diverse structures. For instance, LF Edge was founded in 2019 with 60 initial member organizations, representing a diverse mix of large enterprises and small start-ups (P3-A1, P3-I2). This diversity brings varying expectations and needs, which can sometimes lead to tensions within the foundation (P3-I2).

Solution C1.S: LF Edge operates under a set of bylaws that all members must follow, outlined in its charter document. Membership requires candidates to acknowledge and adhere to these ***established bounding rules***, which help mitigate conflicts and avoid unnecessary discussions within the foundation (P3-I2, P3-I4).

The same approach is applied in the openMDM case. The consortium's objectives, regulations, member roles, responsibilities, privileges, and governance structure are explicitly outlined in the charter (P2-L1). To participate in the collaboration, organizations must agree to these predefined rules. Establishing clear guidelines and boundaries from the outset is crucial for preventing potential conflicts and ensuring the long-term sustainability of the collaboration (P2-I1).

Separating governance and technical responsibilities is another practice. LF Edge separates the responsibilities of its Governing Board from those of its technical committees. The Governing Board, comprising representatives from various membership categories, addresses foundation-wide issues (P3-LF-0, P3-I4). In contrast, projects within LF Edge maintain autonomy, with their own technical steering committees to resolve technical conflicts and update project-specific rules as needed (P3-I2, P3-I4). A further governance organ, the Technical Advisory Council (TAC), acts as a bridge between the Governing Board and the projects (P3-L5).

A further practice is ***managing meetings professionally and ensuring transparency.*** LF Edge TAC holds periodic meetings (weekly or biweekly) with its members. These meetings are facilitated by professional program managers who guide participants in identifying solutions and fostering productive discussions. After each session, the meeting minutes and recordings are made publicly available. This approach ensures transparency and accessibility (P3-I1, P3-I2). Holding periodic meetings and publishing meeting minutes are also practices applied in the openMDM and RACOON cases (P2-I2, P4-J3).

Context C2. Mitigating the risk of a single company's dominance

Problem C2.P: The domination of a single company in a consortium would negatively affect collaboration efforts. For instance, in the openEHR case, the dominance of one of the founding companies (Ocean Informatics) created a negative perception within the community. Some community members felt that they were primarily serving Ocean Informatics' business interests rather than contributing to an independent initiative, which negatively impacted the community dynamic (P4-I1).

In the LF Edge case, the Governing Board oversees foundation decisions and project statuses, with voting rights determined by membership category. Each premium member is granted one vote (P3-LF-0). A dispute arose when IBM, a premium member, acquired another premium member, Red Hat, resulting in a single organization holding two votes. This situation led to disagreements among members (P3-I2, P3-I4).

Solutions C2.S: In the openEHR case, the issue is addressed by restructuring the governance system, with ***representatives from all membership categories*** appointed to the governing board. This approach ensures that a wide range of perspectives is reflected within the foundation (P4-I1, P4-I3, P4-I4). The updated structure prioritizes transparency and adopts a democratic decision-making process (P4-I1).

The governing board determines ***resource allocations through a voting system*** among its members (P4-I3, P4-I4). ***Including representatives from competing companies*** helps prevent any single organization from exerting excessive influence (P4-I1). Additionally, the practice of ***rotating board members*** prevents prolonged dominance by the same individuals and fosters a more diverse range of viewpoints (P4-I4).

In the LF Edge case, the issue is addressed by collaboratively ***reaching a compromise***: an acquiring company and its subsidiary may each retain voting rights only if the parent company owns less than 50% of the subsidiary's shares (P3-I2, P3-LF-0). This rule applies to all similar cases within LF Edge. The decision was documented, and the charter was updated to include guidelines on subsidiary voting rights (P3-L30, P3-LF-0).

The user-led openMDM consortium mitigates this risk by adopting a strategy of ***shared resources and equality***. Driver members have equal rights in terms of resource allocation and influence over project decisions (P2-L1, P2-I1). Each driver member holds a seat on the Steering Committee (SC) and is granted three voting rights (P2-L2).

Similar to the openEHR case, strategic decisions in openMDM are approved by majority vote, often reaching unanimous consensus (e.g., P2-M2, P2-M6, P2-M9, P2-M11).

Context C3. Mitigating concerns about the influence of political factors on decision-making

Problems C3.P: In LF Edge, candidate projects are evaluated by TAC members. TAC also determines the lifecycle stages of each project (P3-L12). The allocation of funding to hosted projects is linked to their lifecycle stages, making project maturity a key factor (P3-L6, P3-LO-1, P3-LP-5). In some cases, concerns have been raised about perceived unfairness in the evaluation process within LF Edge (P3-I3).

Solutions C3.S: To address concerns about fairness, LF Edge emphasizes applying the ***same set of evaluation criteria*** across all projects, reducing potential conflicts of interest among member organizations (P3-L6).

In the case of a disagreement regarding a project's lifecycle evaluation, the issue is addressed by forming ***a subgroup to investigate the project's application in depth*** and sharing the results with TAC members (P3-L13). This approach helps highlight the project's strengths and areas for improvement, facilitating its further advancement (P3-L16). ***Maintaining transparency in decision-making*** is prioritized to foster positive member relationships (P3-I3).

Context C4. Developing Guidelines and Reference Documents

Problem C4.P: Although company-led OS consortia have established management rules, they may still encounter unforeseen issues that were not anticipated or addressed in the existing documentation. This situation has been observed in LF Edge (P3-L28).

On the other hand, a lack of documentation may lead to knowledge loss within consortia. For instance, in the openMDM case, turnover among some service providers negatively impacted the development process of their software project, as it resulted in knowledge loss during the code integration process (P4-M1, P4-M49).

Solution C4.S: In LF Edge, TAC members collaborate to ***develop guidelines and reference documents*** for consistent future use, such as project proposal instructions, project induction guidelines, the Project Lifecycle Document (PLD), and API documentation standards (P3-L03, P3-L11, P3-L13, P3-L29, P3-L31). The consortium ***updates these documents as needed***, incorporating lessons learned and adapting to evolving processes (P3-I2, P3-L45). One example of such documentation is the **"Open Glossary of Edge Computing"** by LF Edge. This document aims to standardize terminology, ensuring consistent vocabulary use across documents and enhancing communication efficiency (P3-L51).

To prevent future issues, the openMDM consortium prepares documentation for different audiences. These documents include ***guidelines, specifications, release notes, and process plans***, aiming to provide technical information and guidance on avoiding recurring problems (P2-I1, P2-I2). This approach helps mitigate knowledge loss and increase user engagement.

Context C5. Mitigating legal conflicts among members

Problem C5.P: In user-led and vendor-led OS consortia, members operate within the same industry. Often, they collaborate with their direct competitors in the field (P3-I1, P2-I1).

Solutions C5.S: Due to the competitive relationships among many consortia members, it is crucial for them to ***adhere to antitrust policies*** to proactively avoid potential legal issues (P3-I1). These policies advise members to refrain from discussing topics such as products, market differentiation, and pricing (P3-I1). To ***comply with antitrust laws***, publishing meeting minutes, sharing the consortium's common activities with the public, and ensuring transparency are important practices (P2-I1).

Context C6. Finding a balance among members' expectations

Problem C6.P: User-led and vendor-led OS consortia includes a diverse range of organizations. For example, LF Edge consists of both small startups and major industry players (P3-I4). This diversity brings advantages, such as greater resource contributions from larger companies and faster adaptability from smaller startups (P3-I4). However, differing expectations and needs can lead to disagreements, particularly in budget decision-making (P3-I2, P3-I3, P3-I4). Members contributing more resources to the foundation tend to have greater influence, requiring project members to persuade decision-makers to gain approval for their requests (P3-I3).

Solutions C6.S: The solutions applied in the LF Edge case include ***maintaining transparency about concerns and requests***, effectively ***communicating and convincing on the benefits of proposed changes***, and ***building positive relationships among members*** (P3-I3). Additionally, ***creating a balanced representation of startups and large corporations*** (e.g.,50% each) could help maintain equilibrium within the consortium (P3-I4).

Context C7. Facilitating mediation among diverse expert opinions

Problem C7.P: In the openEHR and RACOON cases, member organizations consist of experts with unique visions, perspectives, and experiences. Differing ideas can complicate decision-making, as members may have varying functional expectations, technical preferences, and approaches, making it challenging to reach a consensus (P4-I1, P4-J1).

Solutions C7.S: One of the practices applied in the RACOON case is ***recognizing and appreciating the expertise and abilities*** of all individuals involved (P4-J2). The next step is to ***facilitate open dialogue***, ensuring that everyone has the opportunity to share their perspectives (P4-J1, P4-J2, P4-J3).

Mutual respect and understanding play a key role in bridging diverse interests and viewpoints, fostering a collaborative environment and an inclusive spirit that aligns with the organization's objectives (P4-I1, P4-I3, P4-J1, P4-J2, P4-J3).

Context C8. Fostering dialogue and trust among members

Problem C8.P: Company-led OS consortia comprise a diverse range of member companies, bringing together individuals with varying perspectives and viewpoints. As a result, disagreements within the consortia are possible (P3-I2).

Solution C8.S: In-person communication plays a crucial role in fostering trust and strengthening relationships among members (P4-I1, P4-J3). Hosting ***face-to-face meetings*** allows members from different organizations to connect and build a sense of community (P3-I1, P2-I2, P4-I1).

For instance, RACOON hosts sessions at conventions and conferences, promoting communication and helping members establish personal connections (P4-J3). Likewise, openEHR members benefit from in-person meetings, which assist in resolving disagreements (P4-I1). In the LF Edge case, members of different project teams (e.g. the Akraino Team and EdgeX) held in-person meetings to explore collaboration opportunities (P3-L30). In the openMDM case, the entire community meets at the annual meeting, held once a year.

openMDM organizes ***hackathons*** and ***developer workshops***, providing a platform for developers to build trust, share experiences, present best practices, and discuss project requirements and solutions (P2-I2, P2-M5, P2-M28, P2-M43). These events foster trust and mutual understanding among team members and developers, positively impacting the collaborative work

Regular meetings and periodic communication improve information flow and facilitate direct interaction among members (P2-I2, P4-J3). For instance, openMDM committee members hold regular meetings (P2-A5, P2-M2). Discussions cover specifications, technology decisions, job assignments, and project status. To ensure ***transparency***, meeting minutes and assignments are shared on the community's wiki and the consortium's mailing list (P2-M2, P2-M21). Regular communication fosters trust and understanding among team members and developers.

Additionally, as observed in the RACOON case, an ***online platform*** which enables members to communicate with each other supports continuous member engagement (P4-J3). ***Providing an open environment*** where everyone can freely share their opinions is essential for exposing members to diverse perspectives (P3-I2). ***Maintaining transparency*** further strengthens trust (P4-I1, P4-I4).

Context C9. Providing an inclusive and open environment

Problem C9.P: To foster the dialogue among members and encourage their engagement in the community, it is essential to provide an inclusive and open environment (P3-I2, P4-I2).

Solutions C9.S: One practice to provide an inclusive and open environment for new members is ***sharing comprehensive information*** on governance structure and regulations through resources such as onboarding protocols, the website, and consortium wiki pages (P4-J1). ***A well-structured onboarding process*** that clearly outlines regulations can help simplify complexities for newcomers (P4-J1).

Maintaining transparency about projects and processes, along with openly sharing information with members, is crucial for building trust, instilling confidence, and encouraging active participation (P4-J1).

Showing appreciation to people who contributed to the success and providing opportunities for them to speak at conferences are additional strategies that motivate engagement in projects (P4-I3).

Furthermore, ***explaining the impact of projects*** and sharing experiences within the community effectively attract new contributors (P4-I3). In an open environment, even competing vendors can exchange experiences and knowledge, provided they uphold the shared goal of ***maintaining open standards*** (P4-I2).

Context C10. Ensuring transparency

Problem C10: In collaborative efforts toward a shared objective, maintaining openness and transparency among members is crucial (P2-I1, P4-J3). Transparency ensures that decisions are not made in isolation, information is communicated clearly, and every member has access to the necessary details for meaningful contribution (P4-I4, P4-J2).

Solutions C10: To ensure transparency, openMDM and openEHR ***documents its decisions*** and k***eeps the community informed about the roadmap and resource allocation*** (P4-I3, P4-I4). For example, openEHR holds annual meetings to update members on the foundation's progress and status (P4-I2).

Additionally, ***publishing meeting minutes*** and sharing relevant details—such as responsibilities, assignments, bug reports, and achievements—enhances information flow, as observed in the cases of openMDM, openEHR, and RACOON (P4-I4, P4-J3, P2-I1, P2-I2, P2-M2, P2-M4).

Further initiatives include ***providing an open environment for community dialogue*** with board members (P4-I3) and allowing interested members to attend subprogram board meetings (P4-I3, P4-J3).

In the openMDM case, steering and architecture committee meetings are open for anyone to attend, and meeting minutes are generally open to the public. Information is shared with both the public and project participants through community wiki pages and mailing lists (P2-A1, P2-M6, P2-M14, P2-M20, P2-M21).

5.3.2 Efficiency and Productivity Dimension

Context C11. Competition among overlapping projects

Problem C11.P: When company-led OS consortia host multiple projects, competition for resource allocation can arise among them. This issue has been observed in LF Edge. As of June 2023, LF Edge was hosting 11 projects, some of which addressed similar use cases (P3-L15). While the consortium allowed different projects to develop similar solutions to foster competition and innovation, this approach also led to resource inefficiencies and created tensions within the foundation (P3-I2, P3-I3).

Solutions C11.S: LF Edge addressed this issue by focusing on delivering value to adopters through ***encouraging collaboration among projects***, developing complementary solutions and distinct offerings (P3-I2, P3-L45). To clarify project similarities and differences, LF Edge TAC members ***developed a taxonomy*** (P3-L14), and a subgroup was established to ***identify overlaps and potential areas for harmonization*** (P3-L15, P3-L16).

In the openMDM case, the consortium focuses on ***jointly prioritizing goals*** to avoid conflicts (P2-I1).

Context C12. Enhancing cross-project collaboration

Problem C12.P: One of the priorities of LF Edge in 2020 and 2021 was to improve project harmonization and enhance cross-project collaboration (P3-L27, P3-L31).

Solutions C12.S: To address this problem, LF Edge applied one of its practices: the establishment of a cross-project subgroup. The goal of this subgroup was to ***identify collaboration opportunities and clarify the distinct use cases and markets*** targeted by each project (P3-L30, P3-L31, P3-L35). Furthermore, a white paper was crafted to outline how projects could collaborate to address key challenges and demonstrate how they align (P3-L30, P3-L31).

Additional strategies included creating ***a catalog of edge-based services*** spanning multiple projects (P3-L52) and ***developing cross-project demos*** linking at least two projects for specific use cases (P3-L31, P3-L35). These demos served two purposes: fostering collaboration among projects and expanding outreach by showcasing them at industry events (P3-L31, P3-L35, P3-L37).

Context C13. Focusing specific topics in depth

Problem C13.P: Collaborative work often entails diverse needs and expertise requirements, requiring insights from different perspectives (P4-I3). Furthermore, in some cases, to address disagreements and reach decisions, members of company-led OS consortia require more detailed information (P3-L2, P3-L15).

Solutions C13.S: In the case of LF Edge, to analyse specific topics in detail, the consortium ***establishes voluntary subgroups***. These subgroups produce reports on areas such as new project submissions (P3-L16), annual project reviews (P3-L19, P3-L43, P3-L44), and updates to the project lifecycle document (P3-L37). Additionally, TAC gathers feedback and improvement ideas through ***conducting community surveys***, such as one focused on prioritizing outreach committee activities (P3-L35, P3-L78, P3-LO-1).

openEHR hosts four ***sub-groups***, each specializing in different areas of expertise to address the needs of various stakeholders (P4-I2). Within these groups, prioritization, decision-making, and conflict resolution are expected to be managed internally (P4-I2). These groups maintain an open and inclusive approach, welcoming both organizations and individual members with diverse skills and experiences (P4-I2, P4-I3, P4-I4).

RACOON adopts a similar structure, organizing ***working groups*** to focus on specific topics, such as quality assurance (P4-J1, P4-J3). Each working group is led by a leader chosen by its members (P4-J3). These groups are open to all interested participants and encourage collaborative efforts (P4-J1, P4-J3).

Context C14. Sharing knowledge and experience among members

Problem C14.P: When a consortium hosts multiple projects, these projects may encounter similar issues at different stages of their development process. In the case of LF Edge, TAC meetings often focus on the progress of hosted projects. Project representatives share updates and discuss the challenges they face. Occasionally, multiple projects experience similar issues (P3-L70).

Solution C14.S: One practice involves ***providing a platform*** for members to ***share their experiences***. This includes regular meetings, such as TAC meetings in the LF Edge case (P3-L70, P3-L71), or using an online platform, as seen in the RACOON case (P4-J3).

Providing mentorship to hosted projects is another key strategy. For example, in the case of LF Edge, projects must secure sponsorship from at least two TAC members to advance to a higher lifecycle stage. These members provide guidance and support throughout the process (P3-L17).

Another approach involves ***mentoring candidate projects***. This helps them better understand the foundation's structure, rules, and expectations, ultimately enhancing overall efficiency (P3-I2, P3-L40, P3-L63).

Context C15. Sharing hardware resources among members

Problem C15.P: In certain instances, multiple projects hosted by the same consortium can utilize similar hardware, as observed in LF Edge (P3-L54, P3-L55). Allocating the budget to duplicate resources in such cases would result in inefficiency.

Solution C15.S: ***Creating a shared hardware resource pool***, such as LF Edge's Community Lab, improves budget efficiency by allowing members to share resources (P3-L61). For example, in 2020, the Akraino project opened its Community Lab to all members. The lab, supported by donations from the Linux Foundation and other organizations, provides shared hardware resources for project use (P3-LL-0, P3-L40, P3-L52).

Context C16. Aligning joint efforts with other consortia

Problem C16.P: Company-led OS consortia seek to increase collaboration opportunities with other organizations (P3-I4).

Solutions C16.S: LF Edge aims to enhance collaboration and communication with other foundations by ***assigning volunteers to attend meetings of other sector-related OS consortia*** like LF AI&Data, LF Networking, and the Eclipse Edge Native Working Group (P3-L70). This practice helps LF Edge stay informed about developments, align efforts, and ensure compatibility with standards. Additionally, LF Edge invites ***representatives from other consortia*** to attend their TAC meetings, enabling the exchange of insights and exploration of potential collaboration opportunities (P3-L70).

In the case of openMDM, the members primarily serve in the automobile industry. In Germany, this industry plays a significant role, supported by various associations and institutions. Each year, numerous meetings and events are held. ***Attending these sector-specific events*** allows consortium members to meet, connect, and collaborate (P2-I1).

Context C17: Improving projects' health

Problem C17.P: To maintain projects' health is a challenge for open source projects.

Solutions C17.S: In the case of LF Edge, the TAC focuses on creating guidance documents to support all LF Edge projects. One such document is the 'getting started checklist', a ***self-evaluation checklist*** to refine and enhance project management practices for new and existing projects (P3-L41, P3-L58, P3-LG-0). LF Edge conducts ***annual review cycles*** of its hosted projects based on TAC-developed criteria (LA-0). These reviews assess project status, verify alignment with lifecycle stages, and provide recommendations for addressing stage-level expectations (P3-L35, P3-L42-L45). Projects seeking higher lifecycle stages use these reviews to understand requirements and plan for progression. The results are transparently published on LF Edge's wiki, ensuring accountability and supporting project health (P3-L45).

Monitoring and regular assessment are also practices applied in the openMDM case (P2-I1, P2-I2). ***Regular milestone releases*** ease the monitoring process and facilitate the assessment of development progress (P2-I1, P2-I2).

Security is another priority for LF Edge, as projects handle security issues independently. TAC facilitates the sharing of best practices among members, including processes such as public reporting of vulnerabilities, forming Security Issue Review (SIR) teams, using automated vulnerability scanning and penetration testing, and developing threat models (P3-L70, P3-L71).

To improve project sustainability, LF Edge emphasizes ***ensuring independence of projects*** by fostering a balanced community and avoiding dominance by one or two companies. This approach ensures independence and promotes long-term health for projects (P3-L45, P3-L46, P3-L57).

Context 18. Integrating code

Problem 18.P: When different teams work on separate parts of a project without coordination but need to integrate their results to produce a common solution, integration problems can arise. The openMDM consortium encountered this issue at the beginning of its collaboration process (between 2014 and 2016). Initially, the members assigned software development responsibilities to the driver members. Each of these members funded, coordinated, and monitored their part of the development through different service providers. The plan was to integrate the separately developed components in June 2016 (P2-I1, P2-A6).

However, due to a lack of collaboration between service providers and the absence of a central control mechanism or monitoring system, vendor activities were unobservable, and one vendor shirked responsibility. As a result, the development process failed (P2-I1, P2-A6).

Furthermore, since the software components were developed separately, each provider utilized their own tools, repositories, and frameworks (P2-M35, P2-M44, P2-M45, P2-M46). Consequently, some components failed to integrate properly. As a result, the code integration process required more time and effort than anticipated, leading to a delayed release (P2-I1, P2-A3).

Solutions C18.S: Main practice openMDM applied to address this problem is changing their development strategy. They created a ***persistent team of developers*** using the shared pool of resources and assigned ***a project manager*** as the head of the software development team (P2-I1).

The consortium implemented a ***sanction mechanism*** for employed developers. Developer contracts are for a fixed term and are extended based on an evaluation of development efforts (P2-I1, P2-I2).

Using a ***single repository***—in the case of openMDM, this was Eclipse Git—is another established practice (P2-M31). This ensures that developers work within the same repository rather than using separate ones and merging code later. This approach has enhanced collaboration between teams, enabled code versioning, and improved monitoring of the development process (P2-I1, P2-I2, P2-M31). An exception to this approach is the code developed by external contributors, which is integrated into Eclipse Git only after a quality check (P2-M44, P2-M47, P2-M48, P2-M49).

Another practice is ***reviewing code*** before merging it into the main codebase. It is accepted only after being reviewed by another team member, ensuring higher code quality (P2-I2).

5.3.3 Sustainability Dimension

Context C19. Sustaining financial continuity

Problem C19.P: Company-led OS consortia operate as non-profit organizations. If they do not receive external funding, they rely on membership fees as their primary financial resource (P2-I1, P3-I2). Insufficient funding delays the workflow (P2-I1). Therefore, it is essential for the consortia to carefully manage and safeguard its financial resources.

Solutions C19.S: To address this issue, a practice LF Edge applies is to ***encourage member companies to invest more in hosted projects*** by ensuring that consortia activities and budgets align with their expectations, delivering clear value to members (P3-I2).

A further practice applied in both LF Edge and openMDM is to ***attract new members (and projects)*** to increase diversity, contributions, and financial support for the consortia (P2-I1, P3-I2, P3-I4).

Context C20. Expanding outreach and attracting new members

Problem C20.P: If a consortium relies on membership fees, it needs to attract new members to sustain and increase funding (P2-I1). Attracting new members and projects is a strategy to secure the financial stability of company-led OS consortia (P3-I2, P3-I4, P4-I1, P4-I4).

Solution C20.S: To attract more members, consortia should actively ***promote themselves and communicate their activities*** to a broader community through outreach and marketing efforts (P2-I1, P4-I4).

One approach for promoting projects is highlighting project missions, use cases, and interconnections through a ***taxonomy***. In LF Edge, this taxonomy identifies covered areas, gaps, and potential collaboration opportunities (P3-L5, P3-L14, P3-L37).

Another practice the LF Edge applies is to create a ***white paper*** that defines edge concepts, presents the taxonomy, and outlines market verticals (P3-I2, P3-L30, P3-L35, P3-L37, P3-L43, P3-LW-1).

A further strategy is to publish ***annual reports*** ("State of the Edge") ***on industry*** that explores news related to the state of critical infrastructure, networks, hardware, and software (P3-L47, P3-L51).

Both openMDM and LF Edge aim to raise awareness and attract members to their consortia by ***participating in sector-specific events***, for instance, as speakers (P2-I1, P3-L13, P3-L15, P3-L22, P3-L36, P3-L53, P3-L63).

openEHR ***encourages interested organizations to use and experience their technology*** and engage with the community to attract new members, since organizations that are familiar with existing projects are more likely to engage (P4-I2).

Context C21. Persuading decision makers to engage

Problem C21.P: Open-source software (OSS) projects are frequently misunderstood as being cost-free and offering no tangible benefits to participants (P3-I1, P3-I2, P3-I3). These misconceptions influence decision-makers when considering involvement in OSS projects. Additionally, licensing terms in certain standards can hinder organizations from contributing to OSS development and restrict end-users from utilizing it (P3-I3, P3-I4).

Solutions C21.S: Proposed solutions include explaining the benefits of involvement and showcasing ***how innovation can add value to their business***s, ***providing training on open source and its legal aspects***, and emphasizing ***the involvement of competitors*** in OSS projects (P3-I1, P3-I2, P3-I3, P3-I4).

Context C22. Getting end-user insights

Problem C22.P: In vendor-led OS consortia, members require input from end-user companies. In the LF Edge case, having a gap between products and their user needs is compounded by challenges in reaching a diverse range of adopters (P3-I2, P3-I3).

Solutions C22.S: In 2020, LF Edge launched the "Vertical Solutions - End User Community," followed by the ***End-User Solutions Group*** in 2021, aiming to understand user expectations and provide feedback on the usability and value of LF Edge projects (P3-I2, P3-L50, P3-L61, P3-LE-0). However, these initiatives were discontinued (P3-I2).

Alternatively, a member of LF Edge explained that their company addressed this issue by also ***joining a user-led open source consortium***—specifically, LF Energy—which focuses on end-user needs and expectations (P3-I3).

Context C23. Increasing the diversity in the developer community

Problem C23.P: A key motivation for vendors to participate in open-source projects is gaining access to a diverse pool of contributors (P3-I1, P3-I2, P3-B4). A lack of diversity within the community is an observed challenge for LF Edge (P3-I2).

Solutions C23.S: In the case of LF Edge, one practice for attracting more contributors is ***analyzing the enterprise developer market*** and targeting developers who create their own projects (P3-I2). Another approach involves offering ***mentorship programs and internship opportunities to students*** from various educational institutions (P3-I2, P3-L50, P3-L56, P3-LM-0). Additionally, ***organizing hackathons*** serves as an effective strategy (P3-L23, P3-L24, P3-L26).

Context C24. Attracting new users

Problem C24.P: In user-led OS consortia, more users lead to increased software usage, more bug reports, and improved software quality (P2-I2). Furthermore, these users may later become members of the consortia (P2-I1). Low number of users is a problem seen in the openMDM case.

Solutions C24.S: In the case of openMDM, ***being responsive to users*** is important, which shows that the project is alive. For instance, acknowledging reported bugs and informing users that they will be addressed is a way to show responsiveness (P2-I2).

Another practice discussed in the openMDM consortium is the ***creation and sharing of user stories***, which can serve as a type of requirements document and attract organizations with similar needs (P2-I1, P2-M3).

Since most members of the openMDM consortium are from German-speaking countries, they initially developed the graphical user interface (GUI) of their software in German. However, this created an obstacle for potential users from other countries (P2-I1, P2-M52). To overcome this challenge and attract more users from different regions, they developed a GUI in English (P2-I1). ***Using a multilingual GUI*** is another practice that enhances the accessibility and usability of the software.

openMDM follows a product-line development approach, allowing member companies to use openMDM software as a core and customize it by building tailor-made components based on their specific requirements. ***Offering customization*** is a practice openMDM applies to attract more users and members to the consortium (P2-I1). On the other hand, increasing modularity carries the risk of adding complexity to code development and slowing down the development process (P2-I2).

Context C25. Safeguarding data privacy and data security

Problem C25.P: Working with data presents challenges in safeguarding data privacy and security. This issue is evident in the RACOON case. Complying with data privacy and security regulations is a complex and time-consuming process due to varying requirements and the complicated nature of the task (P4-J1, P4-J3). The main challenge lies in safeguarding stored patient data from unauthorized access while facilitating multicentric scientific research using data from all member institutions (P4-J1).

Solutions C25.S: To address this problem, RACOON applies security measures such as ***data encryption*** and ***access control*** to prevent misuse and leaks. The consortium stores the data on its cloud servers by anonymizing them. They restrict access to data only to the authorized users (P4-J2).

Another practice applied in both RACOON and openEHR is ***delegating the responsibility for securing data*** to the data-owning organizations, not to the consortium (P4-I1, P4-I3, P4-J1, P4-J3).

openEHR ensures that the openEHR platform and published data models ***align with data regulation standards***, such as the General Data Protection Regulation in Europe and Health Information Privacy standards in the United States (P4-I3). Similarly, RACOON stays in alignment with the data security and privacy regulations (P4-J1, P4-J3).

Context C26. Alignment with market needs and diverse data regulations

Problem C26.P: Alignment with market needs is essential for making an impact and achieving success (P4-I3). In particular, in the openEHR case, which operates in the healthcare sector where things constantly evolve, staying up to date with market trends, regulations, and changes is a challenge (P4-I1).

Furthermore, each country has distinct regulations and requirements. It is challenging to ensure that data models fit to the regulations in different countries (P4-I1, P4-I3). Deployment of openEHR in different countries requires mapping between openEHR standards and the specific standards employed in various clinics (P4-I1).

Solutions C26.S: ***The involvement of policymakers in the community*** is valuable, as it helps keep the community informed about data regulations and enables them to address specific requirements (P4-I3).

Another approach is developing open-source mapping tools to ***define mapping processes***. However, due to the inherent diversity, these tools cannot address all variations, but they can provide a foundational basis. Software vendors can customize these solutions based on the requirements of individual institutions (P4-I1).

The main focus to handle this challenge is to ***establish interoperability and create uniform standards*** (P4-I1).

Table 5.3: Practices in Governance Dimension

ID	Definition	Cases
Context C1	**Mitigating management conflicts**	
Practice C1.S1	Establishing bounding rules	openMDM, LF Edge
Practice C1.S2	Separating governance and technical responsibilities	LF Edge
Practice C1.S3	Managing meetings professionally and ensuring transparency	openMDM, LF Edge
Context C2	**Mitigating the risk of a single company's dominance**	
Practice C2.S1	Including representatives from all membership categories in the governance board	openEHR
Practice C2.S2	Allocating resources based on voting results among board members	openMDM, openEHR

ID	Definition	Cases
Practice C2.S3	Including representatives from competing companies on the board	openEHR
Practice C2.S4	Rotating board members	openEHR
Practice C2.S5	Collectively reaching compromises and applying same rule to all members	LF Edge
Practice C2.S6	Sharing resources and offering equal rights to driver members	openMDM
Context C3	**Mitigating concerns about the influence of political factors on decision-making**	
Practice C3.S1	Applying same set of evaluation criteria to all projects	LF Edge
Practice C3.S2	Investigating cases in depth before decision-making	LF Edge
Practice C3.S3	Maintaining transparency in decision-making	LF Edge
Context C4	**Developing guidelines and reference documents**	
Practice C4.S1	Developing guidelines and documents	openMDM, LF Edge
Practice C4.S2	Revising documents as needed	LF Edge
Context C5	**Mitigating legal conflicts among members**	

ID	Definition	Cases
Practice C5.S1	Adhering to antitrust policies	LF Edge
Practice C5.S2	Complying with antitrust laws	openMDM
Context C6	**Finding a balance among members' expectations**	
Practice C6.S1	Maintaining transparency about reasons of concerns or requests	LF Edge
Practice C6.S2	Communicating and convincing others on the benefits of proposed changes	LF Edge
Practice C6.S3	Building positive relationships	LF Edge
Practice C6.S4	Maintaining a balanced representation of startups and large cooperations within the consortium	LF Edge
Context C7	**Facilitating mediation among diverse expert opinions**	
Practice C7.S1	Recognizing and appreciating the expertise and abilities of individuals involved when addressing issues	RACOON
Practice C7.S2	Facilitate open dialogue where each person has an opportunity to express their perspectives	RACOON
Practice C7.S3	Showing mutual respect and understanding to each other	openEHR, RACOON
Context C8	**Fostering dialogue and trust among members**	

ID	Definition	Cases
Practice C8.S1	Organizing face-to-face meetings	openMDM, LF Edge, openEHR, RACOON
Practice C8.S2	Organizing events for experience sharing (e.g. hackathons, workshops)	openMDM, LF Edge,
Practice C8.S3	Organizing regular meetings and ensuring periodic communication	openMDM, RACOON
Practice C8.S4	Providing an online platform for communication	RACOON
Practice C8.S5	Providing an open environment	LF Edge
Practice C8.S6	Maintaining transparency	openEHR
Context C9	**Providing an inclusive and open environment**	
Practice C9.S1	Sharing comprehensive information on governance structure and regulations to the new members	RACOON
Practice C9.S2	Having a well-structured onboarding protocol	RACOON
Practice C9.S3	Maintaining transparency about projects and processes	RACOON
Practice C9.S4	Showing appreciation to people who contributed to the success	openEHR

ID	Definition	Cases
Practice C9.S5	Explaining the impact of projects and sharing experiences with the community	openEHR
Practice C9.S6	Maintaining open standards	openEHR
Context C10	**Ensuring transparency**	
Practice C10.S1	Informing the community about the roadmap and resource allocation	openMDM, openEHR
Practice C10.S2	Publishing meeting minutes and related information about specific topics	openMDM, openEHR, RACOON
Practice C10.S3	Providing an open environment between members and governance board	openEHR, RACOON

Table 5.4: Practices in Efficiency and Productivity Dimension

ID	Definition	Cases
Context C11	**Competition among overlapping projects**	
Practice C11.S1	Encouraging collaboration among projects	LF Edge
Practice C11.S2	Identifying potential areas for harmonization	LF Edge
Practice C11.S3	Jointly prioritizing goals	openMDM

ID	Definition	Cases
Context C12	**Enhancing cross-project collaboration**	
Practice C12.S1	Identifying collaboration opportunities	LF Edge
Practice C12.S2	Clarifying use cases and target markets of each project	LF Edge
Practice C12.S3	Creating a catalog of services across multiple projects	LF Edge
Practice C12.S4	Developing cross-project demos	LF Edge
Context C13	**Focusing on specific topics in depth**	
Practice C13.S1	Establishing voluntary-based subgroups	LF Edge, openEHR, RACOON
Practice C13.S2	Conducting community surveys	LF Edge
Context C14	**Sharing knowledge and experience among projects**	
Practice C14.S1	Enabling experience share among projects	LF Edge, RACOON
Practice C14.S2	Mentoring hosted projects	LF Edge
Practice C14.S3	Mentoring candidate projects	LF Edge

ID	Definition	Cases
Context C15	**Sharing hardware resources among members**	
Practice C15.S1	Creating a pool of resources	LF Edge
Context C16	**Aligning joint efforts with other consortia**	
Practice C16.S1	Assigning representatives to other consortia	LF Edge
Practice C16.S2	Inviting representatives from other consortia	LF Edge
Practice C16.S3	Attending sector-specific events	openMDM
Context C17	**Improving projects' health**	
Practice C17.S1	Providing projects with a self-evaluation checklist	LF Edge
Practice C17.S2	Having annual review cycles for projects	LF Edge
Practice C17.S3	Monitoring and regular assessment	openMDM
Practice C17.S4	Regular milestone releases	openMDM
Practice C17.S5	Facilitating projects to address security issues	LF Edge

ID	Definition	Cases
Practice C17.S6	Ensuring independence of projects	LF Edge
Context C18	**Integrating code**	
Practice C18.S1	Persistent team of developers and a project manager	openMDM
Practice C18.S2	Sanction mechanisms fr employed developers	openMDM
Practice C18.S3	Using single repository	openMDM
Practice C18.S4	Reviewing code	openMDM

Table 5.5: Practices in Sustainability Dimension

ID	Definition	Cases
Context C19	**Sustaining financial continuity**	
Practice C19.S1	Convincing member companies to involve and invest more in hosted projects	LF Edge
Practice C19.S2	Attracting new members and projects	openMDM, LF Edge
Context C20	**Expanding outreach and attracting new members**	
Practice C20.S1	Promote consortia and their activities	openMDM, openEHR

ID	Definition	Cases
Practice C20.S2	Creating a taxonomy to show the coverage area of the projects	LF Edge
Practice C20.S3	Crafting publications about projects	LF Edge
Practice C20.S4	Publishing annual reports on the industry and the consortia	LF Edge
Practice C20.S5	Participating in sector-specific events	openMDM, LF Edge
Practice C20.S6	Encouraging interested organizations to experience the developed technology	openEHR
Context C21	**Persuading decision makers to engage**	
Practice C21.S1	Explaining benefits of involvement	LF Edge
Practice C21.S2	Providing trainings on open source and legal aspects	LF Edge
Practice C21.S3	Emphasizing the involvement of competitors in the projects	LF Edge
Context C22	**Getting end-user insights**	
Practice C22.S1	Creating a program focusing on end-user expectations	LF Edge
Practice C22.S2	Collaborating with a user-led OS consortium	LF Edge

ID	Definition	Cases
Context C23	**Increasing the diversity in the developer community**	
Practice C23.S1	Analyzing the enterprise developer market and targeting independent developers	LF Edge
Practice C23.S2	Offering mentorship programs and internship opportunities	LF Edge
Practice C23.S3	Organizing hackathon events to reach more developers	LF Edge
Context C24	**Attracting new users**	
Practice C24.S1	Being responsive to users	openMDM
Practice C24.S2	Creating and sharing user stories	openMDM
Practice C24.S3	Using a multilingual GUI	openMDM
Practice C24.S4	Offering customization	openMDM
Context C25	**Safeguarding data privacy and data security**	
Practice C25.S1	Anonymizing and storing data on a central cloud server with authorized access control	RACOON
Practice C25.S2	Delegating the responsibility for securing data to the data-owning organizations	openEHR, RACOON

ID	Definition	Cases
Practice C25.S3	Staying in alignment with data security regulations	openEHR, RACOON
Context C26	**Alignment with market needs and addressing diverse data regulations**	
Practice C26.S1	Including policymakers in the community	openEHR
Practice C26.S2	Developing open-source mapping tools to define mapping process	openEHR
Practice C26.S3	Establish interoperability and create uniform standards	openEHR

6

Discussion

This section discusses our findings in relation to three overarching research questions.

6.1 Characteristics of Company-Led Open Source Consortia

The main difference between user-led and vendor-led OS consortia is the driver members, who influence the development direction of projects. In user-led OS consortia, driver members are user organizations operating in non-software industries. They pool resources and guide the development of software needed for their internal processes. Conversely, in vendor-led OS consortia, companies in the software industry serve as driver members. They focus on developing generic software features that benefit the industry as a whole.

In user-led OS consortia projects, accepting volunteer contributors is uncommon. Feedback is generally received from users within member or user organizations. Conversely, in vendor-led OS consortia projects, feedback is also expected from individual developers who use the software or components in their own projects. Compared to user-led OS consortia, vendor-led OS consortia place greater emphasis on contributions from the open-source community.

In both user-led and vendor-led OS consortia, the output is not expected to provide a competitive advantage to members over one another. Instead, they focus on developing undifferentiated features.

In both user-led and vendor-led OS consortia, members sign formal agreements outlining binding rules, defined roles and responsibilities, and details of the consortium structure. This practice, emphasized in the business collaboration literature, contributes to successful collaboration (Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001).

Both user-led and vendor-led OS consortia commonly establish their own initiatives or collaborate with an umbrella foundation. These legal entities hold the IP rights for the consortia, collect membership fees, and provide support in governance, technical matters, and legal aspects (Riehle & Berschneider, 2012; Eckert et al., 2019). Some company-led OS consortia choose to establish their own foundations. For example, in our case, openEHR founded its own foundation. Conversely, openMDM and LF Edge collaborate with umbrella foundations—openMDM with the Eclipse Foundation and LF Edge with the Linux Foundation. Both approaches—establishing an independent foundation or working with an umbrella foundation—have their own benefits and drawbacks. For instance, since umbrella foundations already have established by-laws, structures, and processes, community members can focus on software development without being distracted by time-consuming governance tasks. In contrast, consortia that seek independence and full control over their resources opt for autonomous organization (Eckert et al., 2019).

The establishment process of company-led OS consortia, with all processes and decisions reviewed from the outset, would be a valuable topic for future research, providing useful insights for decision-makers involved in founding such projects.

6.2 Motivations of Companies to Create or Involve in Company-Led Open Source Consortia

The primary difference between user-led and vendor-led OS consortia lies in the motivation of driver members to create or join these consortia.

In user-led OS consortia, the primary motive is the collaborative development of software that member organizations require for their internal processes. One key reason for initiating such collaborations is the need to define specifications and lead the development process based on functional requirements. Partnering with organizations that share the same interest helps reduce development costs and enhance software functionality through experience and knowledge sharing.

In vendor-led OS consortia, the primary motives include generating revenue by building software components on collaboratively developed undifferentiated software and remaining actively engaged in innovation. Additionally, keeping pace with competing companies within the collaboration and participating in the establishment of industry standards are crucial factors for vendors. Overall, in both types of consortia some of the motives are similar. For instance, collaborative work helps reduce development and maintenance costs in both cases, making it more attractive for companies to join. Operating in a multi-organizational environment provides access to external knowledge, while exchanging knowledge and experience enhances productivity and accelerates innovation in the field.

Due to the open-source nature of the produced work, dependence on vendors decreases. Furthermore, its open-source nature enhances the adoption of collaborative work, which can facilitate the establishment of industry standards.

6.3 Good Practices to Follow for Successful Company-Led Open Source Consortia

Across four cases, we identified 90 practices in 26 contexts. We categorized these practices into three dimensions: governance, efficiency and productivity, and sustainability.

The practices discussed in this thesis represent a collection of diverse approaches. Some practices were applied in all four cases, while others appeared in only one. Rather than focusing solely on common practices, we aimed to collect and present as many practices as possible. This approach was designed to provide practitioners with a deeper understanding of the practices and assist them in selecting and applying those most suitable for their specific contexts.

The literature highlights building trust among members as one of the most critical success factors for collaboration (Mattessich & Monsey, 1992; Bruce et al., 1995; Rai et al., 1996; Hoffmann & Schlosser, 2001; Chin et al., 2008; Rikkiev & Mäkinen, 2009; Petter et al., 2014). Our analysis also found that fostering dialogue and trust among members was emphasized across all cases. Practices such as organizing face-to-face events, holding regular meetings, maintaining transparency, and providing an open environment are commonly applied in these consortia.

Maintaining transparency and fostering an open environment are key practices observed across various contexts. Transparency is ensured through practices such as sharing information about the roadmap and resource allocations, publishing minutes from regular meetings for transparency, and providing an open space where community members can engage with governing bodies.

In company-led OS consortia, driver member organizations typically operate within the same industry and are often direct competitors. To mitigate legal conflicts, they must comply with anti-trust regulations established by their companies and industries, as well as adhere to national anti-trust laws. In the openMDM case, besides maintaining good relations among members, transparency is emphasized to ensure compliance with anti-trust laws. The consortium consists of companies from the German automotive industry, which are prohibited from holding private discussions. Instead, they are required to communicate and plan openly, in a public and transparent manner.

Creating an open and inclusive environment facilitates the adaptation process for new members. For instance, well-structured onboarding processes and clear communication of rules and governance structures are established practices in community-led OSS projects (Barcomb et al., 2020). Additionally, recognizing and appreciating contributors is a further practice to encourage participation applied in the openEHR consortium.

Sharing knowledge and experience among members enhances productivity in collaborations, aligning with findings in the business collaboration literature (Chin et al., 2008; Petter et al., 2014). This practice is applied in both user-led and vendor-led OS consortia. Additionally, having mentors who provide guidance within collaborations is another practice that contributes to increased productivity (Bruce et al., 1995; Rai et al., 1996; Rikkiev & Mäkinen, 2009). In the case of LF Edge, the mentorship practice is applied to both candidate and ongoing projects.

In the sustainability dimension, the primary problem we observed was ensuring financial sustainability. In the cases of openMDM, LF Edge, and openEHR, attracting new members is crucial, as these consortia rely on membership fees. On the other hand, at the time of our research, the RACOON consortium was funded through external funds, an observed practice in the early stages of other user-led OS consortia. During the external funding phase, consortia do not require additional financial support. However, once this funding period ends, they must seek new resources to sustain their projects.

To attract new members, consortia employ various strategies, including speaking at conferences, publishing annual reports on their projects and industry trends, and releasing white papers related to their initiatives. However, convincing decision-makers to join open-source projects remains a challenge. Despite the long history of OSS projects, some managers are still hesitant to engage with them. LF Edge consortium members highlighted several practices to address this hesitation, such as explaining the benefits of participation, providing training on OSS dynamics, and emphasizing the presence of competitive companies within the consortium.

A key difference between user-led and vendor-led OS consortia lies in their expectations of users. In the openMDM case, users are seen as potential new consortium members, making it essential to promote the project and attract their attention. Conversely, in the LF Edge case, users are considered potential customers for each of the involved companies. As collaborative efforts to gather end-user insights did not continue, one LF Edge member shared their individual solution: joining a user-led OS consortium.

Another key difference between vendor-led and user-led OS consortia is the role of the developer community. In both LF Edge and openEHR, our interviewees emphasized the importance of diversity within the developer community. In contrast, openMDM and RACOON consortia do not focus on attracting developers. In the case of LF Edge, strategies for attracting new talent and engaging them in projects include offering internship opportunities for students and organizing hackathons.

These practices were collected through key stakeholder interviews and document reviews on two user-led OS consortia and two vendor-led OS consortia across the automotive, healthcare, and information technology industries. As a future research direction, distributing a quantitative questionnaire to additional consortia in various industries could further validate these findings and contribute to the development of a cross-industry best practices guidebook.

7

Conclusion

The focus of this thesis was on company-led open source consortia. We categorized company-led OS consortia into two types: user-led open-source (OS) consortia and vendor-led open-source (OS) consortia. These two types of company-led OS consortia differ from each other in terms of their driver members and the motives for their involvement.

User-led OS consortia are consortia led by end-user organizations from non-software industries. Their primary goal is to develop non-differentiating software that meets their functional requirements for internal business use. These consortia evolve through the involvement of companies from the same industry that share similar needs. Vendor-led OS consortia are consortia led by information technology companies. Their primary goal is to collaboratively develop a non-differentiating software base, which can later serve as a foundation for developing differentiating components to be sold for profit.

One of the contributions of this thesis is the identification of the characteristics of user-led and vendor-led OS consortia, with a focus on their consortium structure, membership structure and key actors, adopted software development approach, and the outputs of collaborative work.

An additional contribution of this thesis is the identification of the motives behind companies' involvement in these consortia. We categorized the motives of driver members in user-led OS consortia into four dimensions: control, cost, productivity and innovation, and sustainability. The primary motive for these organizations is to steer the development direction of the software they need. Similarly, we categorized the motives of driver members in vendor-led OS consortia into four dimensions: revenue, competition, productivity and innovation, and reputation. While the productivity and innovation aspects are similar in both types of consortia, the other dimensions and motives differ.

Although these consortia have different dynamics and their members have distinct motives, their governance structures and collaboration dynamics are similar. Based on our findings from four cases, we identified 90 practices across 26 contexts that contribute to successful collaboration. The findings presented in this thesis provide a foundation for further research on both user-led and vendor-led OS consortia. Furthermore, they offer insights for practitioners already involved in such consortia or considering participation.

8

REFERENCES

Ågerfalk, P. J., & Fitzgerald, B. (2008). Outsourcing to an unknown workforce: Exploring opensourcing as a global sourcing strategy. MIS Quarterly, 32(2), 385-409. DOI:10.2307/25148845

Baldwin, C., & Von Hippel, E. (2011). Modeling a paradigm shift: From producer innovation to user and open collaborative innovation. *Organization Science, 22(6),* 1399-1417. DOI:10.2139/ssrn.1502864

Barcomb, A., Stol, K. J., Fitzgerald, B., & Riehle, D. (2020). Managing episodic volunteers in free/libre/open source software communities. *IEEE Transactions on Software Engineering*, 48(1), 260-277.

Brandenburger, A.M. and Nalebuff, B.J. (1996), Co-Opetition, Doubleday Currency, New York, NY.

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative research in psychology*, *3*(2), 77-101.

Bruce, M., Leverick, F., Littler, D., & Wilson, D. (1995). Success factors for collaborative product development: a study of suppliers of information and communication technology. *R&D Management, 25(1)*, pp. 33-44. DOI:10.1111/j.1467-9310.1995.tb00898.x

Bryman, A. (2016). Social research methods. Oxford university press.

Chesbrough, H. W., & Appleyard, M. M. (2007). Open innovation and strategy. California management review, 50(1), 57-76.

Chin, K. S., Chan, B. L., & Lam, P. K. (2008). Identifying and prioritizing critical success factors for coopetition strategy. *Industrial Management & Data Systems.* DOI:10.1108/02635570810868326

Courant, P. N., & Griffiths, R. J. (2006). Software and collaboration in higher education: A study of open source software. *New York: Ithaca. Retrieved January*, *30*, 2009.

Díaz, O., Genero, M., Contell, J. P., & Piattini, M. (2024). Adding relevance to rigor: Assessing the contributions of SLRs in Software Engineering through Citation Context Analysis. *Computer Science Review*, *53*, 100649.

Dybå, T., Dingsoyr, T., & Hanssen, G. K. (2007, September). Applying systematic reviews to diverse study types: An experience report. In *First international symposium on empirical software engineering and measurement (ESEM 2007)* (pp. 225-234). IEEE

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review, 14(4)*, 532-550. DOI:10.5465/amr.1989.4308385

Eckert, R., Stuermer, M., & Myrach, T. (2019). Alone or Together? Inter-organizational affiliations of open source communities. *Journal of systems and software*, *149*, 250-262.

Felizardo, K. R., Mendes, E., Kalinowski, M., Souza, É. F., & Vijaykumar, N. L. (2016, September). Using forward snowballing to update systematic reviews in software engineering. *In Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement* (pp. 1-6).

Fitzgerald, B. (2006). The transformation of open source software. *MIS quarterly*, 587-598.

Fortuin, F. T., & Omta, S. W. F. (2008). The dark side of open innovation: a survey of failed inter- company cooperation. *In Proceedings of the 8th International Conference on Management in AgriFood Chains and Networks.*

Germonprez, M., Allen, J. P., Warner, B., Hill, J., & McClements, G. (2013). Open source communities of competitors. *Interactions, 20(6)*, 54-59. DOI:10.1145/2527191

Grand, S., Von Krogh, G., Leonard, D., & Swap, W. (2004). Resource allocation beyond firm boundaries: A multi-level model for open source innovation. *Long Range Planning*, *37*(6), 591-610.

Guba, E. G. (1981). Criteria for assessing the trustworthiness of naturalistic inquiries. *Educational Communication and Technology, 29*(2), pp. 75-91. DOI:10.1007/BF02766777

Guion, R. M. (2011). *Assessment, measurement, and prediction for personnel decisions*. Taylor & Francis.

Harutyunyan, N., Riehle, D., & Sathya, G. (2020). Industry best practices for corporate open sourcing. In *Proceedings of the 53rd Hawaii International Conference on System Sciences*.

Hoffmann, W. H., & Schlosser, R. (2001). Success factors of strategic alliances in small and medium-sized enterprises—An empirical survey. *Long Range Planning*, *34(3),* 357-381. https://doi.org/10.1016/S0024-6301(01)00041-3

Hunter, P., & Walli, S. (2013). The rise and evolution of the open source software foundation. *IFOSS L. Rev.*, *5*, 31.

Joo, C., Kang, H., & Lee, H. (2012). Anatomy of open source software projects: evolving dynamics of innovation landscape in open source software ecology. In *The 5th International Conference on Communications, Computers and Applications (MIC-CCA2012)* (pp. 96-100). IEEE.

Iivari, N., Hedberg, H., & Kirves, T. (2008, September). Usability in company open source software context-Initial findings from an empirical case study. In *IFIP International Conference on Open Source Systems* (pp. 359-365). Boston, MA: Springer US.

Izquierdo, J. L. C., & Cabot, J. (2020). A Survey of Software Foundations in Open Source. arXiv preprint arXiv:2005.10063.

Kitchenham, B. (2004). Procedures for performing systematic reviews. Keele, UK, Keele University, 33(2004), 1-26.

Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009).

Systematic literature reviews in software engineering–a systematic literature review. *Information and software technology*, *51*(1), 7-15.

Kitchenham, B., & Brereton, P. (2013). A systematic review of systematic review process research in software engineering. *Information and software technology*, *55*(12), 2049-2075.

Levy, M., & Germonprez, R. M. (2015). Is it egalitarianism or enterprise strategy? Exploring a new method of innovation in open source. *In 21st AMCIS 2015*.

Linåker, J., & Regnell, B. (2020). What to share, when, and where: balancing the objectives and complexities of open source software contributions. *Empirical Software Engineering*, *25*, 3799-3840.

Midha, V., & Palvia, P. (2012). Factors affecting the success of Open Source Software. *Journal of Systems and Software, 85(4),* 895-905. DOI:10.1016/j.jss.2011.11.010

Mattessich, P. W., & Monsey, B. R. (1992). *Collaboration: What Makes It Work. A review of research literature on factors influencing successful collaboration.* Amherst H. Wilder Foundation.

Patton, M. Q. (1990). Qualitative evaluation and research methods. SAGE Publications, Inc.

Petter, R. R. H., Resende, L. M., de Andrade Júnior, P. P., & Horst, D. J. (2014). Systematic review: an analysis model for measuring the coopetitive performance in horizontal cooperation networks mapping the critical success factors and their variables. *The Annals of regional science*, *53*, 157-178.

Pinto, G., Steinmacher, I., Dias, L. F., & Gerosa, M. (2018). On the challenges of open-sourcing proprietary software projects. *Empirical Software Engineering*, *23*, 3221-3247.

Rai, A., Borah, S., & Ramaprasad, A. (1996). Critical success factors for strategic alliances in the information technology industry: An empirical study. *Decision Sciences, 27(1)*, 141-155. DOI:10.1111/j.1540-5915.1996.tb00848.x

Radtke, N. P., Janssen, M. A., & Collofello, J. S. (2009). What makes Free/Libre Open Source Software (FLOSS) projects successful? An agent-based model of FLOSS projects. *International Journal of Open Source Software and Processes*, *1*(2), 1-13. DOI:10.4018/jossp.2009040101

Riehle, D. (2010). The economic case for open source foundations. *Computer, (1)*, 86-90. DOI:10.1109/MC.2010.24

Riehle, D., & Berschneider, S. (2012). A model of open source developer foundations. In *IFIP International Conference on Open Source Systems*.

Riehle, D. (2019). The innovations of open source. Computer, 52(4), 59-63.

Rikkiev, A., & Mäkinen, S. (2009). Success factors for technology integration convergence collaborations: Empirical assessment. In *2009 Portland International Conference on Management of Engineering & Technology*. DOI:10.1109/PICMET.2009.5262187

Roberts, R. E. (2020). Qualitative Interview Questions: Guidance for Novice Researchers. *Qualitative Report*, 25(9)

Schaarschmidt, M., Bertram, M., & von Kortzfleisch, H. F. (2011). Exposing differences of governance approaches in single and multi vendor open source software development. *In IFIP International Working Conference on Governance and Sustainability in Information Systems-Managing the Transfer and Diffusion of IT.*

Schaarschmidt, M. and Stol, K.-J., (2018) 'Company Soldiers and Gone-Natives: Role Conflict and Career Ambition Among Firm-Employed Open Source Developers', forthcoming conference, ICIS 2018, San Francisco, 13-16 December.

Schwab, B., Riehle, D., Barcomb, A., & Harutyunyan, N. (2020). The Ecosystem of openKonsequenz, A User-Led Open Source Foundation. In *IFIP International Conference on Open Source Systems*. DOI:10.1007/978-3-030-47240-5_1

Shaikh, M., & Cornford, T. (2010, January). 'Letting go of Control to Embrace Open Source: Implications for Company and Community. In *2010 43rd Hawaii International Conference on System Sciences* (pp. 1-10). IEEE.

Strauss, A., & Corbin, J. (1990). *Basics of qualitative research: Grounded theory procedures and techniques.* Sage Publications, Inc.

Teixeira, J., & Lin, T. (2014, May). Collaboration in the open-source arena: the webkit case. In *Proceedings of the 52nd ACM conference on Computers and people research* (pp. 121-

129).

Teixeira, J., Mian, S., & Hytti, U. (2016). Cooperation among competitors in the open-source arena: The case of OpenStack. *arXiv preprint arXiv:1612.09462*.

Weikert, F., Riehle, D., & Barcomb, A. (2019, October). Managing commercial conflicts of interest in open source foundations. In *International Conference on Software Business* (pp. 130-144). Cham: Springer International Publishing.

West, J., & O'Mahony, S. (2005). Contrasting community building in sponsored and community founded open source projects. *In Proceedings of the 38th Annual HICSS.* IEEE. DOI:10.1109/HICSS.2005.166

West, J., & Gallagher, S. (2006). Challenges of open innovation: the paradox of firm investment in open-source software. *R&D Management*, *36*(3), 319-331. DOI:10.1111/j.1467-9310.2006.00436.x

Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th international conference on evaluation and assessment in software engineering* (pp. 1-10).

von Hippel E., & von Krogh G., (2003). Open-source software and the "Private-Collective" Innovation Model: Issues for Organization Science. Organization Science 14(2):209-223.

Yenişen Yavuz, E., Barcomb, A., & Riehle, D. (2022). Problems, Solutions, and Success Factors in the openMDM User-Led Open Source Consortium. *Communications of the Association for Information Systems*, *51*(1), 13. https://doi.org/10.17705/1CAIS.05122

Yenişen Yavuz, E., & Riehle, D. (2025) Why and How Do Organizations Create User-Led Open Source Consortia? A Systematic Literature Review. *In Information and Software Technology*, 107681. https://doi.org/10.1016/j.infsof.2025.107681

Yenişen Yavuz, E., Riehle, D., & Mehrotra, A. (2025). Why do companies create and how do they succeed with a vendor-led open source foundation. *Empirical Software Engineering*, *30*(1), 1-49. https://doi.org/10.1007/s10664-024-10588-9

Yenişen Yavuz, E., Shrivastava, A., Riehle, D., & Putz, F. (2025). Governance Practices for

Open Source Foundations in the Healthcare Sector. In: Papatheocharous, E., Farshidi, S., Jansen, S., Hyrynsalmi, S. (eds) Software Business. ICSOB 2024. Lecture Notes in Business Information Processing, vol 539. Springer, Cham. https://doi.org/10.1007/978-3-031-85849-9_30.

Yin, Robert K. (2018). *Case study research and applications*. Sage.

Zhang, Y., Zhou, M., Stol, K. J., Wu, J., & Jin, Z. (2020). How do companies collaborate in open source ecosystems? an empirical study of openstack. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering* (pp. 1196-1208).

Zhang, Y., Zhou, M., Mockus, A., & Jin, Z. (2021). Companies' participation in OSS development–an empirical study of openstack. *IEEE Transactions on Software Engineering*, *47*(10), 2242-2259.

Zhang, Y., Liu, H., Tan, X., Zhou, M., Jin, Z., & Zhu, J. (2022a). Turnover of companies in OpenStack: Prevalence and rationale. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, *31*(4), 1-24.

Zhang, Y., Stol, K. J., Liu, H., & Zhou, M. (2022b). Corporate dominance in open source ecosystems: a case study of OpenStack. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (pp. 1048-1060).

A

Paper 1

This chapter is based on the article titled ”Why and How Do Organizations Create User-Led Open Source Consortia? A Systematic Literature Review” originally published in *Information and Software Technology*:

Yenişen Yavuz, E., & Riehle, D. (2025). Why and How Do Organizations Create User-Led Open Source Consortia? A Systematic Literature Review. In *Information and Software Technology*, 107681. https://doi.org/10.1016/j.infsof.2025.107681

Information and Software Technology 181 (2025) 107681

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Why and how do organizations create user-led open source consortia? A systematic literature review

Elçin Yenişen Yavuz *, Dirk Riehle

Computer Science Department, Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany

A R T I C L E I N F O

Keywords:
Open source foundations
User-led open source consortia
Collaborative software development
Open-source software projects
User-driven open-source software development
Community-source software development
Coopetition
SLR
Systematic literature review

A B S T R A C T

Context: User-led open source (OS) consortia (foundations) consist of organizations from industries beyond the software industry collaborating to create open-source software solutions for their internal processes. Initially pioneered by higher education organizations in the 2000s, this concept has gained traction in recent years across various industries.

Objective: This study has two research objectives. The first objective is to provide an overview of the current state of the art in this field by identifying previously studied topics and gathering examples from different industries. The second objective is to understand the structure of user-led OS consortia and the motivations of organizations for participating in such consortia.

Method: To gain a comprehensive understanding of this phenomenon, we conducted a systematic literature review, covering the years 2000 to 2023. Furthermore, we performed thematic analysis on 43 selected studies to identify and examine the key characteristics, ecosystems, and the benefits organizations gain from involvement in user-led OS consortia.

Results: We identified 43 unique papers on user-led OS consortia and provided details on 14 sample user-led OS consortia projects. We defined 19 characteristics of user-led OS consortia and 16 benefits for organizations' involvement. Additionally, we outlined the key actors and their roles in user-led OS consortia.

Conclusion: We provided an overview of the current state of the art in this field. We identified the structure of user-led OS consortia and the organizations' motivations for participating in such consortia.

1. Introduction

Open-source software (OSS) development is a development approach where source code is openly shared, allowing developers and software engineers to use, modify, and contribute to it while collaborating on the development process—all without charge. Initially, OSS development projects were primarily driven by individual contributors; however, over time, corporate organizations began participating actively [1].

One strategy for corporate involvement in OSS is to release proprietary software as open-source code and foster a community around it [2–5]. This strategy enables companies to establish widely recognized standards, drive innovation, develop markets for complementary products and services, and build positive relationships with their target audience [5].

Another strategy is to engage in OSS development in collaboration with other corporate entities. These collaborations typically occur under legal entities such as foundations or consortia. We classify the collaborative OSS development approaches of organizations into two categories: vendor-led open source foundations (or consortia) and user-led open source foundations (or consortia).

In vendor-led open source (OS) consortia, collaborative efforts are primarily driven by software vendor organizations aiming to develop software components for use in their products [6–8]. In contrast, in user-led open source (OS) consortia, development efforts are steered by organizations from non-software industries with the goal of developing software tailored to their specific internal needs [9,8]. The common characteristic of these two types of consortia is that their leading members are organizations rather than individuals.

OSS projects steered by companies, rather than individuals, are becoming increasingly common in practice. However, the literature on these types of foundations and projects remains limited. Few studies have focused on vendors' involvement in such projects (e.g., [10,8]). On the other hand, the literature lacks a clear explanation of the definition and structure of "user-led OS consortia." In OSS literature, "users" are

* Corresponding author at: Friedrich-Alexander-Universität Erlangen-Nürnberg, Professur für Open-Source-Software, Martensstr. 3, 91058 Erlangen, Germany.
E-mail addresses: elcin.yenisen@fau.de (E. Yenişen Yavuz), dirk@riehle.org (D. Riehle).

https://doi.org/10.1016/j.infsof.2025.107681
Received 6 August 2024; Received in revised form 27 January 2025; Accepted 30 January 2025
Available online 1 February 2025

primarily defined as volunteer developers who are often the end users of the software they contribute to. However, the involvement of user organizations in OSS development and their collaborative efforts with other organizations have not been systematically investigated.

The first examples of user-led OS consortia emerged in higher education in the early 2000s, pioneered by universities in the United States. This expansion was driven by the growing demand for customized software solutions and the need for independence from vendors. Since then, user-led OS consortia have gained significant traction and popularity across various industries.

Current literature on user-led OS consortia primarily focuses on specific project examples, with most of the investigated projects originating in the education sector. However, a comprehensive explanation of the structure of this model across different industries is lacking.

In this research, we have two research objectives (RO): Our first RO is to provide an overview of the current state of the literature on user-led OS consortia. Our second RO is to identify the general structure of user-led OS consortia and define the motivations of organizations for engaging in these consortia.

To achieve our first RO, we formulated three research questions (RQ). These are:

RQ.1.1. What is the current state of the art in the literature on user-led open source consortia?
RQ.1.2. Which user-led open source consortia have been investigated in the literature?
RQ.1.3. Which research topics about user-led open source consortia does the literature address?

To achieve our second RO, we have three additional RQs:

RQ.2.1. What are the defining characteristics of user-led open source consortia?
RQ.2.2. How do organizations engage with user-led open source consortia?
RQ.2.3. Why do user organizations create user-led open source consortia?

To address these research questions, we conducted a systematic literature review (SLR) by following the guidelines of Kitchenham [11] and Kitchenham & Brereton [12]. We analyzed the relevant literature qualitatively using thematic analysis [13].

As a result of our analysis, we contribute to the literature with the following key findings:

- We identify *43 unique papers* on user-led OS consortia and *synthesize information* they provided on different topics. With this contribution, we aim to provide a literature list for researchers working on this topic, and for practitioners, considering involvement in OS consortia projects.
- We present *14 sample user-led OS consortia projects* from the 44 papers we reviewed. We collected and presented information about these projects, including their industry, initiation goal, initiation year, and status as of May 2024. This contribution proves that this approach is applicable across different industries and captures the attention of various stakeholders.
- We identify *19 defining characteristics* of user-led OS consortia based on three key features: being led by user organizations, following collaborative software development approaches, and offering the developed software as open source. Furthermore, we present *key actors and their roles within the ecosystem* of user-led OS consortia. These contributions are beneficial for establishing governance practices around such consortia. They serve as the first steps to provide an understanding about their structure and management.
- We define *16 motivations* behind organizations' participation in user-led OS consortia by comparing the benefits of this approach with alternative options. This contribution is beneficial for practitioners to better understand these motives, communicate the benefits to others, and attract more members to their projects.

The structure of this paper is as follows: In Section 2, we present related work on open source foundations and user-led OS consortia definitions. In Section 3, we describe the methodology we employed in this study. We present the obtained results in Section 4, and discussion in Section 5. In Section 6, we outline the limitations of our study. Lastly, in Section 7 we provide the conclusion of this study.

2. Related work

In Section 2.1, we provide an overview of open source foundations, while in Section 2.2, we present the definitions of user-led open-source consortia used in the literature.

2.1. Open source foundations

Open source (software) foundations are non-profit organizations that serve as impartial platforms for open-source software (OSS) projects. They play a crucial role in managing and distributing funds to support these initiatives, while also protecting the rights of project members and contributors through a legal framework. Additionally, they often provide governance support to their members [14,15,6].

The origins of free and open-source software (OSS) can be traced back to the Free Software Foundation (FSF), established by Richard Stallman in 1985. Stallman is credited with introducing the concepts of 'copyleft'—an alternative to copyright— and the 'General Public License' [16]. The FSF is a non-profit organization that collects and distributes funds for early software development projects, most notably the GNU Project, which aimed to develop a completely free operating system [17].

Other early examples of open source foundations include the Apache Software Foundation, the Linux Foundation, and the Eclipse Foundation. The Apache Software Foundation was established in 1999 to ensure the continuity of original HTTP Project and subsequent projects [18]. The collaborative approach, meritocratic governance structure and community development process of ASF became a blueprint for other open source foundations.[1]

With the increased adoption of Linux operating systems, the popularity of Linux-kernel projects grew. As the scope, complexity, and number of contributors to the Linux-kernel project grew, support for its expanding community was required. To address this need, the Linux Foundation was established in 2000 [18,19]. Similarly, the Eclipse Foundation was founded in 2004 to support the Eclipse IDE project [18].

All of these foundations were initially established to support community-led OSS projects. Community-led OSS projects are initiated and managed by individual developers [8]. These projects follow a meritocratic governance model, where contributors gain governance roles based on the consistency and quality of their contributions [6,20]. In most community-led OS foundations, corporate entities are not formally recognized as members. However, they can support hosted OSS projects by funding individual contributors, offering infrastructure resources, or sponsoring project-related events [21].

The Apache Foundations has continued in this direction, accepting only individual members. To support community-led OS projects, it accepts sponsorship from organizations, but does not allow institutions to be the members of the projects.[2] On the other hand, the Linux Foundation and the Eclipse Foundation accept both individual and institutional members. They have expanded their OSS project portfolios to include community-led OS projects, vendor-led OS consortia projects,

[1] https://www.apache.org/foundation/how-it-works/
[2] https://www.apache.org/foundation/governance/

and user-led OS consortia projects.

Vendors participate in collaborative open-source development projects to help establish industry standards, accelerate innovation within the field, and enhance productivity through resource sharing [7,10,8, 22]. Furthermore, vendors may offer complementary hardware, software, or services related to the open-source projects they support [10, 22]. Developers working on these projects are mostly paid employees of member companies [7]. However, volunteer contributors are also welcome to these projects. Some examples of vendor-led OS consortia hosted by the Linux Foundation are LF Edge,[3] and the Cloud Native Computing Foundation.[4] Examples from the Eclipse Foundation include Eclipse IoT Working Group[5] and the Adoptium Working Group[6] [8].

User-led OS consortia involve end-user organizations that steer OSS development by providing requirements and financial incentives. These organizations are the primary consumers of the OSS being developed. IT service providers primarily act as development partners that implement the specifications and develop the software [9].

We use the terms foundation and consortium synonymously in this article. Riehle & Berschneider [6] explain the distinction between these terms based on their goals—whether they serve their members or the public—, and their jurisdiction of incorporation, which depends on the country in which they are established. However, rather than focusing on the legal distinctions between these collaborations, we emphasize their structure and objectives in relation to software development efforts.

2.2. Definition of user-led open source consortia

The first examples of user-led open source consortia were observed in higher education. Courant & Griffith [23] used the term “directed open source” to describe this model. Wheeler [24] and Liu et al.[25] investigated examples in higher education and referred to this approach as “community-source software development”.

Chesbrough & Appleyard [26] and Perr et al. [27] classified “community-source software development” as an open source business model and explored the benefits of involvement in these collaborations from the perspective of software vendors. Chesbrough & Appleyard [26] referred to this business model as “self-service”, defining it as “consortia of end-user organizations”.

Riehle [28] classified OSS foundations into two categories: developer foundations and user foundations. Developer foundations are those that are steered by software vendors or individual developers. In contrast, user foundations are founded and managed by user organizations—rather than software vendors—with the goal of developing OSS for their own use.

Almigheerbi et al.[29] proposed implementing this model in Libyan higher education organizations, focusing on the development of ERP packages. They referred to this model as “Collaboratively-Developed Enterprise Resource Planning (CD-ERP)”.

Schwab et al. [30] and Yenişen Yavuz et al. [9] referred to this model as “user-led open source consortia”. Yenişen Yavuz et al. [9] highlighted the potential for confusion when referring to the model employed in higher education projects as "community-source software development," as this term may mistakenly imply a similarity to open source projects led solely by developers without organizational involvement—commonly referred to as community-led open source development.

As this research constitutes a literature review, we employ diverse terminology to explore relevant findings. However, we call this phenomenon “user-led open source foundations” and “user-led open source consortia”. In the rest of the paper, we use the term “user-led OS consortia” to explain this phenomenon.

[3] https://lfedge.org/
[4] https://www.cncf.io/
[5] https://iot.eclipse.org/
[6] https://adoptium.net/

3. Methodology

We chose to perform a systematic literature review (SLR) for this research. An SLR is a form of secondary study that focuses on “identifying, evaluating and interpreting all available research relevant to a particular research question, topic area, or phenomenon of interest” [11]. An SLR is a method for adopting an Evidence-Based Software Engineering (EBSE) research approach. Inspired by evidence-based medicine, EBSE aims to support software development decisions by synthesizing insights from high-quality research studies [31].

The potential contributions of SLRs can be categorized as “backward-oriented”, which involves synthesizing existing knowledge or aggregating evidence from previous studies, and “forward-oriented”, which focuses on theory building or identifying research gaps for future exploration [32]. Our research focuses on both backward and forward orientations. Our first objective is to synthesize existing knowledge by investigating literature on user-led OS consortia. Our second objective is to develop a theory based on the information we have collected.

We employed the methodologies proposed by Kitchenham [11] and Kitchenham & Brereton [12] in conducting this systematic literature review. In the initial step, we investigated existing systematic literature reviews on user-led OS consortia; however, we did not find any. In the second step, we developed a literature review protocol outlining our research goals and rationale for the literature review, our search strategy, paper selection criteria, and data extraction strategy. We adhered to this protocol. In the third step, we conducted the review and documented the results. Finally, in the fourth step, we report the review process and its result in this paper. In the following section, we describe the details of the third step: “conducting the review”.

3.1. Search strategy

In the search strategy step, we defined the keywords, specified the timeframe, and selected the digital libraries for the search.

We identified four sets of keyword lists. We began with terms commonly used in the literature to define this model, such as “community source”, “directed open source”, and “user-led OS consortia.” The second set includes terms related to the structure of the software development process, such as collaborative OSS development, intercompany OSS development, and sponsored OSS development.

User-led open source consortia can take different organizational forms, such as foundations, working groups, or consortia. Organizations can initiate their own foundations, or alternatively, they can operate under an established umbrella foundation. In our third set of keywords, we included prominent umbrella foundations for user-led OS consortia, namely the Apereo Foundation and the Eclipse Foundation (Working Groups).

During our prior research [9], we encountered notable user-led OS projects and consortia, including Kuali, Sakai, openKonsequenz, openMAMA, and the Academy Software Foundation. We utilized these terms as the fourth set of keywords, which we continuously updated as we discovered new projects or consortia.

In Table 1, we provide a detailed keyword list and its corresponding search results.

Set 1: Terms which are used in the literature to define user-led OS consortia such as “community source”, “directed open source”, “user-led open source consortia”, and “user-led open source foundations”
Set 2: Terms which are used to define organizational involvement in open-source software development such as “collaborative open-source software development”, “intercompany OSS development”, and “sponsored OSS development”.

Table 1
List of keywords and search results.

Classification	Keyword	Search results	After 1st step	After 2nd step	After 3rd step
Set 1: Definition	Community source AND open source	1323	1067	352	40
	Directed open source	23	19	2	1
	User led open source consortia	0	0	0	0
	User led open source consortium	0	0	0	0
	User-led open source consortia	1	0	0	0
	User-led open source consortium	0	0	0	0
	User-led open source foundation	11	1	1	1
Set 2: Open source development approach	Collaborative open-source software development	99	82	56	0
	Collaborative OSS development	33	28	24	0
	Company led open-source software development	0	0	0	0
	Company led OSS	4	4	3	0
	Company led OSS development	0	0	0	0
	Company-led open-source software development	0	0	0	0
	Company-led OSS development	0	0	0	0
	Inter-company open-source software development	0	0	0	0
	Inter-company OSS development	0	0	0	0
	Intercompany open-source software development	0	0	0	0
	Intercompany OSS development	0	0	0	0
	Open source cooperative projects	0	0	0	0
	Sponsored open-source software development	6	6	3	0
	Sponsored OSS development	2	2	1	0
	User led open source cooperative	0	0	0	0
	User-led open source cooperative	0	0	0	0
	User-sponsored OSS development	0	0	0	0
Set 3: Umbrella Foundation	Apereo Foundation	223	172	113	0
	Eclipse Foundation AND industry working groups	7	7	4	0
Set 4: User-led OS Consortium or Project Names	Academy Software Foundation	58	42	27	2
	Kuali AND community source	207	23	16	2
	Nordic Institute for Interoperability Solutions	94	70	51	1
	open source AND openMAMA	40	23	19	3
	openKonsequenz	45	25	23	3
	openPASS AND Eclipse Working Group	21	19	15	0
	Sakai AND community source	403	108	48	0
	samvera AND open source AND governance	94	86	66	1
	RESULTS	**2694**	**1784**	**824**	**54**

Set 3: Terms which present umbrella foundations, such as “Eclipse Foundation”, and “Apereo Foundation”
Set 4: Terms which present known examples of user-led OS consortia or their projects such as “Kuali”, “Sakai”, “openKonsequenz”, “openMAMA”, “Academy Software Foundation”, “Nordic Institute for Interoperability Solutions” and “samvera”

We specified our keywords to generate a list of related papers published between 2000 and 2023. We set the initial year for our search as 2000 because the user-led OS consortia phenomenon began with the Sakai project, which was initiated in 2003 through a collaboration among the University of Michigan, Indiana University, Stanford University, and the Massachusetts Institute of Technology in the United States [33]. We conducted our keyword search in the electronic databases of Google Scholar, Web of Science, the ACM Digital Library, IEEE Xplore, and Scopus.

3.2. Search and selection process

To streamline our search process, we defined inclusion and exclusion criteria, and created a data extraction table to document and monitor all obtained results.

We formulated three inclusion criteria based on the publication year, focus, and type of the study to be included. These criteria are as follows:

- **Publication year:** The studies should have been published between 2000 and 2023.
- **Focus:** The study should focus on a user-led open source consortium, a project, or the overall model itself.
- **Type of study:** The study should fall into one of the following categories:
  - Empirical research papers
  - Discussion or opinion papers
  - Experience-sharing papers authored by individuals who are or were participants of any user-led OS consortium or project

We established four exclusion criteria. Any results that meet these criteria are to be eliminated during the selection process. These criteria are as follows:

- **Language:** Studies that are not written in English.
- **Duplicates**: Search results that are duplicates.
- **Non-concurrent manuscripts:** Search results that do not consist of complete manuscripts, such as conference agendas, journal announcements, interview scripts, lecture notes, presentations, or editorials.
- **Student theses:** Bachelor’s theses, master’s theses, and dissertations.

To collect studies published between 2000 and 2023, we searched each defined keyword individually using the specified search engines. For example, we searched for a keyword on Google Scholar and recorded all results by noting the author’s first name, publication year, and the first word of the title. Additionally, we saved the URL of each study for further review. During this process, we identified and marked studies that were not written in English or were duplicates.

After this initial exclusion process, we focused on the type and content of the papers. We reviewed their abstracts and, when necessary, the full manuscripts. In some cases, to better understand the structure of the projects mentioned in the studies and to determine the inclusion or exclusion of these papers, we conducted additional internet searches to gather more information about the projects.

Fig. 1 illustrates the sequential steps of the paper search and selection process, along with the corresponding results.

Our literature search returned a total of 2694 results. We did not restrict our search to any specific section of the studies; instead, we searched "all fields" within the search engines used for the query. The inclusion and exclusion process was conducted through three distinct steps.

In the first step, we excluded search results that were not in English and removed duplicate papers, resulting in a total of 1784 manuscripts.

In the second step, we conducted a thorough content scan and searched for the presence of our designated keywords within the papers. Whenever we found that the keywords did not align with the intended meaning we were seeking, we categorized these papers as "keyword mismatch" and subsequently excluded them from further consideration. For instance, there were cases where the term "community-source" yielded incorrect outcomes, such as references to the "Sun Community Source License." Similarly, when searching for "Sakai," we encountered research papers in which the term appeared as the author's name, despite not being directly related to the Sakai project itself. In this step we excluded 668 papers due to incorrect keyword matches. Furthermore, we eliminated manuscripts that did not meet our inclusion criteria based on the type of study. This entailed excluding materials such as conference agendas, journal announcements, interview scripts, lecture notes, and editorials. Additionally, we made a deliberate decision to exclude bachelor's theses, master's theses, and dissertations. As a result, 292 papers that did not align with the desired study type were eliminated. After this step, we were left with 824 papers.

In the third step, we thoroughly reviewed the titles, abstracts, and bodies of the manuscripts, carefully evaluating their content and its relevance to user-led OS consortia. During this process, we eliminated studies that did not align with the focus of our research. For instance, if a manuscript discussed the use of the "Sakai" quiz tool for student evaluation, we excluded it from our final selection of articles. Another example includes manuscripts that focused on authors' experiences using software developed by a user-led OS consortium or comparing it with other alternatives in the market to decide on implementations; these were categorized as "not on focus". At the end of this step, we excluded 770 studies that did not align with the focus for our research.

Conversely, manuscripts that focused on the creation, governance, or structure of user-led open source consortia or foundations in general, as well as projects associated with these consortia, were labeled as "related" and designated for the quality analysis step. Thus, we collected a total of 54 manuscripts directly relevant to the user-led OS consortia topic.

Table 1 presents the keywords used for the search along with the corresponding results, while the full list of papers is available in ([34]: Appendix A).

3.3. Quality assessment

To assess the quality of the 54 selected papers, we focused on the reporting of results, rigor of the studies, and the credibility of results. We adapted the quality criteria used by Dybå et al. [35] and Kitchenham & Brereton [12] to develop a quality model. We present the list of questions and corresponding answer options we used in our quality model in Table 2.

We began by gaining an understanding of the overall structure of the papers. Our first set of questions concerns defining the research type and determination of the research methods applied in these studies. Dybå et al. [35] excluded the discussion papers from their systematic literature review; however, we did not follow this approach. Like Kitchenham & Brereton [12], we included research papers, discussion papers, and experience papers. Since this research topic impacts both academic research and practical applications in different fields, we did not want to overlook the perspectives of experts involved in user-led open source consortia projects or its implication in the industry. Considering their type, we applied different evaluation criteria to papers in different categories.

The second set of questions includes the criteria for reporting these studies. The third set focuses on the rigor and trustworthiness of results, while the fourth set addresses the credibility of findings. Dybå et al. [35] and Kitchenham & Brereton [12] apply credibility criteria primarily considering quantitative studies. In addition, we included assessment criteria for qualitative studies. However, we did not include "methods' appropriateness to the goals of studies" and "relevance criteria as the value of the study for research or practice", since we aimed to minimize subjective interpretation as much as possible.

Both authors actively reviewed the articles to decide on their inclusion or exclusion. The first author manually evaluated the quality of each article using our quality model. Our primary focus during the quality evaluation of the articles was on the reporting process, the rigor and trustworthiness of findings, and the credibility of results. The first author then used the quality evaluation to suggest a decision regarding inclusion or exclusion.

Our exclusion criteria for manual assessment were as follows:

- If a paper does not provide final results (findings) or only partially provide results, it will be excluded.
- If a research paper does not provide detailed information about the sample AND data collection AND data analysis, it will be excluded.
- If a research paper does not define its research method AND we are unable to determine it, it will be excluded.
- If a research paper has a published extended version with almost the same content, the first version of the paper will be excluded.
- If an experience-sharing paper does not have an author involved in the sample project being discussed, it will be excluded.

The second author developed a scoring function to assign a quality score to each article. The scoring function uses the distance metric (square root of sum of squares) to calculate a quality score over the questions of Q4 to Q13 of the model. We adjusted the cut-off value—-where articles with a higher score were included, articles with a lower score were excluded—based on the maximum alignment with our qualitative assessment. The cut-off value, on the scale of 0 to 1, was set at 0.69.

The second author compared the quality score results with the first author's manual assessment and identified ten problematic cases of disagreement, categorized as follows:

- Articles that were marked for inclusion based on the first author's manual evaluation but received a low score from the scoring function.
- Articles that were marked for exclusion based on the first author's manual evaluation but received a high score from the scoring function.
- Articles that fell into the middle range of 0.65 to 0.75.

The second author reviewed these problematic articles and suggested their re-evaluation. The first and second authors then collaboratively decided on the inclusion or exclusion of each article. As a result, the first and second author created a joint assessment, leading to the inclusion of 42 papers in the systematic review.

We excluded nine articles, since they did not meet our quality criteria. Furthermore, we excluded three articles of good quality that contained similar text to other three articles but were published in different venues with extended content. In these cases, we included only

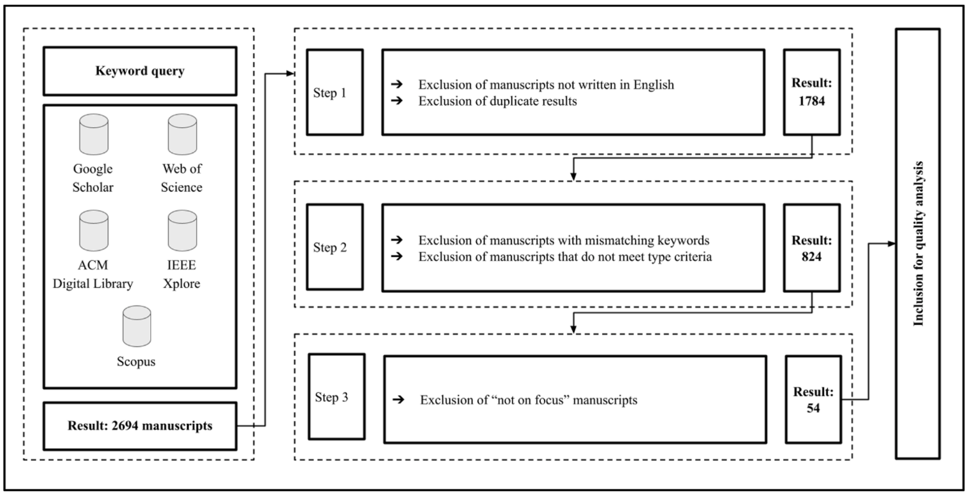

Fig. 1. Search and selection process.

Table 2
Question list for the quality assessment of collected papers.

Category	Number	Question	Answer options & Scoring points
Overview about the study	Q1	What is the type of the study?	Research paper, discussion/opinion paper, experience paper
	Q2	What is the type of research? (applied to research papers)	Qualitative, quantitative, mixed-method, mathematical and simulation models, software tool article, not applicable
	Q3.a	What research method is used in the study? (authors' claim)	Experiment, Quasi-Experiment, Case study, Qualitative survey (Interview Study), Quantitative survey, Grounded theory research, (Systematic) literature review, Observation study, Action research, Experience sharing, Lessons learned, Game-theoretical model, Statistical analysis, Not applicable, None
	Q3.b	What research method is used in the study? (our observation)	Experiment, Quasi-Experiment, Case study, Qualitative survey (Interview Study), Quantitative survey, Grounded theory research, (Systematic) literature review, Observation study, Action research, Experience sharing, Lessons learned, Game-theoretical model, Statistical analysis, Not applicable, None
Reporting	Q4	Is there a clear statement of the aims of the study?	Yes (1) / Partly (0.5) / No (0) / Not applicable
	Q5	Is there an adequate description of the context in which the research or observation was carried out?	Yes (1) / Partly (0.5) / No (0) / Not applicable
	Q6	Is there a clear statement about findings?	Yes (1) / Partly (0.5) / No (0) / Not applicable
Rigor and trustworthiness	Q7	Is the description of the sample and the sample selection process explained in detail?	Yes (1) / Partly (0.5) / No (0) / Not applicable
	Q8	Is the data collection process explained in detail?	Yes (1) / Partly (0.5) / No (0) / Not applicable
	Q9	Is the data analysis process explained in detail?	Yes (1) / Partly (0.5) / No (0) / Not applicable
Credibility	Q10	Is there a limitation or credibility section in the research?	Yes (1) / Partly (0.5) / No (0) / Not applicable
	Q11	If the type of research is qualitative: Are any quality practices such as member checking, prolonged engagement, triangulation, peer debriefing used in the research process?	Yes (1) / Partly (0.5) / No (0) / Not applicable
	Q12	If the type of the research is quantitative: are any methods used to compare results (such as control groups in experiments)?	Yes (1) / Partly (0.5) / No (0) / Not applicable
	Q13	For experience papers: Are one of the authors involved in the discussed project or consortium?	Yes (1) / Partly (0.5) / No (0) / Not applicable

the updated version of each article. Although one paper (e.g., [23]) was published as a report rather than in a traditional research publication, we included it in our research and analysis due to its relevance.

We present the list of included studies in the results section of this paper (in Section 4), and results of our quality assessment in ([34]: Appendix B).

3.4. Snowballing

As the third step in our data collection process, we conducted forward [36] and backward snowballing [37] using the 42 papers resulting from the quality assessment process.

For the forward snowballing process, we used Google Scholar. We searched each of the collected papers and recorded the names and URLs of the papers that cited them using the "cited by" function. This process was repeated for each paper, except for Chesbrough & Appleyard [26]. Since the primary focus of Chesbrough & Appleyard [26] is on "open innovation," it had an overwhelming number of citations (2364). To narrow the search, we employed an automated data collection tool, Publish or Perish,[7] and restricted our search using the keywords "open innovation and strategy" and "self-service." The term "open innovation strategy" is the title of the paper, while "self-service" is the term Chesbrough & Appleyard [26] used to describe the user-led OS consortia concept. After compiling the list of papers and removing duplicates, we obtained a total of 617 papers.

Next, we excluded duplicates, papers not written in English, papers that were inaccessible, papers published after 2023, and papers that did

[7] https://harzing.com/resources/publish-or-perish

not meet our inclusion criteria based on their type. This process left us with 221 papers. We reviewed the abstracts of these papers and, when necessary, the full manuscripts. From this review, we identified three potentially relevant papers; however, these papers did not meet our quality standards based on the selection criteria outlined in Section 3.3. As a result, we were unable to include any additional papers following the forward snowballing analysis.

For the backward snowballing process, we compiled all references listed in the included papers. Nine of the papers did not provide reference lists. From the remaining 33 papers, we gathered a total of 1223 references. After removing duplicates, papers not written in English, inaccessible papers, papers published before 2000, and those that did not meet our inclusion criteria, 740 papers remained for further review. We reviewed the abstracts of these papers and, when necessary, the full manuscripts. From this review, we identified one relevant paper, which was subsequently included in our results.

We provide the list of snowballing results in ([34]: Appendix G).

In Table 3, we provide a comprehensive list of the related literature along with their unique identifiers (IDs) used in this research.

3.5. Data extraction and synthesis of the extracted data

Once we identified the related papers for our research, we extracted and documented key information from each publication, including the "publication title", "authors of the publication", "publication year", "publication type", "published venue", and if applicable, "project studied in the paper". To maintain clarity and traceability, we assigned unique identifiers (IDs) to each paper, which we then utilized in the results section to attribute the extracted information to its respective source.

To address our research questions, we conducted a qualitative data analysis following the six steps of the thematic analysis procedure proposed by Braun & Clarke [13].

In the first step, we familiarized ourselves with the data by taking notes on the content of each paper during the literature selection process. In the second step, we started creating initial codes. Using a qualitative data analysis tool (MaxQDA) ,[8] we generated codes by reading the full manuscripts of each paper. At this stage, we started developing a codebook in a Google Spreadsheet based on the initial codes.

In the third step, we refined our coding scheme by consolidating and clustering the initial codes into sub-themes and main themes. We created a category for codes that did not fit into any of the themes or were not directly related to our research questions, with the intention of revisiting them again. At this point, our approach diverged from Braun and Clarke's [13] methodology. While Braun and Clarke recommend using thematic maps in this step—described as a less detailed but similar alternative to a codebook—we chose to continue developing and refining our codebook instead.

In the fourth step, we revisited each of the relevant papers, carefully examining our codes and their associations with the emerging themes. We continuously revised and updated the codes and themes as needed, eliminating any codes that did not fit into a category or were unrelated to our research questions.

By the time we reached the fifth step, we had developed a clear set of themes and codes. We established precise definitions for the themes and incorporated the most relevant quotes corresponding to each code into the codebook.

The final step of thematic analysis involves producing a report. We present our coding list and themes in our final codebook ([34]: Appendix C) and showcase the results of our data analysis in the Results section of this paper. The process of creating codes and themes, along with supporting examples, is depicted in Fig. 2.

Table 3
Related literature and identifier (IDs) codes.

ID	Reference	ID	Reference	ID	Reference
S1	[38]	S16	[39]	S31	[40]
S2	[41]	S17	[25]	S32	[42]
S3	[43]	S18	[44]	S33	[45]
S4	[46]	S19	[47]	S34	[30]
S5	[26]	S20	[48]	S35	[49]
S6	[23]	S21	[50]	S36	[33]
S7	[51]	S22	[52]	S37	[53]
S8	[54]	S23	[55]	S38	[56]
S9	[57]	S24	[58]	S39	[24]
S10	[59]	S25	[60]	S40	[61]
S11	[62]	S26	[63]	S41	[64]
S12	[65]	S27	[27]	S42	[66]
S13	[67]	S28	[68]	S43	[9]
S14	[69]	S29	[70]		
S15	[71]	S30	[28]		

4. Results

In this section, we present the research findings related to our two research objectives. Each objective will be further explored through subsections addressing related research questions.

4.1. RO. 1. Identification of the state of the art in user-led open source consortia literature

As a result of our literature search and selection process, we collected 43 related papers. We present a sample of search results in Table 4, and the full list in ([34]: Appendix D). Descriptive statistics on the distribution of the included studies are provided in Section 4.1.1, details on user-led OS projects investigated in the literature are discussed in Section 4.1.2, and key concepts explored in the literature are outlined in Section 4.1.3.

4.1.1. RQ.1.1. What is the current state of the art in the literature on user-led open source consortia?

To address RQ1.1, we analyzed the distribution of studies focusing on three aspects: study type, publication venue, and publication year.

Among the 43 manuscripts we collected, 19 (44 %) are peer-reviewed research papers, 13 (30 %) are experience papers, and 11 (26 %) are opinion papers. Fig. 3 illustrates the type distribution of the collected studies.

Studies focusing on user-led OS consortia were published in various venues. Of the 43 studies collected in this research, 18 (42 %) were published in journals, 13 (30 %) in conference proceedings (including congress and symposiums), and eight (19 %) in industry specific magazines. Furthermore, we included in our analysis three book chapters and one report (9 %).

The majority of the *research papers* (17 out of 19) were published in journals and conference proceedings. The remaining two research papers were published in Communications of the ACM Magazine and Organization for Open-Source Software Study Report. *Experience papers* were written by founders or members of different user-led OS consortia. Of these, 69 % were published in journals and conference proceedings, while the remaining 31 % appeared in magazines and books. *Discussion (or opinion) papers* focused on the general structure of user-led OS consortia. The authors of these papers were either founders of different user-led OS consortia or sector professionals. Of the discussion papers, 45 % were published in industry-specific magazines, while the remaining papers were published in journals, conference proceedings, and one book.

We present the type and publication venue distribution of the studies we collected in Fig. 4.

[8] https://www.maxqda.com/

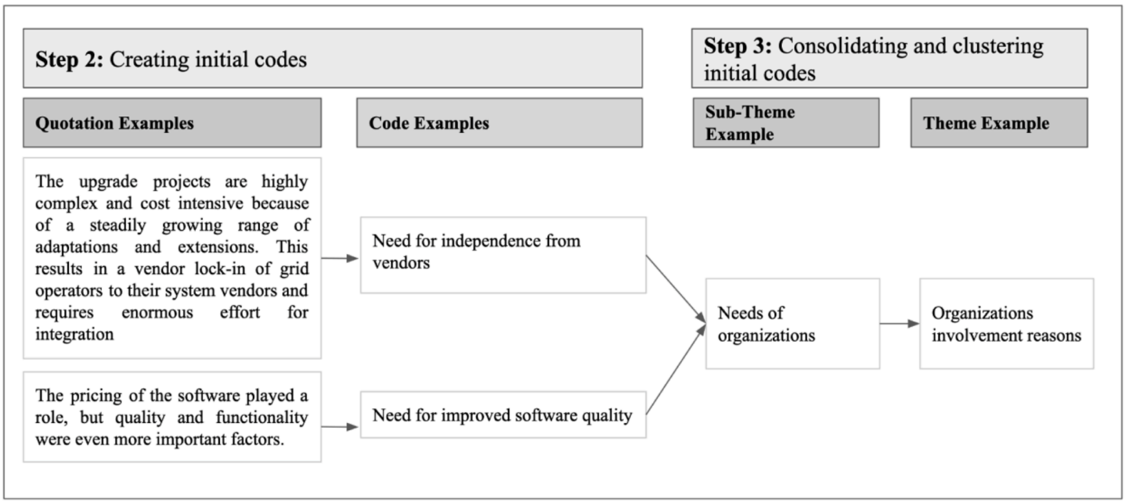

Fig. 2. Steps 2 and 3 of the data analysis process.

Table 4
Sample search results examining the topic of user-led open source consortia.

ID	Title	Author(s)	Year	Type	Venue	Sample	Industry
S10	Tapestries of Innovation: Structures of Contemporary Open Source Project Engagements	Germonprez, M., Levy, M., Kendall, J. E., & Kendall, K. E.	2020	Research paper	Journal of the Association for Information Systems	open-MAMA	Finance

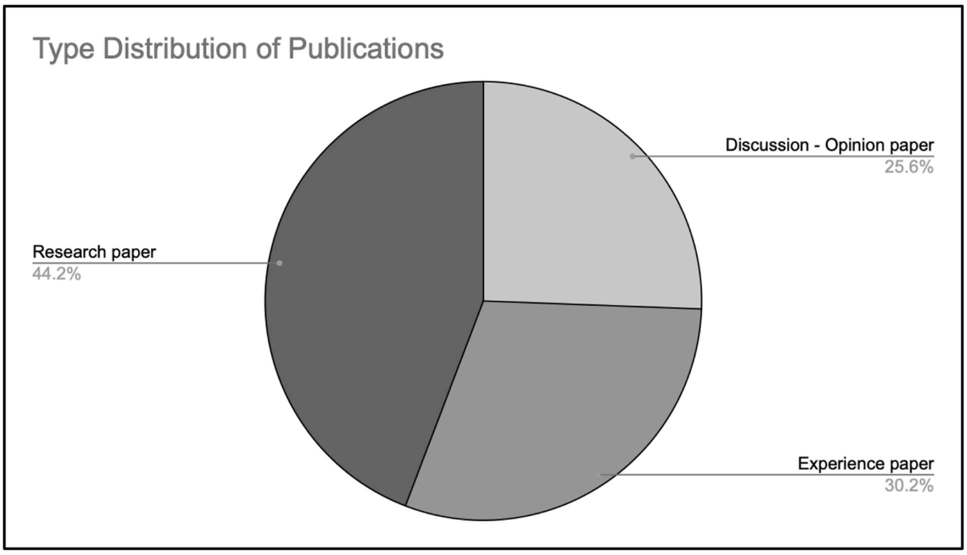

Fig. 3. Distribution of study types in user-led open source consortia research.

The first examples of user-led OS consortia projects were initiated at the beginning of 2000s by higher education institutions. From 2004 to 2023, 67 % of the published papers had a focus on user-led OS consortia or projects from higher education. The first paper in a different industry beyond higher education was published in 2013. From 2013 to 2023, 29 % of papers were focused on consortia from other industries beyond higher education. A further 4 % of the papers explained the general structure of the user-led OS consortia without focusing on any industry. The years 2007, 2010, and 2020 stand out as being particularly significant in terms of numbers of papers published. Fig. 5 illustrates the distribution of the literature based on the published year.

4.1.2. RQ.1.2. Which user-led open source consortia and projects have been investigated in the literature?

Nearly half (43 %) of the identified projects and their associated consortia in the literature originate from the higher education industry. The Kuali Foundation and the Apereo Foundation are two umbrella organizations that host user-led open-source (OS) projects in higher education.

The Kuali Foundation was established in 2004 by a group of universities and colleges in the United States [S8]. Its initial focus was to ensure financial sustainability and coordination for the Kuali Financial Systems (KFS) project, a user-led open-source consortium project [S8]. Following the success of KFS, the Foundation expanded its open-source initiatives to include a research administration system (Kuali Coeus), a student information system (Kuali Student), a library system (Kuali

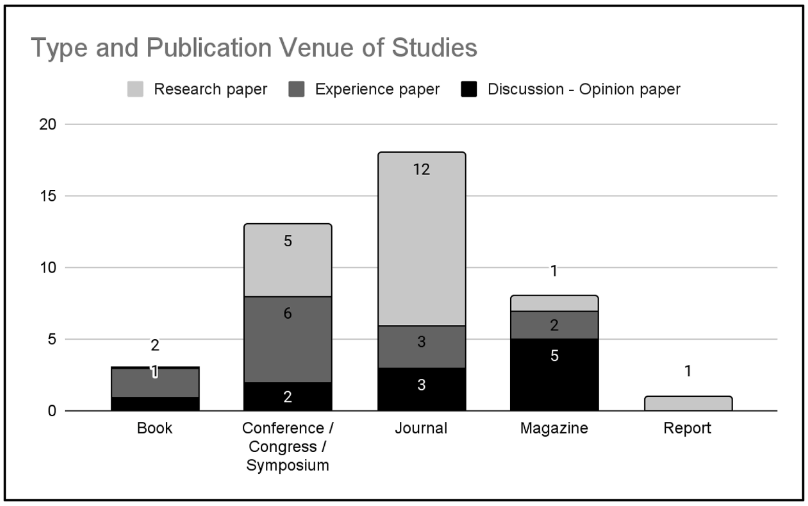

Fig. 4. Distribution of study types and publication venues in user-led open source consortia research.

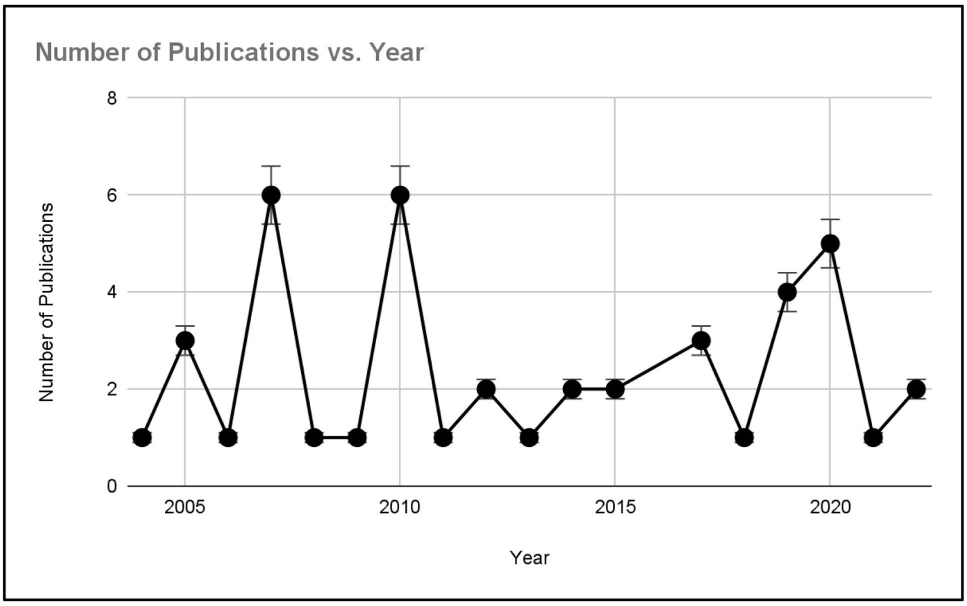

Fig. 5. Distribution of publication years in user-led open source consortia research.

OLE), and middleware applications (Rice) [S8]. In 2014, the Kuali Foundation created a for-profit company, KualiCo, with the goal of "sustaining community" [S4]. KualiCo focuses on a cloud-based software-as-a-service (SaaS) model, offering paid cloud services for Kuali products [S4]. As of September 2023, the Kuali Foundation has been dissolved as a legal entity and transitioned into a for-profit Kuali company [72].

The Apereo Foundation was established in 2012 through the merger of two organizations: Ja-Sig and the Sakai Foundation [73,74]. The Sakai Foundation, incorporated in 2005, aimed to sustain the Sakai Learning Management System (LMS) project and its community [S35, S36, S40]. Following the merger, ownership of the Sakai LMS project was transferred to the Apereo Foundation. In addition to Sakai, the Apereo Foundation also hosts other projects discussed in the literature, including Open Source Portfolio (OSP) and OpenCast. ePresence was an in-house developed streaming tool at the University of Toronto. In 2005, the university decided to open-source the project and create a consortium around it [S29]. Initially, the consortium followed a dual-license approach. After two years (in 2007), the members decided to adopt a single open-source license (BSD) for all versions [S29]. However, the open-source version of ePresence is no longer actively maintained.

Two projects, FOLIO and Hyku for Consortia, originate from the library industry. FOLIO was hosted by the Kuali Foundation from 2010 to 2016, during which it was known as the Kuali OLE project. In 2016, following the establishment of KualiCo, the members of the Kuali OLE project decided to leave the foundation and started a non-profit organization: the Open Library Foundation (OLF) [S42]. Since then, the FOLIO project has been hosted by the OLF.

Hyku for Consortia is a collaborative project initiated in 2018 by the Pennsylvania Academic Library Consortium (PALC) and the Private Academic Library Network of Indiana (PALNI). The project aims to build an open-source institutional repository (IR) on the Hyku platform and make it available for use by libraries [S25].

Other projects and consortia discussed in the literature originate from the automotive, energy, finance, entertainment, and geospatial industries. openMDM is a consortium of companies in the automotive industry. Initiated by Audi in 2012, it became an open-source consortium in 2014 under the umbrella of the Eclipse Foundation [S43]. openKonsequenz is a consortium of energy providers focused on software development for energy grid operation management. It was initiated in 2013 in Germany [S11]. openMAMA is a consortium in the finance industry. Its main project is a data transfer platform that supports standardized data formats and is used by financial institutions [S9].

The Academy Software Foundation (ASWF) is a consortium of motion picture and visual effects organizations. Established in 2018, its goal is to support OSS development within the motion picture content creation industry [S13]. OpenColorIO is one of the projects hosted by ASWF [S37]. As of August 2023, both openMAMA and the Academy Software Foundation are hosted under the umbrella of the Linux Foundation.

Two projects, Oskari and X-Road, are publicly funded and led by governmental organizations. Oskari is a geospatial software project initiated by the National Land Survey of Finland (NLSF) in 2008 [S14, [75]]. In 2011, the NLSF released its source code openly, and in 2014, a community of organizations called the Joint Development Group was established around this project (Oskari, n.d.). Since 2017, Oskari has been an incubation project under the umbrella foundation of OSGeo [S14, Oskari, n.d.].

X-Road is a data exchange platform developed by the Nordic Institute for Interoperability Solutions (NIIS), a consortium formed by governmental organizations from Estonia and Finland [S31]. X-Road is used in both the public and private sectors. In the public sector, it supports systems such as population registers, health insurance registers, and vehicle registration systems. In the private sector, it is utilized by energy, telecom, and banking institutions [S31].

We present the sample distribution of user-led OS consortia studies in a matrix format in Table 5.

Beyond extracting data from the literature, we conducted an online search to gather additional information about these projects. We focused on the use cases, industry, foundations, and active years of these projects. We present the details of each project in Table 6.

It is noteworthy that we identified numerous additional projects developed by user-led OS consortia; however, no existing studies were found about these consortia.

4.1.3. RQ.1.3. Which research topics about user-led open source consortia does the literature address?

To address RQ.1.3, we conducted thematic analysis [13] and applied the concept-matrix approach for presentation [76]. We synthesized the individual topics from primary studies into a concept hierarchy across all studies. The top-level (root) concepts are referred to as key concepts. We identified five key concepts across all studies, which are:

1. General structure of user-led OS consortia
2. Governance of user-led OS consortia
3. Ecosystems of user-led OS consortia
4. Creation of a specific user-led OS consortium
5. Development process of a specific user-led OS project

In this section, we provide detailed explanations of each concept, along with the studies classified under these concept categories. The concept matrix is presented in Table 7, and the distribution of subconcepts shown in Table 8.

4.1.3.1. General structure of user-led open source consortia. In this category, we collected studies focusing on the overall structure of the user-led OS consortia model and its impact across different areas.

User-led OS consortia concept is explained as a category of business model enabled by OSS [S5, S27, S30]. This model is defined as “consortia of end-user organizations jointly developing applications to be used by all” [S5, S27]. Since the users of the software develop it to meet their own needs, it is referred to as “self-service” [S5]. The goal is to create value through shared resources and to increase flexibility and innovation potential [S19]. Targeting vertical, enterprise, or back-office applications, this model focuses on a specific market segment [S27]. The user-led OS consortia concept represents one of the innovations of open source in the business model category [S30]. It is a type of open source foundation, which could be referred to as a “user foundation” [S30].

The user-led OSS development approach provides enrichments to involved institutions by shared experiences and best practices [S3]. Key components of user-led OS projects, particularly in the educational sector, include finding stakeholders which have similar problems to solve, identifying development partners which would provide technical support [S3, S26], securing financial support for the projects [S3, S26], coordinating resources [S26], and ensuring institutional commitment to the sustainability of the projects [S24]. Institutional involvement in user-led OS consortia projects requires aligned goals, sufficient resources, and shared values [S41].

The collaborative OSS development approach among for-profit companies enables them to spare time and resources in developing basic functionalities. This allows them to focus on differentiating their products based on feasibility, unique features, and advancing their own strategy [S9, S10, S16]. Open source communities offer a platform for strategic innovation to for-profit companies [S10, S16].

User-led OS consortia model originated in higher education under the name “community source”. Studies examining the general structure of this model, particularly in the context of higher education, are primarily experience or opinion papers authored by university members.

International collaboration and OSS movement play a significant role in resource creation for higher education [S2]. Various models and technologies have been applied in the Information Technology (IT) education community [S32]. One such approach is the user-led OS consortia model, with early examples including the Kuali and Sakai projects [S32, S39]. Contributions from user organizations to OSS projects are considered more reliable for mission critical projects and complex systems [S6]. Collaborating on user-led OS consortia projects helps mitigate the risks associated with proprietary software systems in higher education institutions [S38, S39]. These risks include misalignment with the institution’s operating model, implementation complexities, high implementation costs, and dependency on vendor behavior [S38, S39].

4.1.3.2. Governance of user-led open source consortia. We classified studies that focus on governance policies, governance practices, challenges and solutions of user-led OS consortia under the category of “governance of user-led OS consortia”.

The challenges faced by the user-led OS consortia projects include managing developers, finding high-quality developers, high turnover rates among developers, and ensuring the sustainability of projects [S17]. “Outsourcing developers” is one proposed solution to these challenges [S17].

When outsourcing the development process, a potential problem is the division of software development responsibilities among different vendors without a consortium-wide authority [S43]. To address this issue, the development process should be monitored through regular assessments and clearly defined milestones. Additionally, a dedicated project manager, supported by a consistent team of developers, should oversee the process, to ensure its success [S43].

Some of the other challenges are a low number of leading members, insufficient financial resources to sustain projects, delayed project releases, slow return on investment, turnover among service provider members and knowledge loss, a small user base, and lack of awareness about the projects [S43]. On the other hand, factors that can help overcome these challenges and achieve success include having clearly defined rules and boundaries, collective prioritization, openness and transparency, shared resources and equality, member commitment, established governance rules and legal structures, periodic

Table 5
Sample distribution of user-led open source consortia studies.

	Umbrella Foundation	Apereo Foundation			Kuali Foundation			Open Library Foundation	Hyku Partners	Eclipse Foundation	openKonsequenz	Linux Foundation	ASWF		Oskari Community	NIIS	None	
	Focus (Projects or General)	Sakai LMS	OpenCast	OSP	Kuali Financial Systems	Kuali Rice	Kuali Foundation	OLE project - FOLIO	Hyku for consortia	openMDM	openKonsequenz	openMAMA	OpenColorIO	ASWF	Oskari	X-Road	ePresence	General
ID	**Study**																	
S1	[38]																x	
S2	[41]	x																
S3	[43]																	x
S4	[46]						x											
S5	[26]																	x
S6	[23]																	x
S7	[51]	x																
S8	[54]						x											
S9	[57]											x						
S10	[59]											x						
S11	[62]										x							
S12	[65]		x															
S13	[67]													x				
S14	[69]														x			
S15	[71]		x															
S16	[39]											x						
S17	[25]				x													
S18	[44]	x				x												
S19	[47]						x											
S20	[48]						x											
S21	[50]						x											
S22	[52]						x											
S23	[55]						x											
S24	[58]																	x
S25	[60]								x									
S26	[63]	x		x														x
S27	[27]																	x
S28	[68]				x													
S29	[70]																x	
S30	[28]																	x
S31	[40]															x		
S32	[42]	x			x													
S33	[45]					x												
S34	[30]										x							
S35	[49]	x																
S36	[33]	x																
S37	[53]												x					
S38	[56]																	x
S39	[24]	x		x	x		x											x
S40	[61]	x																
S41	[64]																	x
S42	[66]							x										
S43	[9]	x								x								

Table 6
User-led open source consortia projects identified in the literature.

Project Name	Industry	Goal	Initiation Year	Status in May 2024	Foundation/ Initiative
Sakai LMS Project	Higher Education	Developing an online collaboration and learning environment for managing, delivering, and assessing student learning	2003	Continues as a user-led OS project	Apereo Foundation
Open Source Portfolio	Higher Education	Developing an online e-portfolio for personal representation, teaching, learning, assessment and accreditation	2003	Since 2005 it is a part of Sakai Project	Apereo Foundation
Kuali Financial Systems Project	Higher Education	Developing a financial services system specifically for colleges and universities	2004	Since 2014 it is a commercial product	Kuali Company (prior structure: Foundation)
ePresence	Higher Education	Developing a web-based streaming and collaboration tool for large-scale broadcast of events over the Internet	2005	inactive	None
Opencast (prior name: Opencast Matterhorn)	Higher Education	Developing an open source video recording and management system to use for lectures	2008	Continues as a user-led OS project	Apereo Foundation
openMAMA	Finance	Building an open platform to publish market data from multiple sources and multiple vendors in a standardized format	2010	Continues as a user-led OS project	Linux Foundation
Oskari	Geospatial	Developing a software to view, visualize, analyze and edit spatial data	2011	Continued as a user-led OS project	Oskari Joint Development Forum
openKonsequenz	Energy	Building software systems that are used in energy grid operation management	2013	Continue as a user-led OS project	openKonsequenz Cooperative
openMDM	Automotive		2014	Continue as a user-led OS project	Eclipse Foundation
FOLIO (prior name:Kuali OLE)	Library	Developing an open source platform for libraries	2016	Continues as a user-led OS project	Open Library Foundation
X-road	Neutral	Building a data exchange layer solution which ensures confidentiality, integrity and interoperability between data exchange parties	2017	Continues as a user-led OS project	Nordic Institute for Interoperability Solutions (NIIS)
OpenColorIO	Entertainment	Setting standards for color management in visual effects industry	2018	Continues as a user-led OS project	Academy Software Foundation (ASWF)
Hyku for Consortia	Library	Building a collaborative institutional repository based on Hyku Software	2019	Continues as a user-led OS project	Hyku Partners

communication, organizing events, and promoting hosted projects [S43].

Another challenge is sustaining continuity in such collaborations [S22]. Specific challenges in this area include community governance, defining the roles of commercial affiliates, maintaining a family-like atmosphere, sharing cross-project knowledge, and coordinating projects [S22]. One proposed solution to these issues is implementing a modular organizational design [S22, S23].

Addressing the diverse requirements of member organizations poses another challenge. Proposed solutions include achieving technological flexibility and customization [S18]. For example, in the Kuali Rice System project, five levels of customization are implemented: label customization, modification and addition of document types, workflow customization, code modification, and the addition of new modules [S18]. Similarly, in the Sakai project, the architecture is designed to enable both flexibility for use and flexibility for development [S18].

Focusing on the Kuali example, three phases of the governance process have been identified: 1) creating the community, 2) balancing the interests, and 3) sustaining the community [S4]. In the first phase (2004–2006), the focus was on supporting the OSS development process. During the second phase (2006–2014), the foundation prioritized meeting the diverse interests of stakeholders by creating customizable features. In the third phase (since 2014), the governance strategy shifted toward establishing a commercial company, KualiCo, to offer paid cloud-based services. With this strategy, the Kuali community aimed to create a hybrid model combining OSS principles with commercial market concepts [S4].

To address management challenges faced by OSS consortia in public sector organizations, a framework for community-based lifecycle planning has been proposed [S14]. This framework offers guidance on defining what needs to be managed, who should manage it, how it should be managed, and how to finance the management and development process. Implementing this framework impacts product acceptance and quality, resource pooling, and project sustainability processes [S14].

4.1.3.3. Ecosystem of user-led open source consortia. Papers discussing topics related to the actors in user-led OS consortia and their relationships are categorized under the "ecosystem of user-led OS consortia" category.

The ecosystem of the openKonsequenz consortium is examined in the literature [S34]. This consortium comprises three types of members. The first type includes energy company providers, specifically distribution system operators, who take the lead role in driving the development direction. These members provide the necessary financial investment and human resources and are referred to as the driver members. The second type consists of software vendors, who contribute to the development process with the long-term goal of spreading their technology and strengthening their future market position. The third type includes consultants, who aim to profit from consulting projects, and research groups, which benefit from the data generated by the project [S34].

Motivations for user organizations to participate in OSS consortia include cost reduction, independence from vendors, and options for system customization [S19, S34]. Additionally, developer training—both in terms of gaining system expertise and building strong social bonds through community involvement—is another key incentive [S19]. The size, financial power, and IT capabilities of individual institutions significantly influence decisions to join such consortia [S20]. Other institutional factors impacting these decisions include established norms, monitoring mechanisms, institutional similarity, availability of external funding, vendor behaviors, and the role of information technology [S21]. Furthermore, individual factors such as personal motives, opportunities for learning, and levels of trust have influence on decision-makers within institutions [S21].

Another topic explored in the literature is the interaction between participants in multi-organizational software development consortia [e.g., S28, S33]. Participants who supervise processes or provide functional advice tend to have the highest positional embeddedness, which correlates with the time they dedicate weekly and their level of influence [S28]. Those who invest significant hours in the projects gain recognition from others, enhancing their embeddedness within the network.

Table 7
Distribution of research concepts on user-led open source consortia addressed in the literature.

ID	Reference	Study type	Key Concepts General structure of user-led OS consortia	Governance of user-led OS consortia	Ecosystem of user-led OS consortia	Creation of a specific user-led OS consortium / foundation	Development process of a specific user-led OS project
S1	[38]	Experience paper					x
S2	[41]	Discussion - Opinion paper	x				
S3	[43]	Discussion - Opinion paper	x				
S4	[46]	Research paper		x			
S5	[26]	Discussion - Opinion paper	x				
S6	[23]	Research paper	x				
S7	[51]	Experience paper					x
S8	[54]	Experience paper				x	
S9	[57]	Discussion - Opinion paper	x				
S10	[59]	Research paper	x				
S11	[62]	Research paper					x
S12	[65]	Experience paper					x
S13	[67]	Experience paper				x	
S14	[69]	Research paper		x			
S15	[71]	Experience paper					x
S16	[39]	Research paper	x				
S17	[25]	Research paper		x			
S18	[44]	Research paper		x			
S19	[47]	Research paper			x		
S20	[48]	Research paper			x		
S21	[50]	Research paper			x		
S22	[52]	Research paper		x			
S23	[55]	Research paper		x			
S24	[58]	Discussion - Opinion paper	x				
S25	[60]	Experience paper					x
S26	[63]	Discussion - Opinion paper	x				
S27	[27]	Research paper	x				
S28	[68]	Research paper			x		
S29	[70]	Experience paper					x
S30	[28]	Discussion - Opinion paper	x				
S31	[40]	Research paper					x
S32	[42]	Discussion - Opinion paper	x				
S33	[45]	Research paper			x		
S34	[30]	Research paper			x		
S35	[49]	Experience paper					x
S36	[33]	Experience paper					x
S37	[53]	Research paper					x
S38	[56]	Discussion - Opinion paper	x				
S39	[24]	Discussion - Opinion paper	x				
S40	[61]	Experience paper					x
S41	[64]	Discussion - Opinion paper	x				
S42	[66]	Experience paper					x
S43	[9]	Research paper		x			

Strong connections within the network are advantageous, as they increase participants' ability to influence project outcomes. Additionally, participants seeking greater involvement in projects often choose to collaborate with individuals who hold influential roles in project decision-making [S28].

Project participants who work within the same organization tend to communicate more frequently compared to those from different organizations [S28, S33]. While this tendency can strengthen intra-organizational collaboration, it may also limit knowledge flow and reduce overall project effectiveness [S28]. Additionally, factors such as task assignments, clarity regarding task timelines, and the criticality of tasks significantly influence developers' actions and their interactions with one another [S33].

4.1.3.4. Creation of a specific user-led open source consortium. We categorized papers primarily focused on explaining the functionalities of user-led OS consortia (foundations) under the "creation of a specific user-led OS consortium" category. This category includes two papers.

The first, authored by the executive director of the Kuali Foundation in 2010, discusses the status of the foundation as of that year [S8]. The second paper examines the structure of the Technical Advisory Board of the Academy Software Foundation (ASWF) and outlines its goals for 2018–2019 [S13].

4.1.3.5. Development process of a specific user-led open source consortium project. We categorized papers discussing the initiation and development processes of specific user-led OS projects under the "development process of a specific user-led OS project" category. Most of these papers

Table 8
Distribution of subconcepts on user-led open source consortia addressed in the literature.

Key Concepts	Subconcepts	Literature ID	Count
General structure of user-led OS consortia	Business models in OSS development	S5, S27, S30	3
	Collaboration of organizations in OSS development	S9, S10, S16	3
	Values/ Advantages of community source development	S3, S38	2
	Community source model in higher education	S2, S6, S24, S26, S32, S39, S41	7
Governance of user-led OS consortia	Problems and solutions	S17, S22, S43	3
	Achieving goals and sustainability	S18, S23	2
	Governance practices	S4	1
	Lifecycle management	S14	1
Ecosystems of user-led OS consortia	Ecosystem of a user-led OS consortium	S34	1
	Motivations to join community source projects	S19, S20, S21	3
	Interaction between participants in user-led OS projects	S28, S33	2
Creation of a specific user-led OS consortium / foundation	ASWF	S13	1
	Kuali Foundation	S8	1
Development of a specific user-led OS project	ePresence	S1, S29	2
	Hyku for consortia	S25	1
	OLE project	S42	1
	OpenCast	S12, S15	2
	OpenColorIO	S37	1
	openKonsequenz platform	S11	1
	Sakai LMS	S7, S35, S36, S40	4
	X-Road	S31	1

are based on the experiences of the initiators of these projects. The projects examined include ePresence, Sakai LMS, Open Library Environment, X-Road, Hyku for Consortia, OpenCast Matterhorn, OpenColorIO, and openKonsequenz. ePresence was initially developed as an in-house streaming tool by the University of Toronto. In 2005, the university decided to open-source the project and establish a consortium around it [S29]. The primary motivation for this decision was to provide users with the flexibility to tailor the system to their specific needs. The project faced several challenges, including developing high-quality software with distributed development teams, sustaining an active community, identifying a suitable license, and establishing a revenue model [S1]. To address the challenge of generating revenue from the OSS product, the consortium adopted a "dual license" approach. However, this strategy proved unsuccessful. A major issue with the dual-license model was the need to maintain two separate but interrelated software packages, requiring the consortium to duplicate efforts during each release process. This approach not only consumed significant time but also led to usability problems [S29]. Ultimately, the consortium decided to simplify the licensing strategy by offering ePresence under a single open-source license, the BSD license [S29].

The history of the Sakai LMS project, along with its governance structure and the challenges faced during its establishment, is detailed by the project's founders [S35, S36, S40].

The history of the Open Library Environment (OLE) project and its transformation into the FOLIO project is documented in the literature [S42]. From 2010 to 2016, the OLE project was hosted by the Kuali Foundation. However, in 2015, OLE partners decided to leave the foundation and establish their own independent foundation. The primary reason for this decision was the shift in the Kuali Foundation's open-source policies. In 2014, the foundation created a for-profit corporation, KualiCo, to act as a service provider for the open-source products developed under the Kuali Foundation. Another contributing factor was the Kuali Foundation's decision to discontinue support for the Kuali Rice component, which served as the foundational framework for Kuali OLE [S42].

The literature provides insights into various aspects of notable user-led OS projects, including the organizational structure, contributors, and stakeholder-perceived challenges of the X-Road project [S31]; the creation and management process of the Hyku for Consortia project [S25]; the history and technical specifications of the Opencast Matterhorn project [S12, S15]; the details of the OpenColorIO project [S37], and development process and reference architecture of the openKonsequenz platform [S11].

4.2. RO.2. Identification of the structure of user-led open source consortia and motivations of organizations for participation

To explore our second research objective, we formulated three research questions and conducted thematic analysis to address them.

To address RQ.2.1, we searched for defining characteristics of user-led OS consortia and categorized them in three main themes, which are presented in Section 4.2.1. We addressed RQ.2.2 by identifying the actors involved in user-led OS consortia, along with their roles, and goals. We explain our findings in Section 4.2.2. To address RQ.2.3, we examined the benefits of involvement, with the results presented in Section 4.2.3.

4.2.1. RQ.2.1. What are the defining characteristics of user-led open source consortia?

A user-led open source consortium is a community of user organizations and software vendors, working collaboratively to develop OSS for the specific needs of user organizations.

As a result of our thematic analysis, we identified key characteristics of user-led OS consortia, focusing on three dimensions: governance, goal, and work result. Within the governance dimension, we identified the theme of "being led by user organizations". The goal of the consortia is "collaborative software development" to meet their own needs, and their work results in "open-source software".

In this section, we explain the details of each characteristic and present a summary in Fig. 6. The mapping of each characteristic with data sources and related user-led OS consortia is provided in ([34]: Appendix E).

4.2.1.1. Governance: led by user organizations. User-led OS consortia are **initiated, financed, and led by user organizations,** whose primary function is not software development. However, these organizations engage in software development efforts to support their internal processes [S2, S5, S6, S8, S11, S16, S18, S19, S25, S34, S38, S42, S43]. For example, two universities in the USA—Indiana University and the University of Hawaii—initiated the Kuali Financial Systems project. Their goal was to develop software to support their internal finance management processes. With the involvement of other universities with similar needs, they built a consortium around this project [S18].

In user-led OS consortia, user organizations are the drivers of the software development process. They **define requirements** [S2, S6, S11, S16, S21, S25, S31, S43] and **steer the development direction** [S2, S11, S14, S16, S19, S24, S31, S38, S43]. Although external funding options may be available in some cases, the development process is primarily **financed by member organizations** [S8, S14, S15, S25, S26, S31, S34, S40]. In contrast to software vendor companies, user organizations do not focus on developing generic software, but tailor-made **enterprise applications** which are required to fill functionality gaps in related industries [S18].

4.2.1.2. Goal: collaborative software development. User-led OS consortia

	Governance: Led by users organizations	Goal: Collaborative software development	Work result: Open-source software
Consortium structure	➔ Initiated by user organizations ➔ Financed by member organizations	➔ Consist of at least two organizations ➔ Built with shared resources ➔ Formal virtual organizations	
Members	➔ User organizations are driver members ➔ Software vendors are development members	➔ Share the same vision ➔ Have contractual agreements ➔ Have clear roles and responsibilities	
Development process	➔ Requirements defined by user members ➔ Development direction steered by user members	➔ Distributed software development approach	➔ Open to contributions from the community and other user organizations
Developers		➔ Employees of member organizations	➔ Volunteering at organizational level
Software	➔ Focuses on enterprise applications	➔ Has initial closed development stage or initial code	➔ Does not provide a competitive advantage

Fig. 6. Defining characteristics of user-led open source consortia.

consist of multiple partner organizations [S6, S11, S17, S18, S19, S25, S26, S32, S35, S37, S39, S40, S42, S43]. Organizations collaborate by **focusing on a common goal** and sharing the same vision about it [S1, S12, S19, S23, S24, S26, S34, S37, S40, S42, S43]. To ensure the sustainability of projects, partner organizations create non-profit legal entities. This can be a consortium, a foundation, or an initiation. These legal entities may have different legal structures based on the country they are initiated, but from the governance perspective, they follow similar approaches. They create the boundaries and rules of collaborative working. A key principle in establishing the consortium is the use of **formal, contractual agreements** [S14, S25, S28, S32, S41, S43]. In these formal agreements—most often in the form of consortium charters—**membership structures, roles, and responsibilities** of member organizations are outlined. All partners are required to sign these agreements to join the consortium [S14, S18, S22, S39, S43].

Member organizations collaboratively focus on OSS development by **sharing resources** [S5, S6, S12, S14, S17, S18, S19, S20, S21, S22, S25, S27, S34, S39, S43]. In user-led OS consortia projects, the majority of project participants are **employees of member organizations** [S11, S16, S17, S18, S19, S20, S22, S24, S25, S26, S28, S33, S35, S39, S42, S43]. They can be employees of user organizations, development partners within the consortia, or both [S11, S16, S18, S25, S26, S35, S39, S40, S42, S43]. In some cases, project management and coordination tasks are executed by paid staff which are employed by the legal entities, such as foundations [S14, S35, S42, S43].

User-led OS consortia are built as **virtual organizations** [S14, S17, S18, S19, S20, S22, S24, S28] that follow **distributed software development** approaches [S8, S18, S28, S33, S39].

4.2.1.3. Work result: open-source software. Work results of user-led OS consortia projects are open-source software. However, in most cases these projects have an initial **closed code development stage**. User-led OS consortia can evolve around an already developed closed-source software such as in the examples of openMAMA, openMDM, and openColorIO [S16, S37, S43]. The other approach is that partner companies can work to develop the software initially at a closed stage such as in the case of the Sakai project [S36]. In these cases, after software reaches a satisfactory maturity level, it is offered as OSS for the use and contribution of other organizations [S24, S28, S32, S39]. The resulting software **does not provide a competitive advantage or differentiation** for the user member organizations [S3, S6, S9, S16, S21, S34].

OSS offers **flexibility in adaptation** for users [S18, S31, S35]. User organizations can adjust the functionality of the software based on their specific needs. Although the software is **open to any contributions** from individuals or organizations [S16, S24, S31, S32, S37], the **volunteering mostly takes place at the organizational level** [S16, S24, S26, S39].

4.2.2. RQ.2.2. How do organizations engage with user-led open source consortia?

To understand the organizations' engagement with user-led OS consortia, we identified the key actors and their roles within these environments. Based on our thematic analysis, we grouped the key actors into five categories: driver members, development partners (software vendors), user (adopter) members, non-profit organizations, and a legal entity—most often a foundation. In this section, we explain the roles of each actor and present the summary in Figs. 7 and 8.

4.2.2.1. Driver members. Driver members (or organizations) are primarily **user organizations** that need software with specific requirements to fulfill their internal business processes [S8]. The end users of the required software are stakeholders within these organizations, who are typically average computer users rather than software developers [S26].

Driver members engage with user-led OS consortia by financing software projects through **monetary support** and/or **staff resources** [S18, S21, S22, S33, S34, S35, S42, S43]. They define technical requirements [S11, S15] and influence the direction of the software development [S11, S21, S19, S14, S28, S33].

4.2.2.2. Development partners. Development partners (or members) are **software suppliers or vendors** that are involved in consortia [S4, S21, S31, S42, S43]. They work on the software development process based on the requirements defined by driver members [S11, S16, S18, S35, S39, S40]. Since they have expertise in software development, they ease the development process [S3].

Development partners engage with user-led OS consortia by paying a membership fee or offering manpower to the consortia for the development work. In return for their contributions, development partners anticipate providing user organizations with **complementary fee-based services** for software implementation [S3, S8, S21, S22, S28, S31, S40]. Being a development partner in a user-led OS consortium

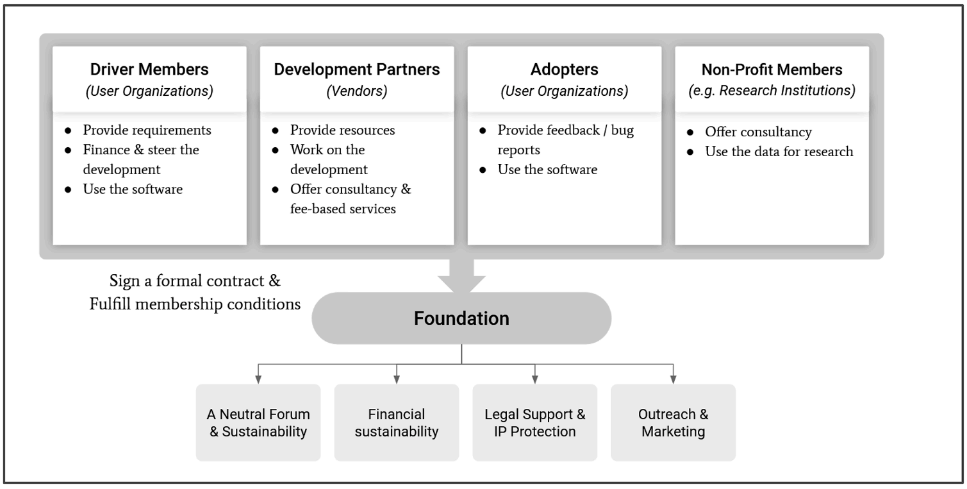

Fig. 7. Actors and their roles in a user-led open source consortium.

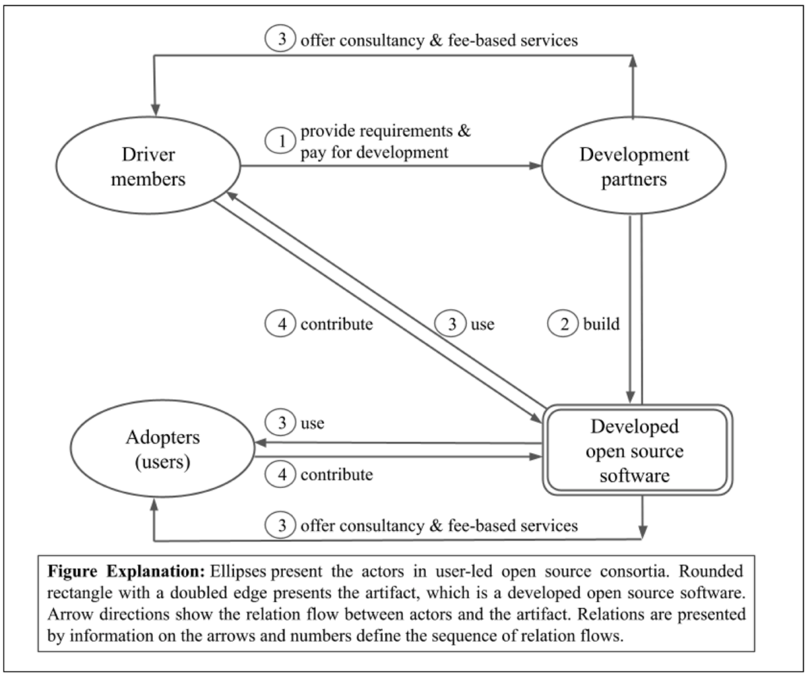

Fig. 8. Relationships among actors in a user-led open source consortium.

enhances suppliers' reputation as **a trusted vendor** and enables them to **spread their technology** in the market [S34].

4.2.2.3. Adopters (User members). **Adopters** or user members are user organizations that **do not have any direct influence on the development process**, but use the software developed [S11, S34, S35, S43]. Like other member organizations, they sign a contractual agreement, but they do not financially contribute to the project as much as driver members. They can contribute projects by providing feedback, bug reports or merging additional functionalities they create for their own needs [S11, S35]. Having user organizations **increases the use of the platform** and helps consortia to **set standards** in the industry [S34].

4.2.2.4. Non-profit members. Non-profit members include **research organizations** or **universities** (not user organizations) working with the consortium [S11, S43]. They can contribute to the project and benefit from the research data [S34].

4.2.2.5. Foundation. User-led OS collaboration members either build their own foundations or join an already established umbrella foundation.

Foundations offer **independent, neutral forums** for member organizations [S13, S16, S39]. As neutral platforms, foundations **protect members' rights** [S41] and **intellectual property** (IP) of the consortium by having the ownership of IP [S13, S16, S28, S39]. Members do not have special rights on the ownership of the software [S40].

Members pay a specific amount of membership fee to join foundations. These fees are centrally collected and distributed to project expenses. In this way, foundations help user-led OS consortia to **ensure financial sustainability** by leveraging financial and staff resources [S4, S8, S15, S35].

(Umbrella) Foundations offer **governance support, technical support**, and **support in quality assurance**. As a part of governance support, they provide a clear path for participation and contribution [S13]. They can facilitate collaboration among involved organizations [S4, S16].

Furthermore, umbrella foundations can provide technology platforms [S16], and development (and test) infrastructures for collaborators to work on collectively [S13, S16, S37]. They can set the properties of the OSS code [S4] and offer easy integration with other OS projects [S11]. Foundations can ensure transparency in the requirement process, offer quality assurance in process [S35], and help to increase quality and quantity of contributions [S13].

Furthermore, foundations help **community development** and **community management** by increasing recognition. In order to reach prospective members, foundations organize conferences and meetings [S35, S39] and provide marketing support [S16]. Foundations can assist choosing the right members into a consortium in terms of culture fit [S35]. They help structuring the community [S4] and building a community of expertise with developers and user organizations [S8]. They keep the community together by coordinating work and projects, and member activities [S4, S8, S35, S37, S39].

4.2.3. RQ.2.3. Why do user organizations create user-led open source consortia?

Organizations that decide to create or join user-led OS consortia often share similar needs, such as an improved version of their existing systems [S11, S21, S34], reduced system complexity [S11], improved software quality [S11, S34], and reduced vendor dependency [S11, S39, S42].

From the literature we reviewed, we found that organizations primarily consider ***control, cost, sustainability,*** and ***productivity*** dimensions by their decisions on developing or buying software. Organizations perform this comparison by evaluating options of *upgrading their existing software, purchasing proprietary software, developing in-house software, using a community-led OSS*, or *joining in a user-led OS consortium*.

In this section, we present the reasons for choosing user-led OS consortia by highlighting the drawbacks of these alternatives and benefits of involvement in a user-led OS consortium. In Fig. 9, we present the benefits of user-led OS consortia involvement considering these four dimensions and the defining features of user-led OS consortia. In ([34]: Appendix F), we present a mapping of each benefit with data sources and related user-led OS consortia.

4.2.3.1. Control dimension. When *proprietary software* is not developed for a specific industry, it may lack the critical functionalities required within that industry [S6, S8, S21, S35]. The literature indicates that most proprietary software products lack specificity to particular sectors, are inflexible in functionality, and are difficult to customize [S6, S12, S18, S20, S21]. Furthermore, upgrades to these products may cause disruptive changes in other connected systems [S6, S8, S21, S38]. A further complaint of the user organizations about proprietary software is its limited capacity for innovation [S12, S26]. When user organizations request improvements for the proprietary software they use, they may experience a reduced ability to control timing of updates [S8], while software vendors often demonstrate a slow pace in implementing new features [S15, S26].

The other option, *in-house software development*, presents the challenge of keeping up with the pace of innovation in the field [S15, S39].

An alternative to proprietary software and in-house software development is the *use of community-led open-source software* (OSS). However, this software may lack the expected functionality [S6, S15, S35]. Dependence on volunteer developers and the risk of insufficient support are further problems [S12]. Other drawbacks of using community-led OSS for organizations include inconsistent governance models, multiple versions of libraries, siloed development, and varying licensing models [S6, S13].

In ***user-led OS consortia***, user organizations are organizations that work within the same industries, and collaborate with the aim of fulfilling **common needs** and software functionality requirements in their field [S2, S6, S12, S15, S25, S34]. Driver members, which mostly consist of user organizations, have the privilege to **define the functionality requirements** of software, prioritize tasks, and **lead the development direction** [S2, S14, S17, S19, S20, S21, S25, S28, S33, S43]. Since user organizations have control over the development process, they can prioritize their needs and **foster movement** for required functionalities [S3, S11, S21, S31, S37].

The software developed by user-led OS consortia is open source and offered to the community with various OSS licenses chosen by each

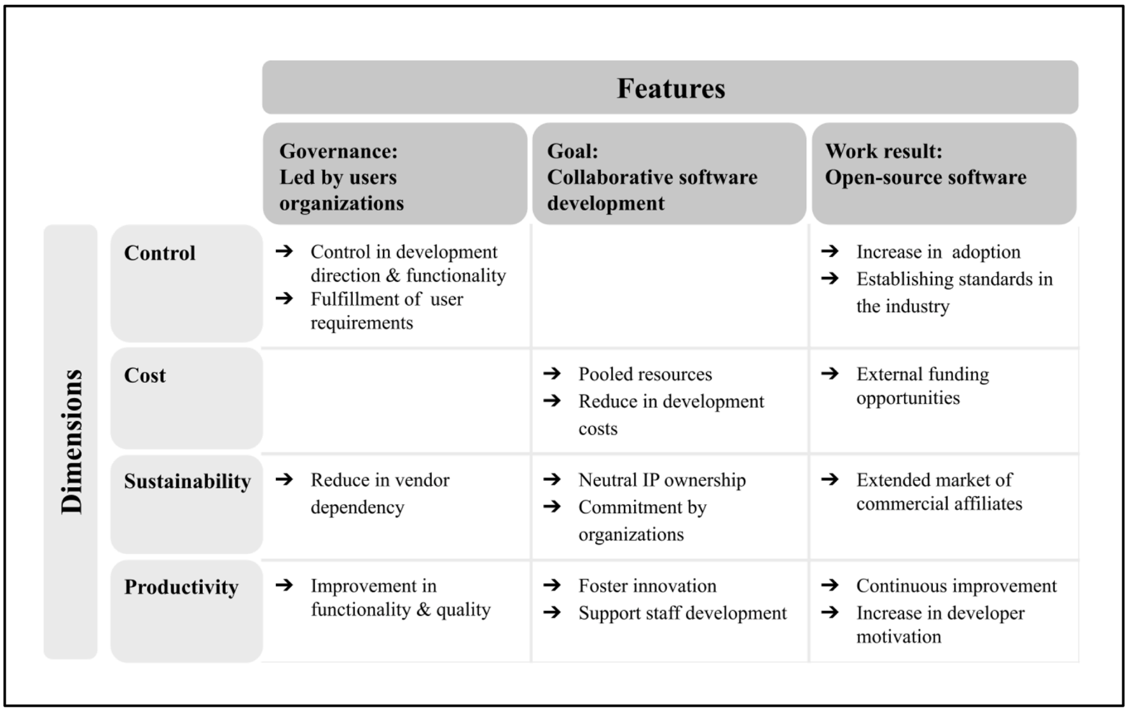

Fig. 9. Benefits of involvement in user-led open source consortia for user organizations.

consortium. This approach increases **the adoption of the software** among organizations with similar needs [S2, S31, S35]. An increase in the number of software users enhances the possibility of **establishing standards in related industry** [S11]

4.2.3.2. Cost dimension. *Proprietary software* products entail high entry costs [S6, S8, S12, S18, S21, S23] and licensing fees [S15, S26, S21]. On the other hand, developing *in-house software systems* without collaboration requires significant investment and maintenance costs for organizations [S21, S26, S28].

In ***user-led OS consortia***, organizations **pool their technical, personnel, and/or financial resources** [S6, S12, S14, S17, S20, S25, S39, S43]. By adopting this collaborative approach, organizations can significantly **reduce software development costs** compared to purchasing proprietary software or developing their in-house solutions [S3, S6, S8, S12, S14, S16, S18, S19, S20, S21, S22, S25, S34, S41]. In addition, organizations benefit from **reduced maintenance and operational costs** [S11, S21, S25, S34].

Since the developed end product is offered as OSS, some user-led OS consortia projects have the opportunity to receive **external funds** from funding organizations [S3, S12, S14, S18, S20, S21, S25, S26, S41, S42]. In examples from the higher education industry, some user-led OS consortia projects received funds from the Andrew W. Mellon Foundation during their early phases (e.g. Sakai, Kuali) [S18, S20, S41]. Later, to sustain development efforts, project participants established legal entities such as foundations, and began collecting fees from member organizations. For library and governmental projects, a similar funding approach can be observed, as seen in initiatives such as Hyku for Consortia and Oskari. For instance, the Hyku for Consortia project received partial funding from the Institute of Museum and Library Services [S25]. Similarly, the National Land Survey of Finland provided funds for the Oskari project during its early development phases [S14].

4.2.3.3. Sustainability dimension. Sustainability of *proprietary software* products depends on their developer companies (software vendors). One of the biggest concerns of user organizations regarding the sustainability of software products is the risk of vendor lock-in [S2, S6, S11, S21, S26, S34]. There is always a risk that vendors or systems may disappear [S8]. Furthermore, working with uncooperative vendors can result in a lack of support [S21, S26].

Alternatively, developing *in-house software systems* presents challenges in keeping the system functionalities up-to-date and ensuring long-term sustainability [S18]. Adding new functionalities to these systems can be cost-intensive [S11, S34].

User-led OS consortia projects **reduce dependence on vendors** compared to proprietary software systems, as these projects are led and financed by user organizations [S2, S6, S11, S34]. Offering the work results as OSS helps **extend the market to more vendors** [S11, S26, S34, S38] and **improve the quality of support services** through competition [S2].

In user-led OS consortia, the owner of the software is neither a vendor nor one of the driver organizations. In general, the **IP ownership** belongs to legal entities (e.g., foundations) representing user-led OS consortia (e.g., S28, S39, S40). These legal entities provide a neutral forum, ensuring the independence and reliability of projects. Another key factor in the sustainability of user-led OS consortia is the **commitment of member organizations** to the consortia and their projects. Organizations sign agreements with the consortia to become members, acknowledging their commitment to working on a project for a defined period. They also commit to regularly investing their resources—-whether in the form of human capital or financial contributions—thereby enhancing project sustainability [S6, S14, S21, S24, S43].

4.2.3.4. Productivity dimension. Since the member organizations in ***user-led OS consortia*** aim to achieve the same functionality in the end product, they follow a **stronger product vision** [S12]. They can focus on the continuous **improvement of software functionality and quality** [S6, S11, S16, S24, S38, S43].

Member organizations and their employees **share knowledge** and experience with each other regarding projects [S14, S25, S33]. This approach **fosters innovation** in projects [S2, S3, S6, S14, S16, S17, S19, S22, S31, S37, S38, S39]. Collaboration helps organizations **build expertise** [S8] and supports **staff development** [S8, S33, S39].

Since the work result is OSS, user-led OS consortia **benefit from the contributions of the community** [S1, S6, S37, S39, S35]. These contributions can take the form of innovative ideas [S2, S9, S14, S39], expert or technical insights [S37, S39], collaboration on future research and development [S1], as well as improvement suggestions and bug fixes [S1, S35].

Working on OSS development has a positive influence on **developer motivation** in user-led OS consortia projects [35, S34]. For instance, Samuel et al. [45] highlight that working collaboratively with other organizations on the Kuali Rice project motivates developers to help each other more, as developers seek to enhance both their organizations' and their own reputations through expertise in software development and contributions to the project [S33].

5. Discussion

In this section, we discuss the results of our study, its contributions, and future research areas.

5.1. The state of the art in user-led open source consortia literature

Our first objective was the identification of the state of the art in user-led OS consortia literature. Our SLR results show that 35 % of the published papers focus on the structure of specific user-led OS consortia or projects. The majority of these papers (12 out of 15) were experience papers. On the other hand, only 16 % of the papers have the focus on governance practices of user-led OS consortia. Existing research on governance primarily examines individual aspects, highlighting a lack of empirical studies on comprehensive governance approaches in user-led OS consortia.

Our analysis reveals that 70 % of studies on user-led OS consortia projects focus on higher education, with nearly half of the projects (6 out of 14) originating in this sector. This dominance may stem from early user-led OS consortia projects, such as Sakai and Kuali, being initiated in higher education. Additionally, many authors of these papers are project creators who discuss their experiences. Another contributing factor might be the project creators' familiarity with academic research and publishing.

Only 26 % of the studies focus on industries outside higher education, such as automotive, energy, finance, library, and entertainment. These studies emerged from 2013 onward. Expending the research effort across different industries will enable a more comprehensive understanding of the characteristics and benefits of user-led OS consortia. Extended research could offer practitioners valuable insights into the possibilities of open-source collaboration.

5.2. The structure of user-led open source consortia

Our second objective was identification of the structure of user-led OS consortia and projects, the ecosystem of user-led OS consortia, and the motivations of organizations for participation.

User-led OS consortia projects are led by user organizations, rather than individual volunteer developers or software vendors. For example, in higher education, the leading partners of user-led OS consortia projects (e.g., Sakai, Kuali) are universities, while in the energy industry (e.g., openKonsequenz) they are energy providers. These consortia emerge to address functionalities unique to related industries or business areas.

The structure of user-led OS consortia offers organizations the opportunity to define requirements based on their expectations and to influence the development direction of the required software.

In most user-led OS consortia projects, the underlying code is initially developed as closed source code, and subsequently made available to the public as open source after the initial release. This approach enables other organizations in the same industry to use the software and contribute to its improvements. For example, in the Sakai case, University of Lleida in Spain used the beta release of Sakai 1.0. version and translated the user interfaces into Catalan. They later contributed to the Sakai project by incorporating translation capabilities into the subsequent versions of the software. With the help of the community, the functionality of the Sakai project improved, and its user base grew [33].

It is expected that the resulting software does not provide a competitive advantage to any of the user members. Brooks [43] highlights that since a user-led OS consortium does not lead to a loss of income and each organization benefits from the collaboration, it attracts more organizations. Working on non-differentiated software encourages even computing companies to cooperate on OSS projects, allowing them to focus on unique features with the time saved [57,59,39]. These results can provide insight to practitioners considering involvement in open-source projects but who are concerned about protecting their competitive advantage.

Another contribution of this research is the presentation of the actors involved in user-led OS consortia. Umbrella foundations such as the Linux Foundation (LF) and the Eclipse Foundation (EF) define actors in user-led OS consortia environments using different terms. The status and voting rights of members are determined by the membership fees they pay. Organizations that aim to steer the development direction pay the highest membership fees and gain voting rights. The ASWF, hosted by the LF, categorizes its members as Premier, General, Associate Members.[9] Another user-led OS consortium, openMDM, hosted by the EF, defines five types of membership: driver members, user members, application vendor members, service provider members, and guests.[10] Through this research, we defined the actors based on organizations' roles in consortia and the development process to provide clarity. We identified five main actors: driver members (organizations), development partners, adopters (users), non-profit organizations, and (umbrella) foundations.

Foundations provide a neutral forum for members and legal protection, help to ensure financial sustainability by collecting membership fees, offer governance support, technical support, and quality assurance. Furthermore, foundations can also support community development and management by organizing conferences, supporting marketing activities, helping member selection, and member activities. Wheeler [24] highlights that working with a foundation helps the organizations to focus on software development, instead of directing their efforts to back-office support.

Some projects lead to the establishment of their own foundations, such as the Kuali Foundation, which originated from the Kuali Financial Systems Project, and the Apereo Foundation, which stemmed from the Sakai Project. Other projects chose to join already established umbrella foundations, such as the LF or the EF. For instance, the LF supports the openMAMA community by offering a governance framework, technology platform, marketing assistance, and IP protection [39]. In this research, we presented example projects that follow either of the two approaches. Each approach has its own benefits and drawbacks. Investigating these approaches could be a valuable topic for future research, providing useful insights for decision-makers involved in the establishment of such projects.

[9] https://cdn.platform.linuxfoundation.org/agreements/aswf.pdf

[10] https://www.eclipse.org/org/workinggroups/mdmwg_charter.php

5.3. Motivations of organizations for participation in user-led open source consortia

Organizations have diverse motives to participate in user-led OS consortia. User organizations are driven by the need for specific or complex functions that are lacking in proprietary software systems. For instance, when the first user-led OS consortia projects emerged, the commercial software available to educational institutions was often adapted from other industries and failed to provide the functionalities needed for educational processes [23]. Indiana University and the University of Hawaii initiated the Kuali Financial Systems project in 2004 with the aim of transitioning their financial information systems to a web-based open-source platform [52]. Liu et al. [44] elaborate that even with commercial software, universities still need to build 15 % of the necessary functions for financial transactions. Consequently, a project dedicated to meeting the fiscal data management and process needs of universities gained traction, leading to a growing number of member universities from 2005 onwards.

Use of proprietary software systems mostly leads to dependence on vendor companies. User organizations can address this challenge effectively by engaging in user-led OS consortia, where they define software functionalities, steer development direction, and cultivate a culture of movement and innovation. An example is openKonsequenz. In 2013 a number of Distribution System Operators (DSOs) of Germany initiated openKonsequenz consortia ([62]). DSOs required to update their software systems regarding external legal regulations. Being dependent on vendors was restricting their quality expectations, schedules and price negotiations. As a result, some of the DSOs collaborated to develop the undifferentiated parts of the software they required to break vendor lock-in ([62]).

For the proprietary software systems, there is a risk of disappearance of vendors or disappearance of systems. On the other hand, in user-led OS consortia, software projects are financed and led by user organizations, the continuity of projects depends on collaborative decisions of member organizations. The ownership of the IP does not belong to one organization, but to the general group (in most cases to the legal identity of the consortia). This approach increases the sustainability chance of software developed by a user-led OS consortium compared to proprietary software or community-led OSS.

A further characteristic of user-led OS consortia is that software is developed collaboratively. Collaborative software development enables reducing costs, and increasing productivity. Driver members finance projects by pooling resources and sharing development costs. This approach allows involved members to avoid the expenses associated with proprietary software licenses, or expenses involved in building or improving software themselves. In a collaborative development environment, participants share knowledge and ideas, learn from each other, and foster innovation through joint creation. Involved organizations benefit from this collaborative approach by building expertise and supporting staff development. Open Color IO (OCIO), an open-source color management library, is an example of the impact of this approach on shaping a project's future. Walker et al. [53] explain that the OCIO project was initially developed by Sony Pictures Imageworks, and open sourced in 2010. While the project successfully established de facto standards for color management in visual effects, community engagement declined for a few years. The revival of the project occurred in 2020 when it was adopted by the Academy Software Foundation. Since this library is valuable to the industry and its member organizations, they provide both financial and intellectual support to ensure the project's long-term sustainability.

In user-led OS consortia projects, the resulting software is open source. This enhances the likelihood of the adoption by other organizations with similar needs. As the user community expands, they can offer feedback and contribute to the code. Another advantage of open sourcing the software is the increased potential for a wider range of vendor options. Moreover, it positively impacts developer motivation.

We provided a list of studies, including their focus and project information, and detailed the user-led OS consortia projects studied in the literature. Furthermore, we synthesized the characteristics of user-led OS consortia, the actors involved in these types of consortia, and the reasons user organizations participate in them. We believe this research will be beneficial for researchers interested in investigating this phenomenon in detail, and it will offer guidance to practitioners interested in creating and developing user-led OS consortia. In our future research, we plan to continue exploring this topic and develop a best practices handbook for practitioners, providing insights into the problems and solutions involved in establishing user-led OS consortia.

6. Limitations

We performed an SLR by following Kitchenham [11] and Kitchenham & Brereton [12] methodology. We adopt Guba's[77] trustworthiness criteria, including credibility, confirmability, transferability, and dependability to discuss the limitations of our research.

Credibility is about the truth of the research findings. Our first concern of credibility was about selecting the relevant studies. To perform the selection process effectively, we used prolonged engagement practice. We conducted the study collection and selection process iteratively from August 2020 to March 2024 to extend the time frame of the papers published and include most up-to-date literature. We used Google Scholar and four other digital libraries to cover as many as possible studies and examined each of the resulting studies with great attention. The second concern of credibility was about the application of qualitative analysis and the results. We performed qualitative analysis by following Braun & Clarke [13]'s thematic analysis guideline. We used peer debriefing practice to ensure the credibility of the results. Although the first author performed the analysis, the method application procedures and results were discussed by periodic meetings with the second author. Furthermore, in a writer's workshop session, we shared the manuscript with our colleagues, and improved it with the feedback we received.

Confirmability is about objectivity. During the research period, the first author evaluated the relevance of each literature at least two times, and checked the data analysis results. While performing qualitative data analysis, we created a codebook and updated it regularly based on our findings. We present the codebook and sample codes in ([34]: Appendix C) for external auditors to examine the analysis we performed.

Transferability concerns establishing context-relevant statements. We addressed our research questions by analyzing related literature about user-led OS consortia. The majority of our findings show similarities in different projects from different industries. We present the mapping of our findings with related projects and consortia in ([34]: Appendix E and F). Although we believe that our findings can be generalizable for user-led OS consortia, it remains for future research to determine whether our findings can be applied to other user-led OS consortia projects.

Dependability refers to having reliable and traceable research findings. To address this concern, we provide the data of our research in the appendix. We present the data of included and excluded literature with details in ([34]: Appendix A), and the list of final literature in ([34]: Appendix D).

7. Conclusion

This research aimed to achieve two primary objectives: understanding the current state of literature on user-led open-source (OS) consortia and identifying their defining characteristics, key actors, and the motivations for user organizations to create and engage in such consortia.

To address our objectives, we conducted a systematic literature review (SLR) covering the years 2000 to 2023, identifying 43 unique papers directly related to user-led open-source (OS) consortia. Notably, the majority of these studies originate from the higher education sector, with papers addressing projects in other industries emerging only after 2013. This pattern highlights a significant gap in research on user-led OS consortia outside the higher education domain. We categorized the literature into five key concepts: the general structure, governance, ecosystem, creation of specific consortia, development processes of user-led OS projects. Our findings reveal gaps in literature addressing governance practices and ecosystem dynamics of these consortia, indicating areas for further exploration.

For the second objective, we conducted thematic data analysis to identify the defining characteristics of user-led OS consortia. We grouped these characteristics into three themes: led by user organizations, collaborative software development, and offering open-source software. Examining the roles within these ecosystems, we identified categories such as driver members, development partners, adopters (users), non-profit members, and foundations. Our analysis also shed light on the motivations for forming and participating in user-led OS consortia. Key benefits include greater control over development processes and functionality, enhanced sustainability, cost sharing, and improved productivity through collaboration.

This study provides a comprehensive overview of the current research on user-led OS consortia, identifies critical gaps in the literature, and highlights the defining characteristics and benefits of this model. These findings lay the groundwork for future research and practical exploration of user-led OS consortia across industries.

Declaration of generative AI in scientific writing

During the writing process of this work the authors used ChatGPT in order to improve and rephrase written paragraphs. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Funding

This research has been funded by the German Research Foundation (Deutsche Forschungsgemeinschaft). The project number of the fund is 506460878 and GZ:RI 2147/11-1 AOBJ: 690853..

CRediT authorship contribution statement

Elçin Yenişen Yavuz: Writing – review & editing, Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Dirk Riehle:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank our reviewers for their comments and suggestions for improvement. We would also like to express our gratitude to Ann Barcomb, Andreas Bauer, Stefan Buchner, Philip Heltweg, Andreas Kaufmann, Julia Mucha, Felix Quast, Georg Schwarz, and Özgür Yavuz for their valuable feedback on this research.

Data availability

We shared the data in the external appendix, with the link provided in the manuscript

References

[1] B. Fitzgerald, The transformation of open source software, MIS Q. (2006) 587–598.
[2] L. Dahlander, M.G. Magnusson, Relationships between open-source software companies and communities: observations from Nordic firms, Res. Policy 34 (4) (2005) 481–493.
[3] L. Dahlander, Penguin in a new suit: a tale of how de novo entrants emerged to harness free and open-source software communities, Ind. Corp. Change 16 (5) (2007) 913–943.
[4] N. Harutyunyan, D. Riehle, G. Sathya, Industry best practices for corporate open sourcing, in: Proceedings of the 53rd Hawaii International Conference on System Sciences, 2020.
[5] J. West, S. O'Mahony, Contrasting community building in sponsored and community founded open source projects, in: Proceedings of the 38th Annual Hawaii International Conference on System Sciences, IEEE, 2005, p. 196c. -196c.
[6] D. Riehle, S. Berschneider, A model of open source developer foundations, in: IFIP International Conference on Open Source Systems, 2012.
[7] M. Schaarschmidt, M. Bertram, H.F. von Kortzfleisch, Exposing differences of governance approaches in single and multi vendor open-source software development, in: IFIP International Working Conference on Governance and Sustainability in Information Systems-Managing the Transfer and Diffusion of IT, 2011.
[8] E. Yenişen Yavuz, D. Riehle, A. Mehrotra, Why do companies create and how do they succeed with a vendor-led open source foundation, Empir. Softw. Eng. 30 (1) (2025) 1–49.
[9] E. Yenişen Yavuz, A. Barcomb, D. Riehle, Problems, solutions, and success factors in the openMDM user-led open source consortium, Commun. Assoc. Inf. Syst. 51 (1) (2022) 13.
[10] J. Teixeira, S. Mian, U. Hytti, Cooperation among competitors in the open-source arena: the case of OpenStack. arXiv preprint arXiv:1612.09462. (2016).
[11] B. Kitchenham, Procedures for Performing Systematic Reviews, 33, Keele University, Keele, UK, 2004, pp. 1–26.
[12] B. Kitchenham, P. Brereton, A systematic review of systematic review process research in software engineering, Inf. Softw. Technol. 55 (12) (2013) 2049–2075.
[13] V. Braun, V. Clarke, Thematic Analysis, American Psychological Association, 2012.
[14] R. Eckert, M. Stuermer, T. Myrach, Alone or together? Inter-organizational affiliations of open source communities, J. Syst. Softw. 149 (2019) 250–262.
[15] J.L.C. Izquierdo, J. Cabot, A survey of software foundations in open source. arXiv preprint arXiv:2005.10063. (2020).
[16] E. von Hippel, G. von Krogh, Open-source software and the "Private-Collective" innovation model: issues for organization science, Organ. Sci. 14 (2) (2003) 209–223.
[17] R. Stallman, Free software foundation (fsf). Encyclopedia of Computer Science, 2003, pp. 732–733.
[18] P. Hunter, S. Walli, The rise and evolution of the open source software foundation, IFOSS L. Rev. 5 (2013) 31.
[19] J. Perlow, The Linux foundation: it's not just the Linux operating system. (2020) Retrieved December 1, 2024 from: https://www.linuxfoundation.org/blog/blog/the-linux-foundation-its-not-just-the-linux-operating-system.
[20] F. Weikert, D. Riehle, A. Barcomb, Managing commercial conflicts of interest in open source foundations, in: International Conference on Software Business, Springer International Publishing, Cham, 2019, pp. 130–144.
[21] M. Shaikh, T. Cornford, Letting go of control to embrace open source: implications for company and community, in: 2010 43rd Hawaii International Conference on System Sciences, IEEE, 2010, pp. 1–10.
[22] Y. Zhang, M. Zhou, K.J. Stol, J. Wu, Z. Jin, How do companies collaborate in open source ecosystems? An empirical study of openstack, in: Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering, 2020, pp. 1196–1208.
[23] P.N. Courant, R.J. Griffiths, Software and collaboration in higher education: a study of open-source software. Organization for Open-source software Study. (2006).
[24] B. Wheeler, Open source 2010: reflections on 2007, Educause Rev. 42 (1) (2007) 49–52.
[25] M. Liu, X. Wu, J.L. Zhao, L. Zhu, Outsourcing of community source: identifying motivations and benefits, J. Glob. Inf. Manag. (JGIM) 18 (4) (2010) 36–52.
[26] H.W. Chesbrough, M.M. Appleyard, Open innovation and strategy, Calif. Manag. Rev. 50 (1) (2007) 57–76.
[27] J. Perr, M.M. Appleyard, P. Patrick, Open for business: emerging business models in open-source software, Int. J. Technol. Manag. 52 (3/4) (2010) 432–456.
[28] D. Riehle, The innovations of open source, Computer (Long Beach Calif) 52 (4) (2019) 59–63.
[29] T.S. Almigheerbi, D. Ramsey, A. Lamek, Information Management in a Collaboratively-Developed Approach to Enterprise Resource Planning—A Higher Education Perspective, Information 11 (3) (2020) 146.
[30] B. Schwab, D. Riehle, A. Barcomb, N. Harutyunyan, The ecosystem of openKONSEQUENZ, a user-led open source foundation, in: Open Source Systems: 16th IFIP WG 2.13 International Conference, OSS 2020, Innopolis, Russia, May 12–14, 2020, Proceedings 16, Springer International Publishing, 2020, pp. 1–13.
[31] B. Kitchenham, O.P. Brereton, D. Budgen, M. Turner, J. Bailey, S. Linkman, Systematic literature reviews in software engineering–a systematic literature review, Inf. Softw. Technol. 51 (1) (2009) 7–15.
[32] O. Díaz, M. Genero, J.P. Contell, M. Piattini, Adding relevance to rigor: assessing the contributions of SLRs in software engineering through citation context analysis, Comput. Sci. Rev. 53 (2024) 100649.
[33] C. Severance, Sakai: building an open source community. CreateSpace Independent Publishing Platform (2011).
[34] E. Yenişen Yavuz, D. Riehle, External appendixes: why and how do organizations create user-led open source consortia? A Systematic Literature Review. https://faubox.rrze.uni-erlangen.de/getlink/fiWwX5LtjBRnm6qU67BuUW/.
[35] T. Dybå, T. Dingsoyr, G.K. Hanssen, Applying systematic reviews to diverse study types: an experience report, in: First international symposium on empirical software engineering and measurement (ESEM 2007), IEEE, 2007, pp. 225–234.
[36] K.R. Felizardo, E. Mendes, M. Kalinowski, É.F. Souza, N.L. Vijaykumar, Using forward snowballing to update systematic reviews in software engineering, in: Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, 2016, pp. 1–6.
[37] C. Wohlin, Guidelines for snowballing in systematic literature studies and a replication in software engineering, in: Proceedings of the 18th international conference on evaluation and assessment in software engineering, 2014, pp. 1–10.
[38] R. Baecker, Open source strategies for educational multimedia. EdMedia+ Innovate Learning, Association for the Advancement of Computing in Education (AACE), 2005, pp. 133–137.
[39] M. Levy, M. Germonprez, Is it egalitarianism or enterprise strategy?, in: Exploring a new method of innovation in open source. In the 21st Americas Conference on Information Systems AMCIS, 2015.
[40] G. Robles, J. Gamalielsson, B. Lundell, Setting up government 3.0 solutions based on open-source software: the case of X-road, in: Electronic Government: 18th IFIP WG 8.5 International Conference, EGOV 2019, San Benedetto Del Tronto, Italy, September 2–4, 2019, Proceedings 18, Springer International Publishing, 2019, pp. 69–81.
[41] J. Baron, J. Willis, R.A. Lee, Creating higher education academic and information technology resources in an international context, Comput. Sch. 27 (3–4) (2010) 288–308.
[42] M. Sabin, J. Leone, IT education 2.0, in: Proceedings of the 10th ACM conference on SIG-information technology education, 2009, pp. 91–99.
[43] L. Brooks, Values of community source development. Syllabus, September, 2004.
[44] M. Liu, H.J. Wang, J.L. Zhao, Technology flexibility as enabler of robust application development in community source: the case of Kuali and Sakai, J. Syst. Softw. 85 (12) (2012) 2921–2928.
[45] B.M. Samuel, H. Bala, S.L. Daniel, V. Ramesh, Deconstructing the nature of collaboration in organizations open-source software development: the impact of developer and task characteristics, IEEE Trans. Softw. Eng. 48 (10) (2022) 3969–3987.
[46] W. Al Bulushi, Governing university-specific open source ERP: how open is open enough? (Vol. 10, (2019) 18.
[47] M. Liu, S. Hansen, Q. Tu, The community source approach to software development and the Kuali experience, Commun. ACM 57 (5) (2014) 88–96.
[48] M. Liu, D.D. Zeng, J.L. Zhao, Cooperative investment decisions in community source development, Int. J. Inf. Technol. Decis. Mak. 13 (01) (2014) 5–28.
[49] C. Severance, Pushing the boundaries of innovation through community source. Merit Member Conference. 2007.
[50] M. Liu, C.E. Hull, Y.T.C. Hung, Starting open source collaborative innovation: the antecedents of network formation in community source, Inf. Syst. J. 27 (5) (2017) 643–670.
[51] J. Farmer, I. Dolphin, Sakai: eLearning and more, in: Proceedings of the 11th European University Information Systems, Manchester, UK, 2005.
[52] M. Liu, S. Hansen, Q. Tu, Keeping the family together: sustainability and modularity in community source development, Inf. Organ. 30 (1) (2020) 100274.
[53] D. Walker, M. Dolan, P. Hodoul, The ASWF takes OpenColorIO to the next level, in: The Digital Production Symposium, 2020, pp. 1–10.
[54] J. Foutty, Kuali Foundation Delivers Open-Source Software, University Information Systems, 2010, p. 33.
[55] M. Liu, S. Hansen, Q. Tu, Sustaining collaborative software development through strategic consortium, J. Strateg. Inf. Syst. 30 (3) (2021) 101671.
[56] B. Wheeler, J. DeStefano. Mitigating the risks of big systems, 2007.
[57] M. Germonprez, J.P. Allen, B. Warner, J. Hill, G. McClements, Open source communities of competitors, Interactions 20 (6) (2013) 54–59.
[58] C.J. Mackie, Open source in open education: promises and challenges, Opening up Education: The Collective Advancement of Education Through Open Technology, Open Content and Open Knowledge (2008) 119–131.
[59] M. Germonprez, M. Levy, J.E. Kendall, K.E. Kendall, Tapestries of innovation: structures of contemporary open source project engagements, J. Assoc. Inf. Syst. 21 (3) (2020) 5.
[60] J. Morris, K. Leonard, Collaborating across consortial boundaries, Collaborative Librarianship 11 (4) (2020).
[61] B. Wheeler, Sakai: a case study in sustainability. OSSWatch. 2007.
[62] A. Goering, J. Meister, S. Lehnhoff, P. Herdt, M. Jung, M. Rohr, Reference architecture for open, maintainable and secure software for the operation of energy networks, CIRED-Open Access Proc. J. 2017 (1) (2017) 1410–1413.
[63] D.R. Nidy, F. Kwok, Community source development: an emerging model with new opportunities. CHI'05 Extended Abstracts on Human Factors in Computing Systems, 2005, pp. 1697–1700.
[64] B. Wheeler, J.L. Hilton, The Marketecture of Community, EDUCAUSE Review 47 (6) (2012) 66–68.
[65] M. Hancock, Partner for more: creating and sustaining collaboration to support campus-based rich media, Contin. High. Educ. Rev. 74 (2010) 73–80.
[66] M. Winkler, Governance case studies: open library environment (OLE), 2018, in: L. G. Arp, M. Forbes (Eds.), It Takes a Village: Open-Source Software Sustainability, LYRASIS, 2018, pp. 17–18.

[67] D. Heckenberg, J.F. Panisset, E. Olin, ASWF technical advisory committee: how to enable an open source community, in: Proceedings of the 2019 Digital Production Symposium, 2019, pp. 1–2.
[68] K.G. Provan, R.H. Lemaire, Positional embeddedness in a community source software development project network: the importance of relationship intensity, R&D Manag. 45 (5) (2015) 440–457.
[69] K. Henttonen, J. Kääriäinen, J. Kylmäaho, Lifecycle management in government-driven open source projects–practical framework, Int. J. Inf. Syst. Proj. Manag. 5 (3) (2017) 23–41.
[70] K. Rankin, R. Baecker, Evolution of an open source strategy, Open Source Business Resource (2007).
[71] M. Ketterl, O.A. Schulte, A. Hochman, Opencast Matterhorn: a community-driven open-source software project for producing, managing, and distributing academic video, Interact. Technol. Smart Educ. 7 (3) (2010) 168–180.
[72] Kuali Foundation. (n.d.). Retrieved June 5, 2024, from https://www.kuali.org/.
[73] Apereo. (n.d.). *About the Apereo Foundation* Retrieved June 5, 2024, from https://www.apereo.org/about.
[74] Apereo Community Blog (n.d.). *About Apereo.* Retrieved June 5, 2024, from https://apereo.github.io/about/.
[75] Oskari. (n.d.) *Community behind the software* Retrieved June 5, 2024, from https://oskari.org/community/oskari_community.
[76] J. Webster, R.T. Watson, Analyzing the past to prepare for the future: Writing a literature review, MIS quarterly (2002) xiii–xxiii.
[77] E.G. Guba, Criteria for assessing the trustworthiness of naturalistic inquiries, Ectj 29 (2) (1981) 75–91.

B

Paper 2

This chapter is based on the article titled "Problems, solutions, and success factors in the open-MDM user-led open source consortium" originally published in *Communications of the Association for Information Systems*:

Yenişen Yavuz, E., Barcomb, A., & Riehle, D. (2022). Problems, solutions, and success factors in the openMDM user-led open source consortium. *Communications of the Association for Information Systems*, 51(1), 13. https://doi.org/10.17705/1CAIS.05122

Communications of the Association for Information Systems

Volume 51 Paper

10-25-2022

Problems, Solutions, and Success Factors in the openMDM User-Led Open Source Consortium

Elçin Yenişen Yavuz
Friedrich-Alexander-University Erlangen-Nürnberg, elcin.yenisen@fau.de

Ann Barcomb
University of Calgary

Dirk Riehle
Friedrich-Alexander-University Erlangen-Nürnberg

Recommended Citation

Yavuz, E. Y., Barcomb, A., & Riehle, D. (2022). Problems, Solutions, and Success Factors in the openMDM User-Led Open Source Consortium. Communications of the Association for Information Systems, 51, pp-pp. https://doi.org/10.17705/1CAIS.05122

Communications of the Association for Information Systems

Research Article DOI: 10.17705/1CAIS.05122 ISSN: 1529-3181

Problems, Solutions, and Success Factors in the openMDM User-Led Open Source Consortium

Elçin Yenişen Yavuz
Computer Science Department,
Friedrich-Alexander-Universität Erlangen-Nürnberg
elcin.yenisen@fau.de

Ann Barcomb
Schulich School of Engineering,
University of Calgary

Dirk Riehle
Computer Science Department,
Friedrich-Alexander-Universität Erlangen-Nürnberg

Abstract:

Open-source software (OSS) development offers organizations an alternative to purchasing proprietary software or commissioning custom software. In one form of OSS development, organizations develop the software they need in collaboration with other organizations. If the software is used by the organizations to operate their business, such collaborations can lead to what we call “user-led open-source consortia” or “user-led OSS consortia”. Although this concept is not new, there have been few studies of user-led OSS consortia. The studies that examined user-led OSS consortia did so through the lens of OSS, but not from the inter-company collaboration perspective. User-led OSS consortia are a distinct phenomenon that share elements of inter-company collaboration, outsourcing software development, and vendor-led OSS development and cannot be understood by using only a single lens. To close this gap, we present problems and solutions in inter-company collaboration, outsourcing, and OSS literature, and present the results of a single-case study. We focus on problems in the early phases of a user-led open-source consortium, the openMDM consortium, and the solutions applied to these problems. Furthermore, we present the factors which lead this consortium to sustained growth.

Keywords: Open Source Software, Collaborative Software Development, Open Source User-Led Consortia, Open Source Foundations, Community Source, Eclipse Foundation, Success Factors, Outsourcing.

This manuscript underwent peer review. It was received 8/11/2021 and was with the authors for eight months for one revision. Julie Kendall served as Associate Editor.

References

Ågerfalk, P. J., & Fitzgerald, B. (2008). Outsourcing to an unknown workforce: Exploring opensourcing as a global sourcing strategy. *MIS Quarterly*, *32(2),* 385-409.

Almeida, F., Oliveira, J., & Cruz, J. (2011). Open standards and open source: Enabling interoperability. *International Journal of Software Engineering & Applications*, *2*(1), 1-11.

Aron, R., Clemons, E. K., & Reddi, S. (2005). Just right outsourcing: Understanding and managing risk. *Journal of Management Information Systems*, *22*(2), 37-55.

Bacon, J. (2012). *The art of community: Building the new age of participation*. O'Reilly Media, Inc.

Baldwin, C., & Von Hippel, E. (2011). Modeling a paradigm shift: From producer innovation to user and open collaborative innovation. *Organization Science, 22(6)*, 1399-1417.

Barcomb, A., Kaufmann, A., Riehle, D., Stol, K. J., & Fitzgerald, B. (2018). Uncovering the periphery: A qualitative survey of episodic volunteering in free/libre and open source software communities. *IEEE Transactions on Software Engineering*, *46*(9), 962-980.

Barcomb, A., Stol, K. J., Fitzgerald, B., & Riehle, D. (2020). Managing episodic volunteers in free/libre/open source software communities. *IEEE Transactions on Software Engineering, 48*(1), 260-277.

Barcomb, A., Stol, K. J., Riehle, D., & Fitzgerald, B. (2019). Why do episodic volunteers stay in FLOSS communities? In *2019 IEEE/ACM 41st ICSE*.

Barthélemy, J., & Quélin, B. V. (2006). Complexity of outsourcing contracts and ex post transaction costs: An empirical investigation. *Journal of Management Studies*, *43*(8), 1775-1797.

Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly*, *11*(3), 369-386.

Berdou, E. (2006). Insiders and outsiders: Paid contributors and the dynamics of cooperation in community led F/OS projects. In *IFIP International Conference on Open Source Systems* (pp. 201-208). Springer.

Boldyreff, C., Nutter, D., & Rank, S. (2004). Communication and conflict issues in collaborative software research projects. In *26th International Conference on Software Engineering - W8S Workshop "Collaboration, Conflict and Control: The 4th Workshop on Open Source Software Engineering".*

Bosu, A., & Sultana, K. Z. (2019). Diversity and inclusion in open source software (OSS) projects: Where do we stand? In *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*.

Bruce, M., Leverick, F., Littler, D., & Wilson, D. (1995). Success factors for collaborative product development: A study of suppliers of information and communication technology. *R&D Management, 25(1)*, pp. 33-44.

Carillo, K. D. A., & Marsan, J. (2016). “The dose makes the poison”- Exploring the toxicity phenomenon in online communities. In *International Conference on Information Systems.*

Castilla, E. J., & Benard, S. (2010). The paradox of meritocracy in organizations. *Administrative Science Quarterly, 55(4)*, 543-676.

Chengalur-Smith, I., Sidorova, A., & Daniel, S. L. (2010). Sustainability of free/libre open source projects: A longitudinal study. *Journal of the Association for Information Systems*, *11*(11), 5.

Chin, K. S., Chan, B. L., & Lam, P. K. (2008). Identifying and prioritizing critical success factors for coopetition strategy. *Industrial Management & Data Systems, 104*(4), 437-454.

Choudhury, V., & Sabherwal, R. (2003). Portfolios of control in outsourced software development projects. *Information Systems Research*, *14*(3), 291-314.

Ciesielska, M., & Westenholz, A. (2016). Dilemmas within commercial involvement in open source software. *Journal of Organizational Change Management*, *29*(3), 344-360.

Colazo, J. A., & Fang, Y. (2010). Following the sun: Temporal dispersion and performance in open source software project teams. *Journal of the Association for Information Systems*, *11*(11), 4.

Conley, C. A., & Sproull, L. (2009). Easier said than done: An empirical investigation of software design and quality in open source software development. In *2009 42nd Hawaii International Conference on System Sciences*. IEEE.

Courant, P. N., & Griffiths, R. J. (2006). Software and collaboration in higher education: A study of open source software. Retrieved from *https://www.campussource.de/opensource/docs/OOSS_Report.pdf.*

Crowston, K., & Howison, J. (2005). The social structure of free and open source software development. *First Monday*, *10*(2).

Crowston, K., Annabi, H., & Howison, J. (2003). Defining open source software project success. In *Proceedings of the International Conference on Information Systems*.

Dacin, M. T., Hitt, M. A., & Levitas, E. (1997). Selecting partners for successful international alliances: Examination of US and Korean firms. *Journal of World Business, 32(1)*, 3-16.

Dahlander, L. (2007). Penguin in a new suit: A tale of how de novo entrants emerged to harness free and open source software communities. *Industrial and Corporate Change*, *16*(5), 913-943.

Dahlander, L., & Magnusson, M. G. (2005). Relationships between open source software companies and communities: Observations from Nordic firms. *Research Policy*, *34*(4), 481-493.

Damian, D. (2003). Global software development: Growing opportunities, ongoing challenges. *Software Process: Improvement and Practice, 8(4),* 179-182.

Daniel, S., Agarwal, R., & Stewart, K. J. (2013). The effects of diversity in global, distributed collectives: A study of open source project success. *Information Systems Research, 24(2)*, 312-333.

Ducheneaut, N. (2005). Socialization in an open source software community: A socio-technical analysis. *Computer Supported Cooperative Work*, *14*(4), 323-368.

Ehls, D. (2017). Open source project collapse–sources and patterns of failure. In *2017 Proceedings of the 50th Hawaii International Conference on System Sciences.*

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review, 14(4)*, 532-550.

Feller, J., & Fitzgerald, B. (2000). A framework analysis of the open source software development paradigm. In *ICIS 2000 Proceedings of the 21st International Conference on Information Systems*.

Filippova, A., & Cho, H. (2015). Mudslinging and manners: Unpacking conflict in free and open source software. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing* (pp. 1393-1403).

Fogel, K. (2005). *Producing open source software: How to run a successful free software project.* O'Reilly Media, Inc.

Fortuin, F. T., & Omta, S. W. F. (2008). The dark side of open innovation: A survey of failed inter-company cooperation. *In Proceedings of the 8th International Conference on Management in AgriFood Chains and Networks.*

Foucault, M., Palyart, M., Blanc, X., Murphy, G. C., & Falleri, J. R. (2015, August). Impact of developer turnover on quality in open-source software. In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering* (pp. 829-841).

Geiger, R. S., Howard, D., & Irani, L. (2021). The labor of maintaining and scaling free and open-source software projects. In *Proceedings of the ACM on Human-Computer Interaction*, *5*(CSCW1) (pp. 1-28).

Germonprez, M., Allen, J. P., Warner, B., Hill, J., & McClements, G. (2013). Open source communities of competitors. *Interactions, 20(6)*, 54-59.

Guba, E. G. (1981). Criteria for assessing the trustworthiness of naturalistic inquiries. *Educational Communication and Technology, 29*(2), pp. 75-91.

Guizani, M., Chatterjee, A., Trinkenreich, B., May, M. E., Noa-Guevara, G. J., Russell, L. J., Cuevas Zambrano, G.G., Izquierdo-Cortazar, D., Steinmacher, I., Gerosa, M.A. & Sarma, A. (2021). The long road ahead: Ongoing challenges in contributing to large OSS organizations and what to do. In *Proceedings of the ACM on Human-Computer Interaction*, *5*(CSCW2), (pp. 1-30).

Haefliger, S., Von Krogh, G., & Spaeth, S. (2008). Code reuse in open source software. *Management Science*, *54*(1), 180-193.

Harutyunyan, N., Riehle, D., & Sathya, G. (2020, January). Industry best practices for corporate open sourcing. In *Proceedings of the 53rd Hawaii International Conference on System Sciences*.

Henkel, J. (2004). Open source software from commercial firms–tools, complements, and collective invention. *Zeitschrift für Betriebswirtschaft*, *4*, 1-23.

Herbsleb, J. D., & Grinter, R. E. (1999). Splitting the organization and integrating the code: Conway's Law revisited. In *Proceedings of the 21st International Conference on Software Engineering*.

Hoffmann, W. H., & Schlosser, R. (2001). Success factors of strategic alliances in small and medium-sized enterprises—An empirical survey. *Long Range Planning*, *34(3),* 357-381.

Howison, J., & Crowston, K. (2014). Collaboration through open superposition: A theory of the open source way. *MIS Quarterly*, *38*(1), 29-50.

Kelly, M. J., Schaan, J. L., & Joncas, H. (2002). Managing alliance relationships: Key challenges in the early stages of collaboration. *R&D Management, 32(1)*, 11-22.

Kochhar, P. S., Kalliamvakou, E., Nagappan, N., Zimmermann, T., & Bird, C. (2019). Moving from closed to open source: Observations from six transitioned projects to GitHub. *IEEE Transactions on Software Engineering*.

Lee, G. K., & Cole, R. E. (2003). From a firm-based to a community-based model of knowledge creation: The case of the Linux kernel development. *Organization Science*, *14*(6), 633-649.

Levy, M., & Germonprez, R. M. (2015). Is it egalitarianism or enterprise strategy? Exploring a new method of innovation in open source. In *21st AMCIS 2015*.

Lin, B., Robles, G., & Serebrenik, A. (2017). Developer turnover in global, industrial open source projects: Insights from applying survival analysis. In *2017 IEEE 12th International Conference on Global Software Engineering*.

Liu, M., Hansen, S., & Tu, Q. (2014). The community source approach to software development and the Kuali experience. *Communications of the ACM, 57(5)*, 88-96.

Liu, M., Hansen, S., & Tu, Q. (2020). Keeping the family together: Sustainability and modularity in community source development. *Information and Organization*, *30*(1), 100274.

Liu, M., Wang, H. J., & Zhao, J. L. (2012). Technology flexibility as enabler of robust application development in community source: The case of Kuali and Sakai. *Journal of Systems and Software, 85(12)*, 2921-2928

Liu, M., Wang, H., & Zhao, L. (2007). Achieving flexibility via service-centric community source: The case of Kuali. In *AMCIS 2007 Proceedings*.

Liu, M., Wu, X., Zhao, J. L., & Zhu, L. (2010). Outsourcing of community source: identifying motivations and benefits. *Journal of Global Information Management, 18(4)*, 36-52.

Liu, M., Zeng, D. D., & Zhao, J. L. (2008). A cooperative analysis framework for investment decisions in community source partnerships. In *AMCIS 2008 Proceedings*.

Lumbard, K., Wethor, G., Goggins, S., Buhman, A., Hale, M., & Germonprez, M. (2020). Welcome? Investigating the reception of new contributors to organizational-communal open source software projects. In *26th Americas Conference on Information Systems, AMCIS 2020*. Association for Information Systems.

Maruping, L. M., Daniel, S. L., & Cataldo, M. (2019). Developer centrality and the impact of value congruence and incongruence on commitment and code contribution activity in open source software communities. *MIS Quarterly*, *43*(3), 951-976.

Mattessich, P. W., & Monsey, B. R. (1992). *Collaboration: What makes it work. A review of research literature on factors influencing successful collaboration.* Amherst H. Wilder Foundation.

Mauerer, W., & Jaeger, M. C. (2013). Open source engineering processes/open source-entwicklungsprozesse. *IT-Information Technology*, *55*(5), 196-203.

Medappa, P. K., & Srivastava, S. C. (2019). Does superposition influence the success of FLOSS projects? An examination of open-source software development by organizations and individuals. *Information Systems Research*, *30*(3), 764-786.

Michlmayr, M. (2004). Managing volunteer activity in free software projects. In *USENIX Annual Technical Conference, FREENIX Track*.

Michlmayr, M., Fitzgerald, B., & Stol, K. J. (2015). Why and how should open source projects adopt time-based releases?, *IEEE Software*, *32*(2), 55-63.

Midha, V., & Palvia, P. (2007). Retention and quality in open source software projects. In *AMCIS 2007 Proceedings*.

Midha, V., & Palvia, P. (2012). Factors affecting the success of open source software. *Journal of Systems and Software, 85(4),* 895-905.

Nafus, D. (2012). 'Patches don't have gender': What is not open in open source software. *New Media & Society, 14(4)*, 669-683.

Narduzzo, A., & Rossi, A. (2005). The role of modularity in free/open source software development. In *Free/Open Source Software Development* (pp. 84-102). Igi Global.

O'Leary, K., Gleasure, R., O'Reilly, P., & Feller, J. (2020). Reviewing the contributing factors and benefits of distributed collaboration. *Communications of the Association for Information Systems*, *47*, 476-520.

Ostrom, E. (1990). *Governing the commons: The evolution of institutions for collective action*. Cambridge University Press.

Patton, M. Q. (1990). *Qualitative evaluation and research methods*. SAGE.

Piccoli, G., & Ives, B. (2003). Trust and the unintended effects of behavior control in virtual teams. *MIS Quarterly*, *27*(3),365-395.

Pinto, G., Steinmacher, I., Dias, L. F., & Gerosa, M. (2018). On the challenges of open-sourcing proprietary software projects. *Empirical Software Engineering*, *23*(6), 3221-3247.

Qureshi, I., & Fang, Y. (2011). Socialization in open source software projects: A growth mixture modeling approach. *Organizational Research Methods*, *14*(1), 208-238.

Radtke, N. P., Janssen, M. A., & Collofello, J. S. (2009). What makes Free/Libre Open Source Software (FLOSS) projects successful? An agent-based model of FLOSS projects. *International Journal of Open Source Software and Processes*, *1*(2), 1-13.

Rai, A., Borah, S., & Ramaprasad, A. (1996). Critical success factors for strategic alliances in the information technology industry: An empirical study. *Decision Sciences, 27(1)*, 141-155.

Rashid, M., Clarke, P. M., & O'Connor, R. V. (2017). Exploring knowledge loss in open source software (OSS) projects. In *International Conference on Software Process Improvement and Capability Determination*.

Riehle, D. (2010). The economic case for open source foundations. *Computer, (1)*, 86-90.

Riehle, D. (2015). The five stages of open source volunteering. In W. Li, M. N. Huhns, W-T., Tsai, & W. Wu (Eds.), *Crowdsourcing* (pp. 25-38). Springer.

Riehle, D., & Berschneider, S. (2012). A model of open source developer foundations. In *IFIP International Conference on Open Source Systems*.

Riembauer, S., Hornung, O., & Smolnik, S. (2020). Knowledge unchained or strategically overseen? Knowledge management in open source software projects. In *Proceedings of the 53rd Hawaii International Conference on System Sciences*.

Rigby, P., German, D., & Storey, M. A. (2008). Open source software peer review practices: A case study of apache server. In *2008 ACM/IEEE 30th International Conference on Software Engineering*.

Rikkiev, A., & Mäkinen, S. (2009). Success factors for technology integration convergence collaborations: Empirical assessment. In *2009 Portland International Conference on Management of Engineering & Technology*.

Sagers, G. (2004). The influence of network governance factors on success in open source software development projects. In *International Conference on Information Systems 2004 Proceedings*.

Schaarschmidt, M., Bertram, M., & von Kortzfleisch, H. F. (2011). Exposing differences of governance approaches in single and multi vendor open source software development. In *IFIP International Working Conference on Governance and Sustainability in Information Systems-Managing the Transfer and Diffusion of IT*.

Schwab, B., Riehle, D., Barcomb, A., & Harutyunyan, N. (2020). The ecosystem of openKonsequenz, a user-led open source foundation. In *IFIP International Conference on Open Source Systems*.

Senyard, A., & Michlmayr, M. (2004, November). How to have a successful free software project. In *11th Asia-Pacific Software Engineering Conference* (pp. 84-91). IEEE.

Singh, P. V. (2010). The small-world effect: The influence of macro-level properties of developer collaboration networks on open-source project success. *ACM Transactions on Software Engineering and Methodology*, 20(2), 1-27.

Singh, V., & Brandon, W. (2019). Open source software community inclusion initiatives to support women participation. In *IFIP International Conference on Open Source Systems*.

Steinmacher, I., Conte, T. U., & Gerosa, M. A. (2015b). Understanding and supporting the choice of an appropriate task to start with in open source software communities. In *2015 48th Hawaii International Conference on System Sciences* (pp. 5299-5308).

Steinmacher, I., Conte, T., Gerosa, M. A., & Redmiles, D. (2015a). Social barriers faced by newcomers placing their first contribution in open source software projects. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*.

Stewart, K. J., Ammeter, A. P., & Maruping, L. M. (2005). A preliminary analysis of the influences of licensing and organizational sponsorship on success in open source projects. In *Proceedings of the 38th Annual HICSS.*

Strauss, A., & Corbin, J. (1990). *Basics of qualitative research: Grounded theory procedures and techniques.* SAGE.

Tiwana, A., & Bush, A. A. (2007). A comparison of transaction cost, agency, and knowledge-based predictors of IT outsourcing decisions: A US-Japan cross-cultural field study. *Journal of Management Information Systems*, *24*(1), 259-300.

Tsay, J., Dabbish, L., & Herbsleb, J. (2014). Let's talk about it: Evaluating contributions through discussion in GitHub. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*.

Van Wendel de Joode, R. (2004). Managing conflicts in open source communities. *Electronic Markets*, *14*(2), 104-113.

Vasilescu, B., Posnett, D., Ray, B., van den Brand, M. G., Serebrenik, A., Devanbu, P., & Filkov, V. (2015). Gender and tenure diversity in GitHub teams. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems* (pp. 3789-3798).

Von Krogh, G., Spaeth, S., & Lakhani, K. R. (2003). Community, joining, and specialization in open source software innovation: a case study. *Research Policy*, *32*(7), 1217-1241.

Weikert, F., Riehle, D., & Barcomb, A. (2019). Managing commercial conflicts of interest in open source foundations. In *International Conference on Software Business*.

West, J., & Gallagher, S. (2006). Challenges of open innovation: The paradox of firm investment in open-source software. *R&D Management*, *36*(3), 319-331.

West, J., & O'Mahony, S. (2005). Contrasting community building in sponsored and community founded open source projects. In *Proceedings of the 38th Annual HICSS.*

Wheeler, B. (2004). The open source parade. *Educause Review*, *39*(5), 68-69.

Wheeler, B. (2007). Open source 2010: Reflections on 2007. *Educause Review, 42(1)*, 49-52.

Williamson, O. E. (1993). Opportunism and its critics. *Managerial and Decision Economics*, *14*, 97-107.

Yenişen Yavuz, E., Barcomb, A., & Riehle, D. (2022). Problems, solutions, and success factors in the openMDM User-Led open source consortium (Appendix). Retrieved from https://faubox.rrze.uni-erlangen.de/getlink/fiRCBgxanJUqXTBFKPLADQ4N/Appendix%20-%20Problems%20Solutions%20Success%20Factors%20of%20openMDM%20User-Led%20Open%20Source%20Consortium.pdf

Yin, R. K. (2018). *Case study research and applications*. SAGE.

Yu, Y., Benlian, A., & Hess, T. (2012). An empirical study of volunteer members' perceived turnover in open source software projects. In *2012 45th HICSS*.

Zhou, M., Mockus, A., Ma, X., Zhang, L., & Mei, H. (2016). Inflow and retention in OSS communities with commercial involvement: A case study of three hybrid projects. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, *25*(2), 1-29.

About the Authors

Elçin Yenişen Yavuz. Elçin Yenişen Yavuz, M.Sc., is a researcher and doctoral student at the Professorship for Open Source Software at Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Germany. She received her master's degree of International Information Systems from the FAU, Germany. Before joining academia, she led various projects in the automotive, healthcare and pharmaceutical industries. Her areas of interest are open innovation, open source software development, collaborative software development, and digital transformation.

Ann Barcomb. Dr. Barcomb is an assistant professor at the Schulich School of Engineering, University of Calgary. Her previous post was at Friedrich-Alexander University Erlangen-Nuremberg, Germany. Dr. Barcomb received her PhD from the University of Limerick, Ireland, in 2019, with a specialization in software engineering, and a master's in information systems from Maastricht University, The Netherlands. In the course of her industry career, she worked as a software developer for multiple firms and as a community manager for RIPE NCC. From the beginning, she has been active in free/libre/open source software, organizing events, speaking at practitioner conferences, and writing for practitioner outlets. Her research is characterized by a desire to understand and generalize processes and practices within free/libre/open source software communities, to facilitate the exchange of knowledge between practitioners.

Dirk Riehle. Prof. Dr. Dirk Riehle, M.B.A., is the Professor of Open Source Software at the Friedrich-Alexander-Universität Erlangen-Nürnberg. Before joining academia, Riehle led the Open Source Research Group at SAP Labs, LLC, in Palo Alto, California (Silicon Valley). Riehle founded the Open Symposium (OpenSym, formerly WikiSym). He was the lead architect of the first UML virtual machine. He is interested in open collaboration principles, methods, and tools, most notably open source and inner source software development. Prof. Riehle holds a Ph.D. in computer science from ETH Zürich and an M.B.A. from Stanford Graduate School of Business. He welcomes email at dirk@riehle.org, blogs at https://dirkriehle.com, and tweets as @dirkriehle.

C

Paper 3

This chapter is based on the article titled "Why do companies create and how do they succeed with a vendor-led open source foundation" originally published in *Empirical Software Engineering*:

Yenişen Yavuz, E., Riehle, D., & Mehrotra, A. (2025). Why do companies create and how do they succeed with a vendor-led open source foundation. *Empirical Software Engineering*, 30(1), 1-49. https://doi.org/10.1007/s10664-024-10588-9

Empirical Software Engineering (2025) 30:40
https://doi.org/10.1007/s10664-024-10588-9

Why do companies create and how do they succeed with a vendor-led open source foundation

Elçin Yenişen Yavuz[1] · **Dirk Riehle**[1] · **Ankita Mehrotra**[2]

Accepted: 24 October 2024 / Published online: 13 December 2024

Abstract

Vendor-led open source foundations are open source foundations led by software vendors rather than individual developers or end-user organizations. Our research investigates why vendors create or join such foundations, and how these foundations succeed. We conducted exploratory single-case study research, with the LF Edge foundation as our case. We collected qualitative data in the form of interviews and text documents, and performed qualitative data analysis for building our theory. We identified 18 motives of vendors' participation in vendor-led open source foundations regarding four aspects: revenue, competition, productivity and innovation, and reputation. To understand how vendor-led open source foundations succeed, we investigated good practices followed by LF Edge applied as preventions for potential problems or solutions for encountered problems. We determined 52 good practices in 20 different contexts, focusing on three dimensions: governance, efficiency and productivity, and sustainability.

Keywords Open source foundations · Open source projects · Best practices · Governance problems · Coopetition · Linux Foundation · Edge foundation

1 Introduction

Open source foundations are non-profit organizations that provide neutral platforms for open-source software (OSS) projects. They play a pivotal role in the collection and distribution of funds to support these projects, safeguarding the rights of project members

Communicated by: Klaas-Jan Stol

✉ Elçin Yenişen Yavuz
elcin.yenisen@fau.de

Dirk Riehle
dirk@riehle.org

Ankita Mehrotra
ankita.mehrotra@fau.de

[1] Computer Science Department, Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany

[2] Erlangen, Germany

share the interview protocol in Appendix A to ensure transparency. We present the list of documents we analyzed with their access links in Appendix B, the codebook we created during our analysis in Appendix C, and the list of code segments supporting our results in Appendix D. Appendices of this study are accessible from this link: https://faubox.rrze.uni-erlangen.de/getlink/fiUGvjNoXvGJ8acKJDfBMt/

Declarations

Conflicts of Interests All authors certify that they have no affiliations with the investigated case involving the LF Edge foundation, and the Linux Foundation.
The authors have no competing interests to disclose.

References

Ågerfalk PJ, Fitzgerald B (2008) Outsourcing to an unknown workforce: exploring opensourcing as a global sourcing strategy. MIS Q:385–409. https://doi.org/10.2307/25148845
Apereo Community Blog (n.d.) *About Apereo.* Retrieved June 20, 2024, from https://apereo.github.io/about/
Brandenburger AM, Nalebuff BJ (1996) Co-opetition, doubleday currency, New York, NY
Braun V, Clarke V (2006) Using thematic analysis in psychology. Qual Res Psychol 3(2):77–101
Bryman A (2016) Social research methods. Oxford university press
Business Wire (2022) Intel accelerates developer innovation with open software-first approach. Business Wire. Retrieved June 20, 2024, from https://www.businesswire.com/news/home/20220928005326/en/Intel-Accelerates-Developer-Innovation-with-Open-Software-First-Approach
Butler S, Gamalielsson J, Lundell B, Brax C, Sjöberg J, Mattsson A, Gustavsson T, Feist, Lönroth E (2019) On company contributions to community open source software projects. IEEE Trans Softw Eng 47(7):1381–1401
Chin KS, Chan BL, Lam PK (2008) Identifying and prioritizing critical success factors for coopetition strategy. Ind Manag Data Syst 108(4):437–454
Cloud Native Computing Foundation (2015) New Cloud Native Computing Foundation to drive alignment among container technologies. Retrieved June 20, 2024, from https://www.cncf.io/announcements/2015/06/21/new-cloud-native-computing-foundation-to-drive-alignment-among-container-technologies/
Eckert R, Stuermer M, Myrach T (2019) Alone or Together? Inter-organizational affiliations of open source communities. J Syst Softw 149:250–262
Germonprez M, Allen JP, Warner B, Hill J, McClements G (2013) Open source communities of competitors. Interactions 20(6):54–59
Grand S, Von Krogh G, Leonard D, Swap W (2004) Resource allocation beyond firm boundaries: a multi-level model for open source innovation. Long Range Plan 37(6):591–610
Guba EG (1981) Criteria for assessing the trustworthiness of naturalistic inquiries. Ectj 29(2):75–91
Guion RM (2011) Assessment, measurement, and prediction for personnel decisions. Taylor & Francis. https://doi.org/10.4324/9780203836767
Hunter P, Walli S (2013) The rise and evolution of the open source software foundation. IFOSS l Rev 5:31
Iivari N, Hedberg H, Kirves T (2008) Usability in company open source software context-Initial findings from an empirical case study. IFIP international conference on open source systems. Springer US, Boston, pp 359–365
Intel Corporation (2021) Open source software: fact sheet. Intel Corporation. Retrieved June 20, 2024, from https://download.intel.com/newsroom/2021/client-computing/OSS-Fact-sheet.pdf
Izquierdo JLC, Cabot J (2020) A survey of software foundations in open source. *arXiv preprint arXiv:2005.10063*
Joo C, Kang H, Lee H (2012) Anatomy of open source software projects: evolving dynamics of innovation landscape in open source software ecology. The 5th International Conference on Communications, Computers and Applications (MIC-CCA2012), Istanbul. IEEE. pp 96–100
Kumar A, Connell J, Bhattacharyya A (2021) Co-opetition for corporate social responsibility and sustainability: drivers and success factors. Sustainability Accounting, Management and Policy Journal 12(6):1208–1238
Linåker J, Regnell B (2020) What to share, when, and where: balancing the objectives and complexities of open source software contributions. Empir Softw Eng 25:3799–3840
Microsoft (n.d.a) Visual Studio Code license and privacy statement. Retrieved June 20, 2024, from https://code.visualstudio.com/license

Microsoft (n.d.b) Visual Studio Code [Repository]. GitHub. Retrieved June 20, 2024, from https://github.com/microsoft/vscode

Open Logistics Foundation (n.d.) Mission | Rethinking open source for logistics. Joining forces. Retrieved June 20, 2024, from https://openlogisticsfoundation.org/about/mission/

OpenInfra Foundation (n.d.) Organizations supporting the OpenInfra Foundation. Retrieved June 20, 2024, from https://openinfra.dev/members/

Petter RRH, Resende LM, de Andrade Júnior PP, Horst DJ (2014) Systematic review: an analysis model for measuring the coopetitive performance in horizontal cooperation networks mapping the critical success factors and their variables. Ann Reg Sci 53:157–178

Pinto G, Steinmacher I, Dias LF, Gerosa M (2018) On the challenges of open-sourcing proprietary software projects. Empir Softw Eng 23:3221–3247

Riehle D (2010) The economic case for open source foundations. Computer 43(01):86–90

Riehle D, Berschneider S (2012) A model of open source developer foundations. IFIP international conference on open source systems. Springer Berlin Heidelberg, Berlin, pp 15–28

Roberts RE (2020) Qualitative interview questions: guidance for novice researchers. Qual Rep 25(9). https://doi.org/10.46743/2160-3715/2020.4640

Schaarschmidt M, Stol K-J (2018) 'Company soldiers and gone-natives: role conflict and career ambition among firm-employed open source developers', forthcoming conference, ICIS 2018, San Francisco, 13–16 December

Schaarschmidt M, Bertram M, von Kortzfleisch HF (2011) Exposing differences of governance approaches in single and multi vendor open source software development. In IFIP International Working Conference on Governance and Sustainability in Information Systems-Managing the Transfer and Diffusion of IT.

Shaikh M, Cornford T (2010) 'Letting go of control to embrace open source: implications for company and community. In: 2010 43rd Hawaii international conference on system sciences. IEEE, pp 1–10. https://doi.org/10.1109/HICSS.2010.1

Stack Overflow (2023) Stack overflow developer survey 2023: most popular technologies - Integrated development environment. Retrieved June 20, 2024, from https://survey.stackoverflow.co/2023/#section-most-popular-technologies-integrated-development-environment

Strauss A, Corbin J (1990) Basics of qualitative research. Sage Publications, Thousand Oaks

Teixeira J, Lin T (2014) Collaboration in the open-source arena: the webkit case. In: Proceedings of the 52nd ACM conference on Computers and people research. pp 121–129. https://doi.org/10.1145/2599990.2600009

Teixeira J, Mian S, Hytti U (2016) Cooperation among competitors in the open-source arena: the case of OpenStack. *arXiv preprint arXiv:1612.09462*

Weikert F, Riehle D, Barcomb A (2019) Managing commercial conflicts of interest in open source foundations. International conference on software business. Springer International Publishing, Cham, pp 130–144

West J, Gallagher S (2006) Challenges of open innovation: the paradox of firm investment in open-source software. R&D Manag 36(3):319–331

YenişenYavuz E, Barcomb A, Riehle D (2022) Problems, solutions, and success factors in the openMDM user-led open source consortium. Commun Assoc Inf Syst 51(1):13

Yenişen Yavuz E, Riehle D, Mehrotra A (2024) Why do companies create and how do they succeed with a vendor-led open source foundation (External Appendix) [Online]. Available: https://faubox.rrze.uni-erlangen.de/getlink/fiUGvjNoXvGJ8acKJDfBMt/

Yin RK (2018) Case study research and applications. Sage Publications, Thousand Oaks

Zhang Y, Zhou M, Mockus A, Jin Z (2021) Companies' participation in OSS development–an empirical study of openstack. IEEE Trans Software Eng 47(10):2242–2259

Zhang Y, Liu H, Tan X, Zhou M, Jin Z, Zhu J (2022a) Turnover of companies in OpenStack: prevalence and rationale. ACM Trans Softw Eng Methodol (TOSEM) 31(4):1–24

Zhang Y, Zhou M, Stol KJ, Wu J, Jin Z (2020) How do companies collaborate in open source ecosystems? An empirical study of openstack. In: Proceedings of the ACM/IEEE 42nd international conference on software engineering. pp 1196–1208. https://doi.org/10.1145/3377811.3380376

Zhang Y, Stol KJ, Liu H, Zhou M (2022b) Corporate dominance in open source ecosystems: a case study of OpenStack. In: Proceedings of the 30th ACM joint european software engineering conference and symposium on the foundations of software engineering. pp 1048–1060. https://doi.org/10.1145/3540250.3549117

Zhou M, Mockus A, Ma X, Zhang L, Mei H (2016) Inflow and retention in oss communities with commercial involvement: a case study of three hybrid projects. ACM Trans Softw Eng Methodol (TOSEM) 25(2):1–29

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

Elçin Yenişen Yavuz is a researcher and doctoral student at the Professorship for Open Source Software at Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Germany. She received her master's degree in International Information Systems from FAU, Germany. Before joining academia, she led various projects in the automotive, healthcare, and pharmaceutical industries. Her research interests include open innovation, open source foundations, open source software communities, collaborative software development, and digital transformation.

Dirk Riehle is a professor of computer science at University of Erlangen. They are also the CEO of Bayave GmbH, a training and consulting firm. Their work helps companies succeed in and through software, with a specialization in open source, inner source, and product strategy. Before joining academia, Prof. Riehle led the open source research group at SAP Labs in Palo Alto, California (Silicon Valley). They also worked for software startups and large corporations in Boston, MA and Zurich, Switzerland, as a software developer, architect, and engineering manager. Riehle holds a Ph.D. in computer science from ETH Zurich and an M.B.A. from Stanford Graduate School of Business. They welcome email at dirk@riehle.org, blog at https://dirkriehle.com, and tweet as @dirkriehle.

Ankita Mehrotra is a change manager at Siemens HealthineersForchheim, Germany. She received her master's degree in International Information Systems from FAU, Germany and Bachelor of Technology (Computer Science Engineering) from India. Before joining her master's program, she worked in roles of IT Analyst and System Engineer in the IT industry for around 6 years. Her areas of interest are change management (process and product), service innovation, and digitalization in the workplace.

D

Paper 4

This chapter is based on the article titled "Governance Practices for Open Source Foundations in the Healthcare Sector" originally published in the proceedings of the *International Conference on Software Business* (ICSOB) 2024, published by Springer in the Lecture Notes in Business Information Processing series:

Yenişen Yavuz, E., Shrivastava, A., Riehle, D., Putz, F. (2025). Governance Practices for Open Source Foundations in the Healthcare Sector. In: Papatheocharous, E., Farshidi, S., Jansen, S., Hyrynsalmi, S. (eds) Software Business. ICSOB 2024. Lecture Notes in Business Information Processing, vol 539. Springer, Cham. https://doi.org/10.1007/978-3-031-85849-9_30.