Is my Dependency Sustainable? A Systematic Mapping on Open Source Software Sustainability

Gabriel Silva Fontes
g.fontes@usp.br
University of São Paulo
São Carlos, Brazil

Vinicius dos Santos
vinicius.dos.santos@usp.br
University of São Paulo
São Carlos, Brazil

Elisa Yumi Nakagawa
elisa@icmc.usp.br
University of São Paulo
São Carlos, Brazil

Abstract

Context: With growing software reuse and interoperability, Open Source Software (OSS) is more ubiquitous than ever. Often sourced for free from volunteer-driven work, many developers lack awareness of the sustainability of OSS they depend on. Adopting OSS components while building software systems has long-lasting consequences, not only bringing benefits, but also risks. Understanding the sustainability of OSS dependencies thus becomes a critical activity during software design and development. However, OSS sustainability is not a very well-defined concept, thus making research and decisions regarding OSS adoption more difficult. **Objective**: Synthesize how the wider concept of OSS sustainability is currently understood "in the wild", explore how it can be characterized, and discuss the impact a better understanding can have on more responsible OSS adoption. **Method**: We conducted a systematic mapping study (SMS) across 111 studies from the literature to extract sustainability understandings, synthetized into aspects of sustainability in OSS projects. **Results**: Social and technical concerns are well-explored and reaffirm the importance of the OSS community. Individual, economic, and environmental definitions remain relatively underexplored. There is a growing usage of the term "sustainability" in OSS research. We propose 19 aspects that can be utilized to understand OSS sustainability. **Conclusion**: Those aspects can be utilized by practitioners who rely on OSS to better understand the projects they depend upon, as well by researchers exploring this ecosystem.

CCS Concepts

• **Software and its engineering** → **Open source model**; *Risk management*; *Software evolution*; *Maintaining software.*

Keywords

Open Source, Open Source Software, Sustainability

ACM Reference Format:
Gabriel Silva Fontes, Vinicius dos Santos, and Elisa Yumi Nakagawa. 2026. Is my Dependency Sustainable? A Systematic Mapping on Open Source Software Sustainability. In *3rd International Workshop on Designing Software (Designing '26), April 12–18, 2026, Rio de Janeiro, Brazil.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3786152.3788587

ACM ISBN 979-8-4007-2385-8/2026/04
https://doi.org/10.1145/3786152.3788587

1 Introduction

Open Source Software (OSS) is widely used in a variety of activities in the modern economy [26]. Present in most business domains, powering critical software infrastructure [6], and prompting deep changes to quality through commodification pressure [23], even in ecosystems dominated by proprietary software [1, 27]. OSS has become a sort of "ethereal" resource: ever-present, deeply involved with everything it touches, and unnoticed by untrained eyes [6, 7].

Software Engineering (SE) researchers and practitioners foresee an ever-present risk with OSS long-term sustainability, as more than 80% of OSS projects are eventually abandoned [22]. It is a challenge to assure the continued existence and quality of a resource that is, by its nature, ever flowing, changing, and almost always distributed without warranties, making OSS adoption a risky decision often made in a non systematic manner. With most programming languages, frameworks, and deployment/orchestration tools being OSS, adopting OSS is frequently an architectural decision, thus hard to make and costly to change. As with all such decisions, the cheapest moment to make them is during the initial phases of software design, so evaluating the sustainability of OSS components becomes a critical design activity for practitioners.

Software sustainability is a relatively recent concept that encompasses multiple aspects in software development/maintenance [18]. Sustainable software aims to not only meet present needs but enable future generations to continue creating and maintaining this software [2]. Sustainability is described in literature as a systemic concept that cannot be comprehended without systemic thinking [25]. To help researchers and software engineers understand, software sustainability dimensions are often used. The Karlskrona Manifesto defines five main dimensions for sustainability: economic, social, technical, individual, and environmental [2].

From a social perspective, sustainable software aims to ensure equitable access to resources and opportunities for all individuals and communities [11]. Economically, it involves preserving long-term value and capital for stakeholders while minimizing risks and maximizing returns [11, 20]. Environmentally, it focuses on minimizing the consumption of natural resources and mitigating environmental impacts associated with software activities [11]. Individually, it prioritizes the well-being and development of human beings, encompassing aspects such as mental and physical health, education, and mobility [2, 17]. Finally, from a technical standpoint, sustainable software systems are designed for longevity, adaptability, and evolution in changing environments [11].

Achieving sustainability in software development involves addressing various challenges through the same five dimensions, including energy efficiency, social equity, economic viability, and environmental impact [9]. It requires integrating sustainability considerations into all phases of the software life-cycle, from requirements gathering and design to deployment and maintenance [24]. Overall, being sustainable in the software development context means creating products that not only fulfill their intended purpose but also minimize negative impacts on society, economy, and the environment, thus continuing their existence for a longer period.

Open Source Software (OSS) aligns closely with the social dimension of sustainability, as it helps create a more equitable software development landscape. Social dimension emphasizes the preservation of communities that maintain and improve OSS, ensuring users have access to high-quality, efficient, and freely available software [2]. OSS is typically sustained by self-organized communities [14] whose goal is to safeguard the ability of future developers to contribute and evolve the software. However, sustainability is a multifaceted concept that is not well-defined in the context of OSS, making it harder for adopters to evaluate the sustainability of their OSS dependencies while designing software systems. The existing body of knowledge on sustainability in OSS remains fragmented across many studies.

The research question (RQ) that has motivated this work is: "*How is sustainability understood in the OSS context?*". Several other secondary studies were published about software sustainability [3, 15, 18]. Linåker et al. [13] published a secondary study on OSS health, which is related to sustainability, but not quite the same. Other authors (e.g., [4, 12]) conduct a longitudinal study on OSS project sustainability that examines the factors influencing the long-term sustainability of OSS, but do not map the literature insights about sustainability in this domain. Therefore, developing a clearer understanding of what software sustainability means in the OSS context is essential to tighten the relations between the OSS community, sustainability researchers, and practitioners trying to better understand their potential OSS dependencies.

This paper aims to map the understandings and aspects that define an OSS project as more sustainable. To do this, we performed a Systematic Mapping Study (SMS) that puts together 111 studies to extract (i) the current understanding of sustainability in OSS projects; (ii) Aspects that comprise or contribute to OSS sustainability; and (iii) future directions and insights from literature. The main contribution of this paper is to present a broader view of which aspects make up OSS sustainability. We propose 19 sustainability aspects of OSS projects that were classified into the five sustainability dimensions. In addition, those aspects revealed attention spots, such as the environmental concerns that are underexplored in the literature. We note that OSS culture influences the investigation, and it becomes a great opportunity to raise awareness on the overall sustainability problem in OSS, given how widely adopted OSS projects are.

The paper is structured as follows: Section 2 presents the research method. Section 3 describes the results of the SMS. Section 4 discusses the SMS results, the threats to validity of our work, and actions performed to mitigate them. Section 5 provides the final considerations and future research perspectives.

2 Research Method

This study is a SMS that follows the guidelines of Petersen et al. [19]. We follow the three-step process that encompasses planning, conduct, and results reporting, as shown in Figure 1. This section presents the first two steps (planning and conduct), outlining the most important elements of the SMS protocol. Section 3 presents data synthesis results.

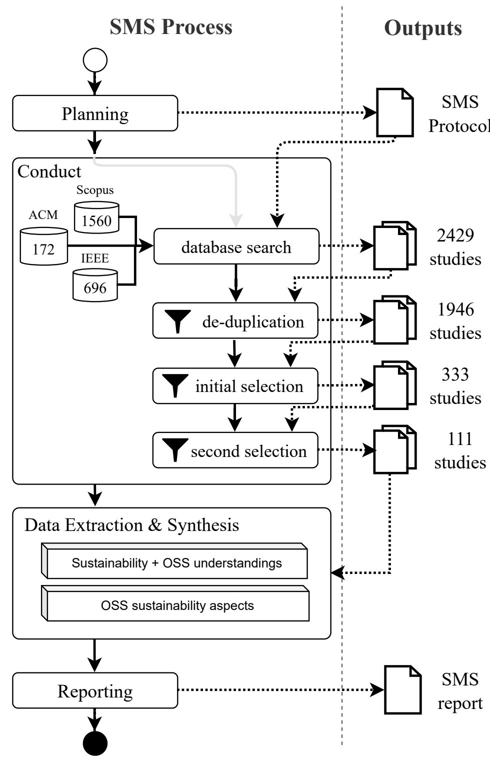

Figure 1: SMS research method workflow

2.1 Planning

This section reports important elements of our SMS protocol that enable further auditions and reproduction/updates. First, we defined our goal using the GQM (Goal Question Metric) [10] approach as: analyzing primary studies *for the purpose of* identifying how sustainability has been understood/addressed in Open Source Software (OSS) *from the point of view of* SE researchers *in the context of* scientific literature. Based on this goal, we defined our research questions (RQs):

RQ1: How is sustainability understood in the OSS domain?
RQ2: Which aspects make an OSS projects more sustainable?

Our search string defines sustainability and OSS as major domains. As we intended to locate studies that address "sustainability" within "OSS", we joined the two with the 'AND' operator. In addition, as OSS frequently appears as one of its synonyms, we joined those with the 'OR' operator. Our final string was: `("sustainability") AND ("open source" OR "free software" OR "FLOSS" OR "OSS")`. This string was developed iteratively and pilot-tested to ensure sensitivity (capturing relevant studies) and precision (excluding irrelevant studies).

We adopted three databases (Scopus[1], ACM Digital Library[2], and IEEE Digital Library[3]), considering they are the well-known and widely used publication databases in the computing area. For the selection, one inclusion criterion (IC) and five exclusion criteria (EC) were defined to ensure a transparent and replicable process: **IC1**: The study addresses sustainability and OSS. **EC1**: Full text of the study is not available. **EC2**: Study is a shorter version of another. **EC3**: Study is a summary of a conference/workshop. **EC4**: Study is not peer-reviewed. **EC5**: Study is written in a language other than English.

2.2 Conducting

Studies collection in electronic databases started by adapting the generic search string to each database. Scopus permits searching three metadata fields (title, abstract, and keywords) simultaneously. ACM Digital Library and IEEE require executing the search separately for each metadata field and combining them into a single studies dataset. We did not limit our search to publication period or field (like Computer Science), aiming to reach as many studies as possible.

Figure 1 illustrates the search process and scrutiny. Our systematic search initially retrieved a total of 2,429 studies (1,561 from Scopus, 172 from ACM Digital Library, and 696 from IEEE Digital Library). Following deduplication based on DOI, and when necessary, title and year, 1,948 unique studies remained. A first selection was conducted by screening titles, abstracts, and keywords against the inclusion and exclusion criteria, resulting in 334 studies for full-text assessment. During the second selection, all 334 studies were fully read and re-evaluated according to the same criteria, yielding 111 unique studies that are presented in Table 1. The selection process counts on a main researcher who was responsible for performing the first selection. In addition, an experienced researcher in sustainability and the SMS process double-checked the selections; finally, conflicts were resolved in consensus meetings.

The synthesis process for identifying OSS sustainability aspects involved a systematic three-step approach. First, fragment extraction was performed, where relevant portions of each study that discussed sustainability were carefully located. To support this process, we used an extraction form that recorded paper metadata (DOI, title, authors, publication year, venue, and venue type), along with the sustainability-related fragments. The extracted data was synthesized using thematic analysis [5], applying both open and axial coding. Based on this, we developed a coding scheme for qualitative synthesis, classifying sustainability-related concepts in OSS.

Our first step in the analysis was to select fragments and extract terms from all the studies obtained through the SMS. The terms were deduplicated and factored into a set of what we called sustainability aspects. These aspects were further grouped according

Table 1: Set of studies selected

ID	Authors	Year	ID	Authors	Year
S001	Mora-Cantallops et al.	2020	S044	Zhang et al.	2023
S002	Suleimenov et al.	2020	S045	Barcomb et al.	2020
S003	Mahaux et al.	2015	S046	Barcomb et al.	2022
S004	Wiggins et al.	2010	S047	Li et al.	2022
S005	Arantes et al.	2011	S048	Yue et al.	2023
S006	Robles et al.	2012	S049	Tan et al.	2023
S007	Petrinja et al.	2012	S050	Macho et al.	2013
S008	Barham et al.	2012	S051	Steinmacher et al.	2014
S009	Riehle et al.	2012	S052	Zhou et al.	2017
S010	Nyman et al.	2012	S053	Valiev et al.	2018
S011	Calefato et al.	2022	S054	Coelho et al.	2018
S012	Nov et al.	2008	S055	Gasparini et al.	2019
S013	Poba-Nzaou et al.	2019	S056	Qiu et al.	2019
S014	Liu et al.	2020	S057	Tan et al.	2020
S015	Ghapanchi et al.	2015	S058	Alami et al.	2020
S016	Santos et al.	2013	S059	Alami et al.	2020
S017	Gamalielsson et al.	2014	S060	Trinkenreich et al.	2020
S018	Butler et al.	2020	S061	Yin et al.	2021
S019	Hata et al.	2015	S062	Zhang et al.	2022
S020	Wang et al.	2022	S063	Shimada et al.	2022
S021	Jensen et al.	2011	S064	Fang et al.	2022
S022	Carillo et al.	2014	S065	Xiao et al.	2022
S023	Sajadi et al.	2023	S066	Gray et al.	2022
S024	He et al.	2023	S067	Zhang et al.	2022
S025	Ye et al.	2003	S068	Ramchandran et al.	2022
S026	Zanetti et al.	2012	S069	Zhou et al.	2022
S027	Izquierdo et al.	2015	S070	Zhang et al.	2022
S028	Barcomb et al.	2019	S071	Stanciulescu et al.	2022
S029	Dias et al.	2021	S072	Dam et al.	2023
S030	Trinkenreich et al.	2023	S073	Yin et al.	2022
S031	Yin et al.	2023	S074	Nakakoji et al.	2002
S032	Guizani et al.	2023	S075	Onoue et al.	2016
S033	Hasselbring et al.	2020	S076	Chengalur-Smith et al.	2010
S034	Zhang et al.	2022	S077	Papamichail et al.	2021
S035	Bird et al.	2007	S078	Liao et al.	2018
S036	Robles et al.	2009	S079	Perens et al.	2005
S037	Gupta et al.	2017	S080	Ju Long et al.	2005
S038	Rastogi et al.	2016	S081	Carillo et al.	2013
S039	Horiguchi et al.	2021	S082	Carillo et al.	2013
S040	Chouchen et al.	2021	S083	Atiq et al.	2016
S041	Xia et al.	2023	S084	Chua et al.	2017
S042	Terceiro et al.	2010	S085	Vainio et al.	2006
S043	Goggins et al.	2021	S086	Sethanandha et al.	2010
S087	Kritikos et al.	2024	S088	Schwarts et al.	2024
S089	O'Neil et al.	2024	S090	Chang et al.	2024
S091	Xiao et al.	2023	S092	Sonabend et al.	2024
S093	Han et al.	2024	S094	Wattanakriengkrai et al.	2023
S095	Nguyen et al.	2023	S096	Sokyina et al.	2024
S097	Santos, Italo	2024	S098	Khan et al.	2024
S099	Zhang et al.	2024	S100	Jahn et al.	2024
S101	Fang et al.	2024	S102	Feng et al.	2024
S103	Han et al.	2024	S104	Linåker et al.	2024
S105	Wang et al.	2023	S106	Wilson, Katrina	2023
S107	Fang et al.	2023	S108	Hovhannisyan et al.	2024
S109	Adejumo et al.	2024	S110	Wattanakriengkrai et al.	2024
S111	Steinmacher et al.	2014			

[1] https://scopus.com
[2] https://dl.acm.org
[3] https://ieeexplore.ieee.org

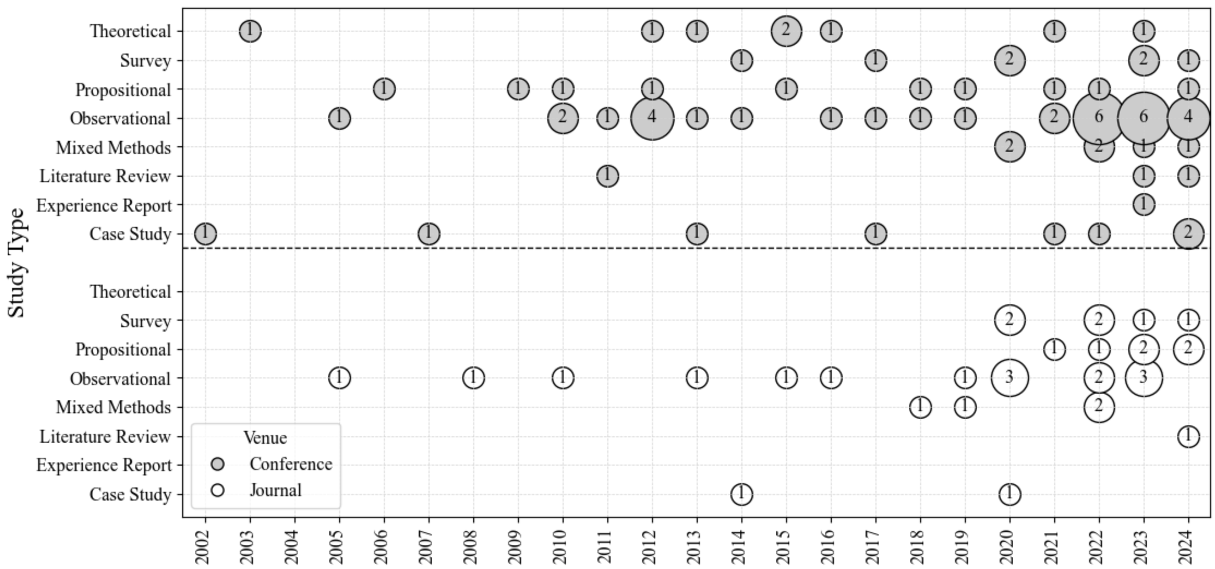

Figure 2: Overall view of studies (distribution by publication year, venue type, and study type)

to their sustainability dimension (social, technical, individual, economic, environmental). We then show how the studies our SMS selected were classified according to these aspects in Section 3.2.

The full replication kit, containing a complete list of studies, selection decisions, raw data extracted, and synthesized data is available in the external material.

3 Results

This section presents the data synthesis from the selected studies, followed by the answers to RQs.

3.1 Overview of Studies

Figure 2 shows the distribution of studies according to their publication year, venue, and type. The studies span 21 years, starting in 2002, with the majority of publications concentrated in the last few years.

The growing understanding of Software Sustainability as a multidisciplinary and systemic concept could explain the increasing number of publications. Many of the individual concerns have been previously researched without an explicit mention of sustainability itself. Concerning publication venues, around two-thirds of studies were published in event proceedings (i.e., 77), while one-third (i.e., 33) were published in journals, and a single book chapter. The increasing number of journal papers may indicate that the topic is becoming more mature over time.

3.2 Sustainability Aspects

Our sustainability aspects were derived from the set of OSS sustainability terms that our SMS extracted. Their intent is to be able to fully express every sustainability concern our selected studies address, while minimizing overlap. The aspects are grouped within established software sustainability dimensions (social, technical, individual, economic, and environmental). These are defined as follows:

Social Dimension. This dimension covers the human and community aspects of open-source software (OSS) projects. It emphasizes the interactions, governance, and social structures that sustain participation and collaboration.

(1) **Contributor Retention**: Also called "stickiness"[8]. The probability of continued contributions over time. It reflects the project's ability to keep its contributors engaged and motivated.
(2) **Contributor Attraction**: Also called "magnetism"[8]. The ability to attract new contributors is often linked to the project's popularity, niche, and functionality, as most contributors are also users/stakeholders.
(3) **Internal Communication**: The effectiveness of communication among peers fosters a sense of community and belonging. It is related to governance transparency.
(4) **Governance Structure**: The structure, transparency, legitimacy, and consistency of leadership, regarding decision-making and project management.
(5) **Openness/Onboarding**: The onboarding process and the project's attitude toward new contributors. This is highly related to governance and heavily affects contributor retention.
(6) **Project Legitimacy**: The perception of the project, which can sometimes be enhanced through marketing. It is composed of smaller characteristics, such as popularity, niche, and age. Niche size interacts with other aspects (such as quality and feature set), increasing popularity. A popular project attracts users who may become contributors, forming a virtuous cycle [21]. Age, when combined with other aspects (e.g. sustained activity and maintainability) is a frequently used and accurate measure for sustainability.

(7) **External Communication/Ecosystem**: The project's relationship with outside projects and its cooperation with external entities, which can enhance features and quality (e.g. through interoperability).

Technical Dimension. This dimension highlights the software engineering aspects of OSS projects. It focuses on the practices, architecture, and development processes that determine the project's long-term sustainability and adaptability.

(1) **Sustained Activity**: One of the most frequently used measures. The ongoing software development activity, including the production of artifacts (release rate, issues, patches) and response times. Heavily influenced by contributor retention and governance; contributes to retention and attraction.
(2) **Functionality, Scope**: The relevance of functional requirements to stakeholders and the project's ability to evolve and incorporate new features over time. Having exclusive functionality on a niche that usually lacks frequently leads to increased project popularity and legitimacy.
(3) **Quality Assurance**: The processes for bug resolution, adherence to quality standards, and testing, which contribute to the software's maintainability and increases its capability of incorporating new features without regressions, which otherwise often cost legitimacy.
(4) **Methodologies, Practices**: The development methodologies and practices adopted by the project, which influence its maintainability, contributor retention, and quality.
(5) **Architecture, Maintainability**: The design of the software architecture, including modularity and code complexity, it affects contributor retention, leads to increased quality and is the limiting factor for development scalability (thus impacting feature set).
(6) **Licensing**: Licensing policies, usually static over the project lifetime, impacting how the software can be used and contributed to. Similarly to governance, it interacts in a very complex way with other aspects. However, there is no conclusive research linking specific license types (e.g. permissive VS copyleft) to higher sustainability.

Individual Dimension. This dimension captures the characteristics of individual contributors. It considers their motivations, knowledge, and skills, which directly affect the quality and direction of the project.

(1) **Personal Motivations**: The individual motivations of contributors, which are often influenced by the sense of community, project niche, and feature set. Influences attraction and retention.
(2) **Personal Knowledge**: The technical expertise and skills of individual contributors. It is affected by project niche/size and affects the quality and feature set of the project.

Economic Dimension. This dimension deals with the financial aspects of sustainability in OSS projects. It highlights the availability, management, and monetization of resources that enable long-term development.

(1) **Financial Resources**: The attraction and management of financial resources, including foundations, crowdfunding, and sponsorships. This influences and is influenced by internal monetization and governance.
(2) **External Monetization**: The ways external actors monetize their involvement with the project and their actions towards it. Includes paid contributors. Highly related to external communication and openness.
(3) **Internal Monetization**: The use of project funds to pay contributors, often seen in projects with corporate governance. The backing organization is, therefore, a heavy factor in the project's continued development.

Environmental Dimension. This dimension focuses on the environmental implications of OSS projects.

(1) **Environmental Requirements**: The concerns related to the project's environmental impact. It is sometimes seen in the form of regulatory compliance (thus relates to governance).

Table 2 shows an overview of our aspects, and which studies address each of them. We can see that concerns overwhelmingly fit the technical and/or social sustainability dimensions, especially concerns that are frequently used as a measure of sustainability (e.g. sustained activity and contributor retention[4]). The technical and social concerns are also closely related, as they form a virtuous cycle[21] of project legitimacy, attraction, and activity[4]. Corporate involvement is also a relevant concern, as it bridges the economic (internal/external monetization) and social (governance[16]) dimensions.

3.3 OSS sustainability understandings

Sustainability in the OSS domain encompasses a project's long-term viability and ability to continue delivering value (S010, S033, S098). It refers to the software potential to maintain the functionality and evolution of the software system in the future (S044). A core characteristic of sustainable software is its ability to exhibit software development and maintenance activity over the long term (S013, S076), ensuring it remains active for a longer period of time (S076) and can evolve to incorporate new demands over time (S029). Sustainability also means creating the necessary conditions for future generations to use and evolve present artifacts (S077).

From a technical perspective, sustainability is closely tied to the software's inherent qualities and structure. Projects need to maintain high-quality standards and implement rigorous methodologies (S008). Being technically sustainable means overcoming inherent challenges of OSS like software complexity and its non-uniform evolution, which can pose threats (S001, S013). Long-living projects usually adopt modular structures and less complex code are incentives for developers and contribute significantly to maintainability (S005, S026, S042, S053). Also, other factors can contribute to sustaining OSS, for instance, the project's ability to reduce bugs and ensure stability (S013), along with a high probability of new revisions being published and consistent maintenance (S007, S071, S095), are also important for tecnical sustainability.

A cornerstone of OSS sustainability is the underlying community of contributors (S002, S017, S025, S074). This involves the continuous attraction of new and retention of existing contributors (S013, S021, S024, S030, S037, S039, S099, S102, S107), recognizing that

Table 2: OSS sustainability aspects

Dimension	ID	Aspect	Studies
Social	A1	**Contributor Retention**	S012, S013, S016, S017, S019, S025, S028, S030, S036, S037, S040, S045, S046, S047, S048, S049, S050, S051, S053, S056, S058, S059, S065, S066, S072, S074, S075, S076, S081, S082, S086, S087, S090, S092, S097, S098, S099, S102, S107, S111
	A2	**Contributor Attraction**	S005, S013, S015, S016, S019, S021, S024, S030, S031, S037, S039, S043, S047, S051, S053, S056, S059, S064, S065, S071, S073, S076, S099, S102, S104, S107
	A3	**Internal Communication**	S002, S003, S012, S013, S014, S017, S022, S023, S027, S029, S003, S030, S031, S043, S061, S068, S069, S071, S073, S076, S081, S083, S084, S085, S087, S092, S096, S097, S100, S102, S104, S106, S111
	A4	**Governance Structure**	S012, S013, S014, S017, S027, S029, S031, S043, S057, S058, S060, S061, S065, S071, S072, S073, S080, S084, S085, S009, S087, S089, S097, S102, S104, S105
	A5	**Openness/Onboarding**	S003, S035, S041, S043, S055, S056, S057, S058, S059, S060, S067, S070, S078, S081, S082, S087, S097, S099, S102, S104, S109, S110, S111
	A6	**Legitimacy**	S006, S007, S015, S043, S053, S064, S071, S076, S078, S080, S088, S095, S098, S101, S107
	A7	**External Communication/Ecosystem**	S012, S013, S014, S043, S045, S006, S069, S071, S087, S104, S107, S110
Technical	A8	**Sustained Activity**	S004, S007, S010, S011, S013, S015, S016, S018, S033, S034, S038, S041, S043, S054, S068, S071, S072, S073, S076, S078, S088, S091, S092, S093, S095, S098, S103
	A9	**Functionality/Scope**	S003, S010, S013, S015, S017, S018, S029, S032, S044, S053, S071, S077, S078, S079, S087, S088, S092, S095
	A10	**Quality Assurance**	S008, S010, S013, S015, S018, S029, S040, S041, S043, S058, S059, S064, S071, S087, S092
	A11	**Methodologies/Practices**	S001, S004, S005, S008, S019, S040, S043, S052, S058, S065, S071, S086, S087, S090, S092, S094, S108, S111
	A12	**Architecture/Maintainability**	S001, S005, S015, S026, S042, S052, S053, S077, S087, S094, S111
	A13	**Licensing**	S017, S013, S015, S087, S092
Individual	A14	**Personal Motivations**	S012, S018, S056, S058, S069, S083, S110
	A15	**Personal Knowledge**	S015, S018, S002, S085, S100, S102, S106, S111
Economic	A16	**Financial Resources**	S013, S015, S016, S020, S032, S043, S005, S062, S063, S064, S076, S085, S087, S089, S092, S104, S110
	A17	**External Monetization**	S012, S014, S015, S034, S052, S067, S070, S079, S087, S092, S099, S104, S106
	A18	**Internal Monetization**	S019, S020, S079, S083, S087, S092
Environmental	A19	**Environmental Requirements**	S003

contributor disengagement threatens sustainability (S066, S090). Essential social aspects include the knowledge and skills contributors bring (S002, S012), fostering community growth (S005, S043, S102), building interpersonal trust (S023), ensuring fairness and acknowledging contributors' motivations (S012, S083, S096, S106). Effective governance rules (S027, S031, S055, S058, S070, S072, S089) with clear leadership (S017), a participative development process (S003), and transparent communication (S013, S029, S074, S100, S104) are vital. Mentoring and support to newcomers (S048, S049, S058, S060, S071, S102) along with promoting an open and inclusive atmosphere and diversity (S061, S092, S097, S099, S102, S104) are also critical for retaining members and promoting long-term success.

Despite OSS having a facilitated distribution due to its free licenses, financial resources (S005, S043, S062, S063, S084, S092), sponsorships, and fundraising (S013, S019, S020, S064, S095, S104) are important for project progress and sustainability. The increasing commercial participation and firm involvement in OSS projects have significant consequences for sustainability (S014, S034, S052, S053, S067, S070, S078, S082, S089, S099), bringing benefits, as well as new challenges, to OSS communities. Sustainability plays an important role in the overall open-source ecosystem (S069, S094, S107), which means that OSS is part of long and complex chain of software dependencies that depends on financial support directly or indirectly.

On an individual level, it is heavily influenced by the contributors themselves, particularly their knowledge and skills (S002) and their intrinsic motivations to participate (S012, S083, S096, S106). Factors such as fairness and perceptions of justice within the project community significantly affect contributors' willingness to engage (S012, S083). A developer's feelings of identity and belonging are also crucial for their retention and the project's long-term survival (S030), while the presence of toxic behaviors among developers can critically jeopardize community sustainability (S096, S104). From an environmental perspective, sustainability is understood to include broader ecological and social requirements (S003), which is not incorrect, but represents a disconnection between OSS development and other environmental aspects.

4 Discussion

This section presents our main findings, outlines some future work, and discusses the threats to the validity of this work, along with the countermeasures we have applied to address them.

4.1 Main Findings

The main findings resulting from our investigation are:

- **A way to evaluate OSS sustainability while designing software**: Our proposed aspects' main intended usage is as a starting point for more specific evaluation frameworks (e.g. with actual metrics). With that said, we believe that these are already useful to practitioners as high-level guidance on key concerns they should be on the look out for while adopting OSS dependencies. We propose that further discussion and research about how OSS sustainability could be evaluated in a more systematic way during software design phases should follow.
- **Sustainability of OSS is uncontrollable, but can be understood**: The first step for organizations when designing

their applications is to recognize that they have no significant control (in most cases) over OSS projects. Considering the wide range of factors affecting sustainability across various dimensions, only systemic and high-level actions have a direct impact on the sustainability of OSS. However, designers can and should understand all aspects that affect the sustainability of projects used as dependencies.

- **Sustainability aligns with OSS culture and values:** OSS is deeply rooted in a culture of collaboration and knowledge sharing, rather than profit-seeking. This culture seems to directly shape the factors that contribute to project sustainability. Introducing sustainability concerns to OSS development community could strengthen OSS, as economic and environmental dimensions, which often conflict in proprietary software development, tend to be more aligned within OSS.
- **Privileged and underexplored dimensions:** Current research tends to prioritize technical and social aspects, with limited attention to individual, economic, and particularly environmental dimensions. The scarcity of studies addressing environmental issues within OSS hints that the relation between the OSS communities and the environmental impacts of their software has not yet been well explored.
- **Sustainability systemic effects awareness:** Adopting a sustainability/aware vision aligns OSS projects with long-term goals. While most research emphasizes sustainability in software i.e., enhancing maintainability and system endurance, domain-specific studies highlight sustainability by software, showing how software can support social, environmental, and organizational goals. This approach encourages system designers and architects to consider the broader systemic effects of their decisions, integrating sustainability into every stage of software engineering.

4.2 Future Work

The future is bright for further research within OSS sustainability. OSS sustainability is different enough from general software sustainability to warrant its own specificities, currently being matured by the research community to move towards widely accepted understandings. Thus research into any of the aspects will be helpful to further their current understanding.

The diversity of sustainability understandings might warrant surveys to gauge how researchers and practitioners from different segments understand the concept, helping understand whether a wide consensus is possible and, if so, what it would look like depending on the demographic.

Our proposed aspects are theoretical, thus they provide mostly a conceptual framework for practitioners to think about OSS sustainability, whether they are adopters or contributors. Researchers can build upon these aspects to create more specific guidance and/or metrics to complement them, working as tools that can be directly used by practitioners to evaluate OSS sustainability.

This study's results can also be useful for future mappings and/or reviews in the OSS sustainability area. By including all of the aspects as synonyms of sustainability, it is possible to locate studies that are sustainability adjacent but don't address it by name directly. This includes important, older, sustainability-related studies from periods where the sustainability terminology was very rarely used.

4.3 Threats to Validity

Errors could occur during SMS execution, especially during study selection. This could compromise the SMS coverage. To address this possible bias, we developed a detailed protocol based on [19], validated by a senior SMS expert, and resolved disagreements through consensus meetings. Similarly, the construction of categories, which requires the interpretation of fragments, may introduce mistakes. This was mitigated by multiple iterations and additional consensus meetings.

Another threat concerns the generalizability of our results, as most analyzed studies stem from academic initiatives, with limited representation from industry or OSS communities. This restricts the scope of our conclusions, which we deliberately confine to the academic context, while validation in industrial or community settings is left for future work. Additionally, our search strategy might not have retrieved all relevant studies. To mitigate this, we refined the search string through pilot studies and targeted well-known SE databases (Scopus, IEEE Xplore, ACM Digital Library). Although synonyms for "sustainability" could not be exhaustively captured, this limitation aligns with the study's objective of clarifying the term's meaning.

Finally, data interpretation poses a potential threat, as many findings rely on the classification of text passages. To reduce subjectivity, we held brainstorming sessions to ensure consistent interpretations of sustainability in OSS, and all extracted data was reviewed by all authors to avoid hasty or biased conclusions.

5 Final Remarks

OSS has become a critical foundation of today's digital infrastructure, yet its sustainability remains an open and multifaceted challenge. In this work, our objective was to clarify how sustainability is understood in the OSS domain, what dimensions and aspects influence it, how these aspects are addressed in research, and the impacts on the design of software systems. Our SMS put together studies from two decades of OSS and sustainability literature, revealing that while social and technical are well-explored, economic and especially environmental dimensions remain significantly underrepresented. The main contribution of this mapping is to provide an overview of sustainability aspects and their interrelations, helping with decisions regarding project direction as well as adoption concerns. By identifying 19 sustainability aspects across five dimensions, our results offer both a conceptual foundation and a practical reference for researchers, maintainers, and adopters who are interested in OSS projects being more resilient in the long term. This mapping raises awareness about the importance of introducing systemic vision to sustain the software ecosystem upon which much of society now depends.

Acknowledgments

This study was supported by CNPq (313245/2021-5), FAPESP (2023/00488-5, 2025/06946-0), and CAPES (001).

References

[1] Adam Alami. 2020. The sustainability of quality in free and open source software. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings (ICSE '20)*. ACM, 222–225. doi:10.1145/3377812.3381402

[2] Christoph Becker, Ruzanna Chitchyan, Leticia Duboc, Steve Easterbrook, Birgit Penzenstadler, Norbert Seyff, and Colin C. Venters. 2015. Sustainability Design and Software: The Karlskrona Manifesto. In *37th International Conference on Software Engineering (ICSE)* (Florence, Italy), Vol. 2. 467–476.

[3] Kristina Rakneberg Berntsen, Morten Rismo Olsen, Narayan Limbu, An Thien Tran, and Ricardo Colomo-Palacios. 2016. *Sustainability in Software Engineering - A Systematic Mapping*. Springer International Publishing, 23–32.

[4] Indushobha Chengalur-Smith, Anna Sidorova, and Sherae Daniel. 2010. Sustainability of Free/Libre Open Source Projects: A Longitudinal Study. *Journal of the Association for Information Systems* 11, 11 (Nov. 2010), 657–683. doi:10.17705/1jais.00244

[5] Daniela S. Cruzes and Tore Dyba. 2011. Recommended Steps for Thematic Synthesis in Software Engineering. In *International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Banff, AB, Canada, 275–284.

[6] Nadia Eghbal. 2016. *Roads and Bridges: The Unseen Labor Behind Our Digital Infrastructure*. Technical Report. Ford Foundation. https://www.fordfoundation.org/work/learning/research-reports/roads-and-bridges-the-unseen-labor-behind-our-digital-infrastructure/

[7] Shane Greenstein and Frank Nagle. 2014. Digital dark matter and the economic contribution of Apache. *Research Policy* 43, 4 (2014), 623–631. doi:10.1016/j.respol.2014.01.003

[8] Hideaki Hata, Taiki Todo, Saya Onoue, and Kenichi Matsumoto. 2015. Characteristics of Sustainable OSS Projects: A Theoretical and Empirical Study. In *2015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering*. IEEE, 15–21. doi:10.1109/chase.2015.9

[9] L. Hilty, W. Lohmann, and E.M. Huang. 2011. Sustainability and ICT - An overview of the field. 13–28 pages.

[10] Heiko Koziolek. 2008. *Goal, Question, Metric*. Springer Berlin Heidelberg, 39–42.

[11] Patricia Lago, Sedef Akinli Koçak, Ivica Crnkovic, and Birgit Penzenstadler. 2015. Framing Sustainability as a Property of Software Quality. *ACM communications* 58, 10 (2015), 70–78.

[12] Zhifang Liao, Libing Deng, Xiaoping Fan, Yan Zhang, Hui Liu, Xiaofei Qi, and Yun Zhou. 2018. Empirical Research on the Evaluation Model and Method of Sustainability of the Open Source Ecosystem. *Symmetry* 10, 12 (Dec. 2018), 747. doi:10.3390/sym10120747

[13] Johan Linåker, Efi Papatheocharous, and Thomas Olsson. 2022. How to characterize the health of an Open Source Software project? A snowball literature review of an emerging practice. In *Proceedings of the 18th International Symposium on Open Collaboration* (Madrid, Spain) *(OpenSym '22)*. Association for Computing Machinery, Article 11, 12 pages. doi:10.1145/3555051.3555067

[14] Manlu Liu, Sean Hansen, and Qiang Tu. 2020. Keeping the family together: Sustainability and modularity in community source development. *Information and Organization* 30, 1 (March 2020), 100274. doi:10.1016/j.infoandorg.2019.100274

[15] Brunna C. Mourão, Leila Karita, and Ivan do Carmo Machado. 2018. Green and Sustainable Software Engineering - a Systematic Mapping Study. In *17th Brazilian Symposium on Software Quality (SBQS)*. Curitiba, Brazil, 121–130.

[16] Kumiyo Nakakoji, Yasuhiro Yamamoto, Yoshiyuki Nishinaka, Kouichi Kishida, and Yunwen Ye. 2002. Evolution patterns of open-source software systems and communities. In *Proceedings of the International Workshop on Principles of Software Evolution (IWPSE02)*. ACM, 76–85. doi:10.1145/512035.512055

[17] Sumaira Nazir, Nargis Fatima, Suriayati Chuprat, Haslina Sarkan, Nurulhuda F, and Nilam N.A. Sjarif. 2020. Sustainable Software Engineering: A Perspective of Individual Sustainability. *International Journal on Advanced Science, Engineering and Information Technology* 10, 2 (2020), 676.

[18] B. Penzenstadler, V. Bauer, C. Calero, and X. Franch. 2012. Sustainability in software engineering: A systematic literature review. In *16th International Conference on Evaluation Assessment in Software Engineering (EASE)* (Ciudad Real, Spain). 32–41.

[19] Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz. 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology* 64 (2015), 1–18.

[20] Maryam Razavian, Giuseppe Procaccianti, and Damian Andrew Tamburri. 2014. Four-Dimensional Sustainable E-Services. In *Informatics for Environmental Protection, Sustainable Development and Risk Management* (Oldenburg, Germany).

[21] Carlos Santos, George Kuk, Fabio Kon, and John Pearson. 2013. The attraction of contributors in free and open source software projects. *The Journal of Strategic Information Systems* 22, 1 (March 2013), 26–45. doi:10.1016/j.jsis.2012.07.004

[22] Charles M Schweik and Robert C English. 2012. *Internet success: a study of open-source software commons*. MIT Press.

[23] Frank van der Linden, Bjorn Lundell, and Pentti Marttiin. 2009. Commodification of Industrial Software: A Case for Open Source. *IEEE Softw.* 26, 4 (jul 2009). doi:10.1109/MS.2009.88

[24] Colin C. Venters, Lydia Lau, Michael K. Griffiths, Violeta Holmes, Rupert R. Ward, Caroline Jay, Charlie E. Dibsdale, and Jie Xu. 2014. The Blind Men and the Elephant: Towards an Empirical Evaluation Framework for Software Sustainability. *Journal of Open Research Software* 2, 1 (2014).

[25] N. Voulvoulis, T. Giakoumis, C. Hunt, V. Kioupi, N. Petrou, I. Souliotis, C. Vaghela, and WIH. binti Wan Rosely. 2022. Systems thinking as a paradigm shift for sustainability transformation. *Global Environmental Change* 75 (2022), 102544.

[26] Johannes Wachs, Mariusz Nitecki, William Schueller, and Axel Polleres. 2022. The Geography of Open Source Software: Evidence from GitHub. *Technological Forecasting and Social Change* 176 (March 2022), 121478.

[27] Likang Yin, Mahasweta Chakraborti, Yibo Yan, Charles Schweik, Seth Frey, and Vladimir Filkov. 2022. Open Source Software Sustainability: Combining Institutional Analysis and Socio-Technical Networks. *Proceedings of the ACM on Human-Computer Interaction* 6, CSCW2 (Nov. 2022), 1–23. doi:10.1145/3555129