Barriers to Open-Source Software Adoption: Review and Synthesis

Dmitrij Petrov *dmitrij.petrov@email.cz*
Friedrich-Alexander-Universität Erlangen-Nürnberg
Nuremberg, Germany

Nikolaus Obwegeser *nikolaus@mgmt.au.dk*
Aarhus University
Aarhus, Denmark

Abstract

Free/Libre and Open Source Software (FLOSS) has had a profound impact on the field of Information Technology. While the adoption rates of FLOSS have been growing, extant research reports various significant barriers that inhibit widespread adoption. To develop a comprehensive overview of FLOSS adoption barriers, this study reviewed and synthesized 44 relevant articles published between 2003 and 2016. Based on the organizing logic of the Technology, Organization, Environment and Individual (TOEI) framework, we categorized the challenges and identified major adoption barriers in each dimension. Technology barriers include vendor lock-in, lack of maturity, and lack of external support. At the organizational level, companies often lack financial and human resources, as well as adequate managerial support. Environmental aspects include culture, policies, and legal frameworks. Individual factors, such as employees' resistance to change and risk-averse leadership may also negatively influence the adoption process.
Keywords: Open Source Software, Adoption, Barriers, TOEI.

1. Introduction

Many organizations recognize that free/libre and open-source software (FOSS/FLOSS) can be a "very viable alternative" to proprietary software [22, 53, 75]. A recent Gartner survey shows that the share of proprietary source code in firms' software portfolios has steadily decreased over time, and many enterprises almost entirely rely on open technologies [27, 77, 99]. Some FLOSS products, such as the Linux operating system, have become a de facto standard and sparked the creation of new complementary products and services – in areas such as cloud and big data. The growth of OSS has also been facilitated by the growing ecosystem of infrastructure providers and corporate supporters, such as GitHub.com and Red Hat Inc. [53, 54]. Besides the growing recognition in private industries, the public sector has also begun engaging with open-source principles and communities, e.g., by trying to accommodate it in governmental procurement processes [15, 16, 23, 94].

Research has addressed many questions regarding FLOSS in organizations, including the adoption process [22, 35, 52, 60, 72]. As the use, quality, and availability of open-source have risen dramatically [38, 52], there is a stream of literature which analyses adoption challenges in various contexts and from different angles. With this study, we aim to review existing literature and provide a comprehensive overview of reported challenges that inhibit adoption of FLOSS in companies. To consolidate existing research that explores these inhibiting factors, our study uses the Technology, Organization, Environment and Individual (TOEI) framework as an organizing frame to provide a structured overview and synthesis of extant research [18].

The remainder of this article is structured as follows. In Section 2 we present related research. Section 3 describes methods used for this study. Then, in section 4 we report our approach to synthesis using TOEI framework and findings of our research organized along the

four dimensions. In section 5, we discuss our results before concluding our work with a summary.

2. Background and Related Research

FLOSS refers to software products offered under a specific open source license that "grants individuals (…) and organizations extensive rights to use, modify, and redistribute the binary and source-code of the original and modified/derived works, without requiring license royalty fees" [24, 52]. As the main distinguishing factor to proprietary software, open source software's defining criterion is that its source code is publicly available [14]. Nevertheless, several different FLOSS licenses are in common use, and the choice of a specific FLOSS license "has an impact on the success or failure" of a product and it "can strongly influence the return on software investment" [14, 21, 80, 82].

Contrary to FLOSS, for proprietary software the source code is kept intentionally secret so that users cannot change and/or redistribute computer programs due to restrictions in the end-user license agreement (EULA) [98]. Such agreements with end-users serve the goal to place certain boundaries and conditions on the use, depending on the intentions of the vendor firm. Table 1 further demonstrates the main differences between FLOSS and proprietary software, alongside core references.

Table 1. Main distinguishing factors between FLOSS and proprietary software.

Factors	Free/Libre Open Source Software	Non-free software	References
Primary actors in the ecosystem	Vendors, End-users/Customers, Independent developers/contributors, Software Foundations	Vendors, End-users/Customers	[55]
Development style & coordination	"Bazaar" (typically no formal methodology) Developed by a large, geographically dispersed community of experts. Developers are self-organized. Decisions are based on transparent, public discussions, accessible to anyone.	"Cathedral" (well-defined methodology with hierarchical, formal, top-down processes). Decisions are made internally. Developers are employees.	[1, 17, 73, 74, 84]
Contributing to software projects	Having source code in the open, contributions are welcomed from everybody and they are judged based on their quality & merit.	Typically, no outside parties can contribute – the vendor develops software alone or with selected contributors.	[17, 43, 44]
Financial costs for the end-user/customer	Free to download with unlimited trial period. May come with some restrictions due to specific licenses.	Users pay a price/fee to offset costs of research and development. The price is set by the vendor. Different pricing models exist.	[10, 71, 83, 85]
Number and dedication of developers, including incentives and rewards	Depending on project's size and importance, large projects have very motivated contributors. Lack of financial incentives. Immediate rewards: recognition, respect, tournament/competition. Delayed rewards: experience, skills for future employment.	Developer size limited by vendor's resources, but developers are contracted with financial incentives/employment.	[9, 25, 28, 46, 79, 100]
Business model and commercialization	Hybrid models exist where vendors can offer commercial add-ons that extend a FLOSS product. Services offered by different firms in the product ecosystem (e.g., implementation, support, training).	Often one-time license fees e.g. for perpetual use or per user. Recently increased use of subscription-based models. Vendors themselves often offer additional services.	[7, 8, 21, 34, 36, 39, 47, 55, 69, 95, 97, 101]
Software quality and testing	Best effort of the whole community – no financial costs or incentive.	Closed test suites and quality assurance processes done by paid professionals.	[1, 59, 71]

While a large number of studies deals with specific aspects of FLOSS adoption, only a few studies follow a systematic review and analysis process to build on the existing body-of-knowledge. A comprehensive overview and structured synthesis of FLOSS adoption challenges is missing, inhibiting future research to systematically advance the field. The main contribution of this study is therefore providing a systematic review of extant research, to serve as a basis for a more cumulative research tradition and future research designs.

In contrast to extant research, our work differs in three key aspects. Firstly, our unit of analysis are all organizations in a global context – instead of applying limitations frequently used in other studies such as specific countries, industries (e.g., software-intensive firms), or firm types (e.g., SMEs) [35, 52]. Secondly, we are not concentrating our effort on barriers of adopting new development practices within an organization (Inner Source), creating a relationship with communities or publishing internally developed software as FLOSS [89]. Rather, we investigate the adoption and integration of FLOSS as for use as a firm's applications or part of the software development processes, e.g., in the form of FLOSS libraries. Thirdly, we narrow ourselves to issues relevant for the primary adoption of FLOSS [38], concerned with firm's initial adoption decision on organizational level ("managers go ahead"). Therefore, our objective in this study is to investigate the challenges that managers and other decision makers perceive in their initial reasoning about FLOSS adoption [20, 22, 26].

3. Methodology

To conduct our review, we closely followed the guidelines for conducting systematic, concept-centric literature reviews in information systems [66, 96]. As a first step, we began with an exploration of the available literature for our objectives by searching Scopus and flosshub.org databases in order to gather primary and secondary sources such as books, journal articles and conference proceedings with relation to FLOSS adoption and barriers. The following key-words were applied for our initial search: "'OSS', 'open (-) source software', 'free software', 'F(L)OSS', 'adoption', 'inhibit(tors)', 'barrier(s)', 'factor(s)', 'challenge(s)', 'integration', 'organization(s)' and 'company(ies)'. Additionally, we complemented our search with queries in Google Scholar for working papers and other reports. To filter the large number of results (~4000 articles), we skimmed the titles and abstracts for relevance, guided by the main criteria: does the study mainly investigate the role of one or multiple factors related to FLOSS adoption?

Through this rigorous search and filter process we arrived at an initial set of 55 documents for further consideration. Each document was fully skimmed for its content to exclude those that are beyond the scope of our research question. After this practical screen, the remaining articles from the different sources were combined into a preliminary sample of 33 articles. Moreover, we extended our sample by applying the snowball technique where we scanned the collected documents for additional citations the authors have referenced too [66, 96]. Each relevant study was then added to our preliminary sample in order to further increase our literature findings. Hence, the sample was increased by 30 additional studies through such snowballing of references [64]. As a final step, to ensure both quality and relevance of articles in our sample, we engaged in an in-depth reading and discussion of all articles. As a result of this final appraisal, we reduced our sample from 63 to 44 articles for further consideration.

Our review qualitatively examined data from studies using a general inductive approach which allowed us to find "frequent, dominant, or significant" themes with relation to FLOSS adoption barriers that were reported across different studies [91]. To extract meaning from a complex set of literature works "through the development of summary themes", we categorized our inductive codes based on the technology, organization and environmental (TOE) framework [18, 66, 91]. The TOE framework has demonstrated its usefulness in a variety of academic studies aiming to understand how organizations adopt technological innovations and thus suits well our research objectives [18, 30, 39]. However, due to its lack of focus on aspects of the individual adopter, a number of researchers proposed to incorporate a fourth dimension – hence the additional element 'individual', together forming the TOEI framework [31–33, 42, 60, 70, 92].

During close readings of all articles, we summarized them to extract adoption barriers, allowing us to establish labels for key inhibiting challenges [91]. Given our primary goal to develop these barriers for subsequent use in the TOEI framework, we compiled a concept matrix in the form of an inhibitor matrix [96]. By conducting open coding during which text segments relating to themes from "each article [was] assigned to one of the broad categories derived from the literature review", a spreadsheet with text snippets about the reported barriers was established [54, 91].

To perform the inductive analysis, we identified challenges that authors have mentioned explicitly as well as implicitly while iteratively analysing each document and its content. Parts of the text or phrases were then labelled so that links among them could be gradually established, i.e. a non-hierarchical list of codes was created [66]. When we were not able to assign a text snippet to the existing coding scheme, we made a new category as we saw fit [54]. As an initial result, we listed over 50 different barriers at various levels of abstraction, i.e., high-level (e.g. risk factor) and low-level (e.g. documentation issues).

4. TOEI Framework of FLOSS Adoption

As a result of our inductive analysis and the subsequent iterative generation of high-level categories, **Fig. 1** presents the challenges mapped into the four classes of the TOEI. Naturally, many of these challenges are interconnected with each other, further inhibiting companies to even consider FLOSS. In the remainder of this section we explore both top-level and individual challenges in a more detailed fashion, and explain the potential interconnections among the factors [66, 91].

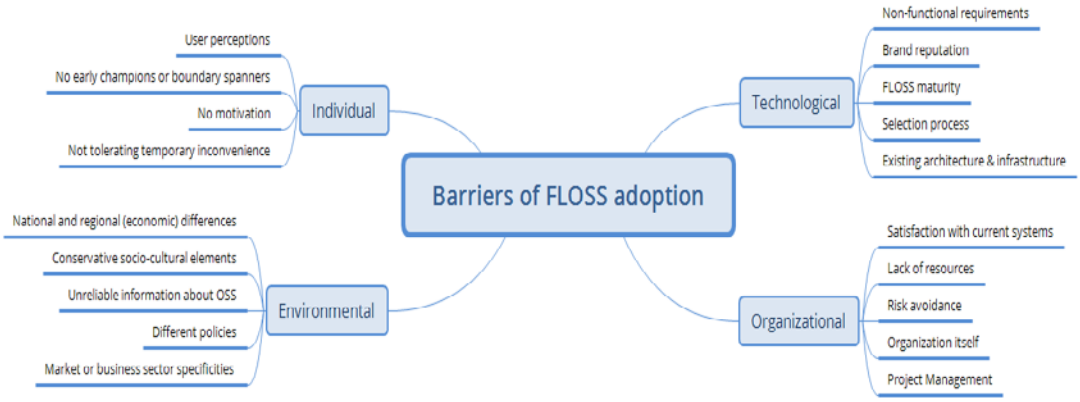

Fig. 1. FLOSS adoption barriers categorized into 4 TOEI dimensions and 19 secondary factors.

4.1. Technological Factors

Across studies, we observed five high-level categories of technological challenges that make FLOSS adoption difficult. We differentiated between functional and non-functional barriers, project reputation, open-source selection processes, and existing IT systems and infrastructure as factors commonly mentioned in the literature. The single most frequently mentioned non-functional inhibitor is technological vendor lock-in. The frequent non-interoperability between two types of applications leads to early rejection of open-source solutions as they are non-conforming to the existing closed standards [31, 61, 92]. To deal with such situations, [51, 101] suggest the use of different policies, one of which is to require proprietary software vendors to support open standards or apply a principle of “explain [the use of proprietary software] and commit [or plan] to use OSS” [11].

Another pressing issue firms repeatedly mention is the security and privacy of open-source solutions [3, 62, 81, 101]. This is particularly relevant in mission-critical sectors such as in finance, healthcare or for public institutions. This challenge is strongly related with the quality of software – a well-known issue that is often discussed both in practice and academia [58, 71]. Indeed, proprietary programs are oftentimes thought to be of higher quality because of being developed by full-time, dedicated developers [32, 76, 101]. However, prior research established that when computer programs are closed-source it does not increase their quality and security [71].

The factor of reliability and uncertainty of FLOSS has been frequently mentioned as an adoption barrier too [57, 60, 72, 92]. Since developers are usually not being paid for their

contributions to OSS, their voluntary effort may stop at any time and no maintenance or further developments will be provided to adopters of the software. Moreover, a general lack of roadmaps and clarity of future plans (and "questionable longevity") often discourages companies to adopt such technology [13, 60, 76, 88]. Strongly related to these issues is the aspect of a FLOSS project's reputation, which plays an important role in adoption behaviour. The more popular the project brand is, or if the project is being backed by a company or sponsor, the more users and developers it can attract, and thus further increase its trustworthiness and reputation [45, 57, 92].

The maturity of FLOSS, similar to their quality, varies considerably. Studies have shown that such software often lacks certain functionalities and features when compared to proprietary counterparts [41, 52, 60, 63, 87]. Additionally, in order to be recognized and integrated in a company, it needs to fit existing organizational operations and have a tangible added value [32, 60, 76].

Due to reported problems of software availability and difficult discovery [13], seemingly chaotic release variety (e.g., which fork to pick) [88], and ownership questions [60], research mentions selection challenges as an adoption barrier. Selecting the right open-source program for a firm or task can be more complicated than a choosing among proprietary, commercial off-the-shelf alternatives. The selection process needs to be adapted and extended to evaluate "new factors", such as terms of the license and FLOSS compliance with standards [13, 29].

In our inhibitor matrix, the most cited technological barrier that prohibits companies to adopt FLOSS software is the lack of support by the community and third-party companies [13, 60, 93]. Some researchers offer an arguably simple explanation: due to the existing IT architecture and perceived challenges of integrating any new technological innovation in the enterprise context, companies require direct access to expertise that can provide support for implementation and initial teething problems [63, 88]. The widespread lack of commercial open-source vendors therefore further acts as an adoption barrier in itself [32].

Technological barriers have played a critical part in the dissemination of FLOSS into corporate environments. Solving the reported issues of technical compatibility and availability of commercial support would likely have a very positive effect on the number of organizations willing to consider FLOSS for their purposes, both in the public and private sector.

Table 2. List of technological barriers and literature references.

Technological Factors	References
Non-functional requirements	[3, 5, 11, 13, 19, 30–33, 37, 38, 40–42, 45, 51–53, 56, 57, 60, 62, 63, 67, 68, 70, 72, 75, 76, 78, 86, 88–90, 92, 93, 101]
Brand reputation	[30, 32, 42, 45, 57, 68, 87, 89, 92]
FLOSS maturity	[3, 5, 19, 30, 32, 37, 40–42, 50, 52, 53, 57, 60, 62, 63, 67, 75, 76, 86, 87, 93, 101]
Selection process	[3, 5, 12, 13, 31–33, 37, 38, 40–42, 45, 48–50, 52, 53, 57, 60, 62, 63, 67, 68, 70, 72, 75, 76, 86–89, 92, 93, 101]
Existing architecture & infrastructure	[3, 11, 13, 31, 32, 38, 42, 45, 52, 53, 57, 60, 62, 78, 86–89, 101]

4.2. Organizational Factors

Within organizational barriers, we categorized the reported inhibitors into the subgroups of lack of resources, satisfaction with the existing systems, organizational characteristics, project management, and risk avoidance.

The most significant organizational barriers reported are lacking financial and human resources. Due to many companies already owning proprietary products and limited IT budgets, firms need to consider switching costs that include both upcoming hidden as well as sunk expenses (e.g. for licenses) [32, 57, 62, 63, 68]. Even though FLOSS is free of charge to use, organizations may need "to hire programmers to supplement their IT staff" – thus the total costs of ownership have to include items such as implementation and execution support, and employee training [42, 53, 57, 76]. Furthermore, particularly in developing countries,

proprietary products are seen to be more prestigious and of higher quality. Paradoxically, low acquisition costs may therefore have actually discouraged the adoption of FLOSS in some cases [87, 92].

Firms have reported a lack of skilled FLOSS specialists and accessibility of open source vendors as a significant inhibitor for the adoption [49, 57, 68, 87]. Indeed, given that many employees have been used working with proprietary software, they need to be provided with the adequate training, which further complicates the transition and increases time and costs [22, 42, 57, 86].

As part of a general trend of organizations to focus on their core competencies, IT systems and their development and support have often been outsourced. Negotiations with suppliers may prove to be difficult due to the lack of FLOSS support among outsourcing providers [62]. Companies have therefore often decided to direct their limited resources to existing proprietary ISs rather than risking something new [32]. Decision makers prefer to avoid potential failures at all costs as the risk of systems not functioning and providing satisfactory features is considerable [48, 57]. Additionally, with no means to transfer accountability (unlike in many outsourcing relations for proprietary software), the full risk for FLOSS remains within the client organization [5, 31, 60, 62].

Satisfaction with existing systems has also played a role in FLOSS non-adoption. Where there has been no user demand, there were no reasons to implement it either [32, 60, 76]. Indeed, if a company accesses proprietary software through existing purchasing agreements for acceptable costs, there is no need to consider switching to alternatives in the first place [31, 52].

Organization size has been mentioned to be another barrier [48, 53, 70]. If the organization is small, it may lack a fully equipped IT department and skilled employees who could support FLOSS integration. On the contrary, the larger the organization is, the more probable open-source adoption can be successfully accomplished [62]. Further, as many studies show, particularly the lack of managerial support, its awareness of alternatives, and the organizational structure has been identified as a burden for corporate change [12, 30, 31, 51].

The best outcomes of FLOSS adoption projects occur when employees themselves, bottom up, voluntarily adopt open-source software, resulting in motivated and satisfied end-users [22]. This is due to the fact that many FLOSS implementation projects have been perceived as unsuccessful, often due to unrealistic implementation times, lack of detailed planning, training, pilot trials, and insufficient project governance [12, 32, 60]. Indeed, for a FLOSS adoption to succeed, managers have to be fully on board and aware of the time required, costs involved, and risks it brings along [52, 57].

Table 3. List of organizational barriers and literature references.

Organizational Factors	References
Satisfaction with current systems	[3, 30–32, 42, 45, 52, 57, 60, 62, 67, 75, 76, 86, 101]
Lack of resources	[3, 5, 11, 12, 19, 30–32, 37, 40–42, 45, 48, 49, 51, 52, 57, 60, 62, 63, 68, 70, 72, 75, 76, 78, 86, 87, 89, 90, 92, 93, 101]
Risk avoidance	[2, 3, 13, 30–33, 37, 38, 40, 41, 48, 49, 57, 60, 62, 68, 70, 72, 75, 76, 86–90]
Organization itself	[12, 30–32, 42, 45, 48, 50, 51, 53, 60, 62, 67, 70, 72, 76, 89, 90]
Project Management	[12, 31, 32, 38, 41, 52, 57, 60, 62, 76, 86, 88, 89, 92]

4.3. Environmental Factors

Environmental factors are concerned with the setting and context in which organizations operate. Within this group, we find following categories: economic and country differences (e.g. in terms of infrastructure), policies and laws, structure and conditions of the industry, cultural issues and nature of the public discourse.

One of the most important challenges have been different policies, which are crucial in the FLOSS acceptance by employers. At governmental level, studies have reported that procurement models are usually discriminatory, non-transparent and not flexible enough, thus severely limiting FLOSS market penetration [11, 62, 65, 90]. In addition, political and public

pressure and lack of the governmental support in terms of planning, future directions and visions have significantly discouraged many firms – especially relevant in sectors heavily relying on public funds, such as education or healthcare [51, 53, 67, 68]. Politics and hidden power structures are "a critical factor" with relation to "how OSS would be used in the future, even before getting to the technology [and other] portion[s] of [open-source] adoption" [62].

Some studies also present the issue of companies required to have IT systems that are in compliance with certain business regulations (e.g., privacy, security, standards, etc.), which usually cannot be achieved without corporate backing [13, 53, 62].

Socio-cultural elements heavily influence the FLOSS adoption patterns. For example, [3] have found in Mali that people of higher age (elders) carry larger decision power than those of younger age. If the elders have not been entirely convinced, other employees would be reluctant to follow instructions. This is covered in Hofstede's concept of cultural power distance, where the higher the distance between different employees, the less likely novelties such as FLOSS are adopted [70]. From this view, enterprises "are reluctant to changes (…) because new things may threaten the existing power structure" [70].

A firm's market condition and its structure play a relevant factor in the decision about free software migration too [11, 32]. [90] states that "type of industry matters" because those with a strong relation to IT (e.g., ICT companies) are naturally more inclined to adopt new technologies. This has also been confirmed by [32, 60] who added that competitors' behaviour (e.g., "other nearby firms had rejected" it) significantly influenced FLOSS adoption decisions in practice.

With relation to organization's internal culture, research has shown that conservative firms do not support free software [31, 68]. Moreover, firm's local needs and requirements – e.g., language localization of open-source products and adaptability to the market – have to be appropriately addressed through engaging with local or regional communities and commercial vendors [3, 6, 101]. Therefore the openness of free software and ability to modify its source code encourages a further development, customization and enables potential cooperation between different parties in the ecosystem [11].

Another considerable barrier to adoption is the lack of a sales organization that promotes the FLOSS project through educating potential future adopters and acts as a central source of knowledge for interested parties [22, 31, 60, 63]. Unclear and incomplete information about open-source projects and the lack of commercial vendors often stop adoption considerations in early stages [32, 60, 68, 93]. Such a lack of marketing efforts and information dissemination goes hand in hand with poor management of the public discourse [53].

To overcome the plethora of environmental barriers, clear policies and legal frameworks have to be created and put into action, while at the same time information about open-source software needs to improve qualitatively and quantitatively.

Table 4. List of environmental barriers and literature references.

Environmental Factors	References
National and regional (economic) differences	[57, 70]
Conservative socio-cultural elements	[3, 6, 31, 62, 68, 70, 93, 101]
Unreliable information about OSS	[32, 41, 53, 60, 68, 93]
Different policies	[11, 13, 19, 31, 32, 38, 41, 42, 51, 53, 62, 67, 68, 70, 87, 89, 90, 93, 101]
Market or business sector specifics	[11, 32, 41, 60, 62, 90]

4.4. Individual Factors

Ultimately, the success of FLOSS implementations, like with other technological innovations, depends on their perceived usefulness and ease of use by the end-user [5, 33, 92]. Yet, a proprietary mindset among users may lead to substantial resistance towards FLOSS transitions, especially when the new technology is seen as a cheap alternative and thus decreases the self-perceived value of employees [22, 31, 42, 62].

Companies aiming to introduce FLOSS often lack champions or sponsors "who connect their organization with external knowledge and can bring the organization in contact with new innovations" [4, 30, 31, 45, 51, 60]. Having such a boundary spanner is particularly relevant because many individuals lack personal awareness and knowledge about alternatives to proprietary software [42]. Therefore, the main task of these champions is to motivate users to try free software which has been shown to encourage broader adoption of open-source [50].

A less frequently mentioned challenge is to overcome the (temporary) inconvenience for employees that a migration to FLOSS would ultimately bring [31]. Users generally want to use their programs in a productive matter, at any time and without disruptions. A transition to FLOSS, just like other technological changes, holds the risk of destabilizing the current work system before the benefits of the new technology can be realized.

Individual factors have to be considered in close relationship with the three other dimensions of the TOEI framework. While we can separate the challenges theoretically in order to allow for a more structured analysis and discussion, many of the challenges that inhibit open-source adoption evidently lead to the users and their behaviour, as it is the users that ultimately hold the power over successful adoption. Apart from hidden financial expenses related to implementation, training and maintenance [11], firms often pay an additional price related to employee morale, changes in the technological and organizational landscape and therefore face user resistance from those affected [12].

Table 5. List of individual barriers and literature references.

Individual Factors	References
User perceptions	[5, 30–33, 37, 40, 42, 45, 57, 62, 70, 89, 92, 93]
No early champions or boundary spanners	[30–32, 45, 51, 53, 60, 62]
No motivation	[50]
Not tolerating temporary inconvenience	[31, 62]

5. Discussion and Conclusion

Our study presents a consolidated view of FLOSS adoption research based on the analysis and categorization of 44 articles into four conceptual areas, each consisting of a number of associated factors that act as barriers to the widespread adoption of FLOSS in organizations.

In line with prior research, our results also support the notion "that OSS had been widely adopted and diffused, but only for specific types of applications or in organizations with particular profiles" [54]. In some industries, such as healthcare, education or public administration, the diffusion has been limited due to the regulatory environment or lack of interoperability with existing software [11, 41, 51]. Moreover, our research further shows that adoption barriers and their importance can differ economically and geographically, and that the different rates of FLOSS adoption by countries may be explained through cultural aspects of power-distance. In countries with a high-power distance, employees are more inclined to maintain the status quo [5, 6, 70].

All in all, due to its growing popularity, the adoption rate of FLOSS in both private and public organizations is steadily increasing. With this study, we provide a comprehensive overview of FLOSS adoption barriers – factors that may negatively influence FLOSS initiatives – as well as insights from extant research on how to overcome these challenges. Based on the results of a systematic literature review, we categorized the reported barriers into the 4 dimensions of the TOEI framework, and further described 19 associated, inhibiting challenges. Our overview sheds new light on FLOSS adoption barriers and may help practitioners to make more informed decisions. For researchers, this study contributes in the form of a knowledge baseline, which can be used to better identify research gaps in our current knowledge about FLOSS adoption, and thus allow for a more structured advance of the field and more informed research designs in future studies.

References

1. Aberdour, M.: Achieving Quality in Open Source Software. IEEE Softw. 24 (1), 58–64 (2007)
2. Årdal, C., Alstadsæter, A., Røttingen, J.-A.: Common characteristics of open source software development and applicability for drug discovery: a systematic review. Heal. Res. Policy Syst. 9 36 (2011)
3. Bagayoko, C.-O., Dufour, J.-C., Chaacho, S., Bouhaddou, O., Fieschi, M.: Open source challenges for hospital information system (HIS) in developing countries: a pilot project in Mali. BMC Med. Inform. Decis. Mak. 10 (1), 22 (2010)
4. Bauer, S., Obwegeser, N., Avdagic, Z.: Corporate Accelerators: Transferring Technology Innovation to Incumbent Companies. In: Proceedings of the 10th Mediterranean Conference on Information Systems. AIS, Paphos,Cyprus (2016)
5. Bhadauria, V.S., Mahapatra, R., Manzar, R.: Factors Influencing Adoption of Open Source Software – An Exploratory Study. AMCIS 2009 Proc. 114 (2009)
6. Bhatt, P., Ahmad, A.J., Roomi, M.A.: Social innovation with open source software: User engagement and development challenges in India. Technovation. 52–53 28–39 (2016)
7. Bonaccorsi, A., Lamastra, C.R., Giannangeli, S.: Adaptive Entry Strategies under Dominant Standards - Hybrid Business Models in the Open Source Software Industry. (2004)
8. Bonaccorsi, A., Piscitello, L., Merito, M., Rossi, C.: How is it possible to profit from innovation in the absence of any appropriability? In: Damiani, E., Fitzgerald, B., Scacchi, W., Scotto, M., and Succi, G. (eds.) Open Source Systems: IFIP Working Group 2.13 Foundation on Open Source Software, June 8--10, 2006, Como, Italy. pp. 333–334. Springer US, Boston, MA (2006)
9. Bonaccorsi, A., Rossi, C.: Comparing motivations of individual programmers and firms to take part in the open source movement: From community to business. Knowledge, Technol. Policy. 18 (4), 40–64 (2006)
10. Bons, E.: Total Cost of Ownership of Open Source Software. Dep. Educ. Child. Serv. Adelaide, Aust. 25 1–71 (2011)
11. Bouras, C., Filopoulos, A., Kokkinos, V., Michalopoulos, Sotiris; Papadopoulos, D., Tseliou, G.: Policy recommendations for public administrators on free and open source software usage. Telemat. Informatics. 31 (2), 237–252 (2014)
12. Brink, D., Roos, L., Weller, J., Van Belle, J.-P.: Critical Success Factors for Migrating to OSS-on-the-Desktop: Common Themes across Three South African Case Studies. In: Damiani, E., Fitzgerald, B., Scacchi, W., Scotto, M., and Succi, G. (eds.) Open Source Systems: IFIP Working Group 2.13 Foundation on Open Source Software, June 8--10, 2006, Como, Italy. pp. 287–293. Springer US, Boston, MA (2006)
13. Chen, W., Li, J., Ma, J., Conradi, R., Ji, J., Liu, C.: An empirical study on software development with open source components in the Chinese software industry. Softw. Process Improv. Pract. 13 (1), 89–100 (2008)
14. Colazo, J., Fang, Y.: Impact of license choice on Open Source Software development activity. J. Am. Soc. Inf. Sci. Technol. 60 (5), (2009)
15. Comino, S., Manenti, F.M.: Government policies supporting open source software for the mass market. Rev. Ind. Organ. 26 (2), 217–240 (2005)
16. Comino, S., Manenti, F.M.: Open Source vs Closed Source Software: Public Policies in the Software Market. (2003)
17. Crowston, K., Wei, K., Howison, J., Wiggins, A.: Free/Libre open-source software development: What we know and what we do not know. ACM Comput. Surv. 44 (2), 7:1--7:35 (2008)
18. DePietro, R., Wiarda, E., Fleischer, M.: The context for change: Organization, technology and environment. In: Tornatzky, L. G. and Fleischer, M. (ed.) The processes of technological innovation. pp. 151–175. Lexington Books, Lexington, MA (1990)
19. Dobusch, L.: Migration Discourse Structures: Escaping Microsoft's Desktop Path. In: Russo, B., Damiani, E., Hissam, S., Lundell, B., and Succi, G. (eds.) Open Source Development, Communities and Quality: IFIP 20th World Computer Congress, Working Group 2.3 on Open Source Software, September 7-10, 2008, Milano, Italy. pp. 223–235. Springer US, Boston, MA (2008)
20. Donaldson, D.R., Yakel, E.: Secondary adoption of technology standards: The case of PREMIS. Arch. Sci. 13 (1), 55–83 (2013)
21. Ferrante, D.: Software Licensing Models: What's Out There? IT Prof. 8 (6), 24–29 (2006)
22. Fitzgerald, B.: Open source software adoption: Anatomy of success and failure. Int. J. Open Source Softw. Process. 1 (1), 1–23 (2009)
23. Fitzgerald, B., Kenny, T.: Open source software the trenches: Lessons from a large-scale OSS

implementation. ICIS 2003 Proc. 2003 (27), 11 (2003)
24. Franco-Bedoya, O., Ameller, D., Costal, D., Franch, X.: Open source software ecosystems: A Systematic mapping. Inf. Softw. Technol. 91 160–185 (2017)
25. Fuggetta, A.: Open source software—an evaluation. J. Syst. Softw. 66 (1), 77–90 (2003)
26. Gallivan, M.J.: Organizational adoption and assimilation of complex technological innovations: development and application of a new framework. ACM Sigmis Database. 32 (3), 51–85 (2001)
27. Gartner: Survey Analysis: Overview of Preferences and Practices in the Adoption and Usage of Open-Source Software. (2011)
28. Ghosh, R.A., Glott, R., Krieger, B., Robles, G.: Free/Libre and Open Source Software: Survey and Study. , Berlin, Germany; Maastricht, The Netherlands (2002)
29. Di Giacomo, P.: COTS and Open Source Software Components: Are They Really Different on the Battlefield? In: ICCBSS'05 Proceedings of the 4th international conference on COTS-Based Software Systems. pp. 301–310. Springer-Verlag Berlin (2005)
30. Gichira, C.M., Kahonge, A.M., Miriti, E.K.: Adoption of Open Source Software by Organizations – A Framework for Kenya. Int. J. Comput. Appl. 59 (7), 25–32 (2012)
31. Glynn, E., Fitzgerald, B., Exton, C.: Commercial adoption of open source software: an empirical study. In: 2005 International Symposium on Empirical Software Engineering, 2005. p. 10 pp.-. (2005)
32. Goode, S.: Something for nothing: management rejection of open source software in Australia's top firms. Inf. Manag. 42 (5), 669–681 (2005)
33. Gwebu, K.L., Wang, J.: Seeing eye to eye? An exploratory study of free open source software users' perceptions. J. Syst. Softw. 83 (11), 2287–2296 (2010)
34. Hall, A.: Open-Source Business Models: Making money by giving it away, (2015)
35. Hauge, Ø., Ayala, C., Conradi, R.: Adoption of open source software in software-intensive organizations – A systematic literature review. Inf. Softw. Technol. 52 (11), 1133–1154 (2010)
36. Hecker, F.: Setting up shop: The business of open-source software. IEEE Softw. 16 (1), 45–51 (1999)
37. Holck, J., Larsen, M.H., Pedersen, M.K.: Identifying Business Barriers and Enablers for the Adoption of Open Source Software. In: Vilnius, L. (ed.) Proceedings of the 13th International Conference on Information Systems Development. p. 15. (2004)
38. Holck, J., Larsen, M.H., Pedersen, M.K.: Managerial and Technical Barriers to the Adoption of Open Source Software. Springer Berlin Heidelberg (2005)
39. Johnston, K., Begg, S., Tanner, M.: Exploring the factors influencing the adoption of Open Source Software in Western Cape schools. Int. J. Educ. Dev. using ICT. 9 (2), 64–84 (2013)
40. Jusoh, Y., Chamili, K., Yahaya, J.H., Pa, N.C.: The selection criteria of open source software adoption in Malaysia. Int. J. Adv. Comput. Technol. 4 (21), 278–287 (2012)
41. Karopka, T., Schmuhl, H., Demski, H.: Free/libre open source software in health care: A review. Healthc. Inform. Res. 20 (1), 11–22 (2014)
42. Kisanjara, S., Tossy, T.: Investigating Factors Influencing the Adoption and Use of Free and Open Source Software (FOSS) in Tanzanian Higher Learning Institutions: Towards an Individual-Technology-Organizational-Environmental (ITOE) Framework. Int. J. Res. Bus. Technol. 5 (2), 645–653 (2014)
43. Kogut, B., Metiu, A.: Open-source software development and distributed innovation. Oxford Rev. Econ. policy. 17 (2), 248–264 (2001)
44. von Krogh, G., Spaeth, S., Lakhani, K.R.: Community, joining, and specialization in open source software innovation: a case study. Res. Policy. 32 (7), 1217–1241 (2003)
45. Laila, U., Bukhari, S.F.A.: Open Source Software (OSS) adoption framework for local environment and its comparison. In: Sobh, T. and Elleithy, K. (eds.) Innovations in Computing Sciences and Software Engineering. pp. 13–16. Springer Netherlands (2010)
46. Lakhani, K.R., von Hippel, E.: How open source software works: "free" user-to-user assistance. Res. Policy. 32 (6), 923–943 (2003)
47. Langham, M.: The business of open source, *OSS Watch*, (2012)
48. Li, Y., Tan, C.-H., Teo, H.-H., Siow, A.: A Human Capital Perspective of Organizational Intention to Adopt Open Source Software. In: International Conference on Information Systems 2005. p. 13. ICIS 2005 Proceedings (2005)
49. Li, Y., Tan, C.-H., Yang, X.: It is all about what we have: A discriminant analysis of organizations' decision to adopt open source software. Decis. Support Syst. 56 (1), 56–62 (2013)
50. Li, Y., Tan, C., Xu, H., Teo, H.-H.: Open source software adoption: motivations of adopters and amotivations of non-adopters. ACM SIGMIS Database. 42 (2), 76 (2011)
51. van Loon, A., Toshkov, D.: Adopting open source software in public administration: The

importance of boundary spanners and political commitment. Gov. Inf. Q. 32 (2), 207–215 (2015)
52. Macredie, R.D., Mijinyawa, K.: A theory-grounded framework of Open Source Software adoption in SMEs. Eur. J. Inf. Syst. 20 (2), 237–250 (2011)
53. Marsan, J., Paré, G.: Antecedents of open source software adoption in health care organizations: A qualitative survey of experts in Canada. Int. J. Med. Inform. 82 (8), 731–741 (2013)
54. Marsan, J., Paré, G., Wybo, M.D.: Has open source software been institutionalized in organizations or not? Inf. Softw. Technol. 54 (12), 1308–1316 (2012)
55. Mickos, M.: MySQL CEO on 2 Open Source Business Models. Heavybit Industries (2013)
56. Midha, V., Palvia, P.: Factors affecting the success of Open Source Software. J. Syst. Softw. 85 (4), 895–905 (2012)
57. Miralles, F., Sieber, S., Valor, J.: An Exploratory Framework for Assessing Open Source Software Adoption. Systèmes d'Information Manag. 11 (1), 85–106 (2006)
58. Mishra, B., Prasad, A., Raghunathan, S.: Quality and profits under open source versus closed source. ICIS 2002 Proc. 2002 (32), (2002)
59. Mockus, A., Fielding, R.T., Herbsleb, J.D.: Two case studies of open source software development: Apache and Mozilla. ACM Trans. Softw. Eng. Methodol. 11 (3), 309–346 (2002)
60. Morgan, L., Finnegan, P.: Open Innovation in Secondary Software Firms: An Exploration of Managers' Perceptions of Open Source Software. SIGMIS Database. 41 (1), 76–95 (2010)
61. Moyle, K.: Total cost of ownership and open source software. (2004)
62. Munoz-Cornejo, G., Seaman, C.B., Koru, A.G.: An Empirical investigation into the Adoption of open Source Software in Hospitals. Int. J. Healthc. Inf. Syst. Informatics. 3 (3), 22 (2008)
63. Nagy, D., Yassin, A.M., Bhattacherjee, A.: Organizational adoption of open source software: barriers and remedies. Commun. ACM. 53 (3), 148–151 (2010)
64. Noy, C.: Sampling knowledge: the hermeneutics of snowball sampling in qualitative research. Int. J. Soc. Res. Methodol. 11 (2008), 4 (2008)
65. Obwegeser, N., Müller, S.D.: Innovation and public procurement: Terminology, concepts, and applications. Technovation. 74–75 1–17 (2018)
66. Okoli, C., Schabram, K.: A Guide to Conducting a Systematic Literature Review of Information Systems Research. Work. Pap. Inf. Syst. 10 (26), 1–51 (2010)
67. Ozel, B., Jovanovic, U., Oba, B., van Leeuwen, M.: Perceptions on F/OSS Adoption. In: Feller, J., Fitzgerald, B., Scacchi, W., and Sillitti, A. (eds.) Open Source Development, Adoption and Innovation: IFIP Working Group 2.13 on Open Source Software, June 11--14, 2007, Limerick, Ireland. pp. 319–324. Springer US, Boston, MA (2007)
68. Paré, G., Wybo, M.D., Delannoy, C.: Barriers to open source software adoption in Quebec's health care organizations. J. Med. Syst. 33 (1), 1–7 (2009)
69. Perr, J., Appleyard, M.M., Sullivan, P.: Open for business: Emerging business models in open source software. Int. J. Technol. Manag. 52 (3–4), 432–456 (2010)
70. Qu, W.G., Yang, Z., Wang, Z.: Multi-level framework of open source software adoption. J. Bus. Res. 64 (9), 997–1003 (2011)
71. Raghunathan, S., Prasad, A., Mishra, B.K., Chang, H.: Open source versus closed source: software quality in monopoly and competitive markets. IEEE Trans. Syst. Man, Cybern. A Syst. Humans. 35 (6), 903–918 (2005)
72. Ramanathan, L., Krishnan, S.: An empirical investigation into the adoption of open source software in Information Technology outsourcing organizations. J. Syst. Inf. Technol. 17 (2), 167–192 (2015)
73. Raymond, E.S.: The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. O'Reilly Media (1999)
74. Riehle, D., Ellenberger, J., Menahem, T., Mikhailovski, B., Natchetoi, Y., Naveh, B., Odenwald, T.: Open collaboration within corporations using software forges. IEEE Softw. 26 (2), 52–58 (2009)
75. van Rooij, S.W.: Adopting Open-Source Software Applications in U.S. Higher Education: A Cross-Disciplinary Review of the Literature. Rev. Educ. Res. 79 (2), 682–701 (2009)
76. van Rooij, S.W.: Perceptions of Open Source Versus Commercial Software. J. Res. Technol. Educ. 39 (4), 433–453 (2007)
77. Ruscio, D. Di, Pelliccione, P.: Simulating upgrades of complex systems: The case of Free and Open Source Software. Inf. Softw. Technol. 56 (4), 438–462 (2014)
78. Russo, B., Zuliani, P., Succi, G.: Toward an empirical assessment of the benefits of open source software. In: Taking Stock of the Bazaar. pp. 117–120. Proceedings of 3rd Workshop on Open Source Software Engineering, International Conference on Software Engineering . Portland, Oregon (2003)

79. Sacks, M.: Competition Between Open Source and Proprietary Software: Strategies for Survival. J. Manag. Inf. Syst. 32 (3), 268–295 (2015)
80. Santos, C.D. dos: Changes in free and open source software licenses: managerial interventions and variations on project attractiveness. J. Internet Serv. Appl. 8 (1), 11 (2017)
81. Schryen, G., Kadura, R.: Open source vs. closed source software: towards measuring security. In: Proceedings of the 2009 ACM symposium on. pp. 2016–2023. ACM (2009)
82. Sen, R., Subramaniam, C., Nelson, M.L.: Open source software licenses: Strong-copyleft, non-copyleft, or somewhere in between? Decis. Support Syst. 52 (1), 199–206 (2011)
83. Shaikh, M., Cornford, T.: Total cost of ownership of open source software: a report for the UK Cabinet Office supported by OpenForum Europe. (2011)
84. Shaikh, M., Henfridsson, O.: Governing open source software through coordination processes. Inf. Organ. 27 (2), 116–135 (2017)
85. Singh, A., Bansal, R.K., Jha, N.: Open source software vs proprietary software. Int. J. Comput. Appl. 114 (18), (2015)
86. Sohn, S.Y., Mok, M.S.: A strategic analysis for successful open source software utilization based on a structural equation model. J. Syst. Softw. 81 (6), 1014–1024 (2008)
87. Spinellis, D., Giannikas, V.: Organizational adoption of open source software. J. Syst. Softw. 85 (3), 666–682 (2012)
88. Stol, K.-J., Ali Babar, M.: Challenges in Using Open Source Software in Product Development: A Review of the Literature. In: Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development. pp. 17–22. ACM, New York, NY, USA (2010)
89. Stol, K.-J., Babar, M.A., Avgeriou, P., Fitzgerald, B.: A comparative study of challenges in integrating Open Source Software and Inner Source Software. Inf. Softw. Technol. 53 (12), 1319–1336 (2011)
90. Thakur, D.: A limited revolution — The distributional consequences of Open Source Software in North America. Technol. Forecast. Soc. Change. 79 (2), 244–251 (2012)
91. Thomas, D.R.: A General Inductive Approach for Analyzing Qualitative Evaluation Data. Am. J. Eval. 27 (2), 237–246 (2006)
92. Tome, L., Johnston, K.A., Meadows, A., Nyemba-Mudenda, M.: Barriers to Open Source ERP Adoption in South Africa. African J. Inf. Syst. 6 (2), 22 (2014)
93. Uwizeyemungu, S., Poba-Nzaou, P.: Barriers to Mission-Critical Open Source Software Adoption by Organizations: A Provider Perspective. In: Proceedings of the Nineteenth Americas Conference on Information Systems. p. 10. AMCIS 2013 Proceedings, Chicago, Illinois (2013)
94. Waring, T., Maddocks, P.: Open Source Software implementation in the UK public sector: Evidence from the field and implications for the future. Int. J. Inf. Manage. 25 (5), 411–428 (2005)
95. Watson, R.T., Wynn, D., Boudreau, M.-C.: JBOSS: The Evolution of Professional Open Source Software. MIS Q. Exec. 4 (3), 329–341 (2005)
96. Webster, J., Watson, R.T.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. MIS Q. 26 (2), xiii–xxiii (2002)
97. West, J.: How open is open enough?: Melding proprietary and open source platform strategies. Res. Policy. 32 (7), 1259–1285 (2003)
98. Williams, S., Stallman, R.M.: Free as in Freedom (2.0): Richard Stallman and the Free Software Revolution. The Free Software Foundation, Boston (2010)
99. Wurster, L.: Open Source Software Hits a Strategic Tipping Point, *Harvard Business Review*, (2011)
100. Ye, Y., Kishida, K.: Toward an understanding of the motivation Open Source Software developers. In: Proceedings of the 25th international conference on software engineering. pp. 419–429. (2003)
101. Yildirim, N., Ansal, H.: Foresighting FLOSS (free/libre/open source software) from a developing country perspective: The case of Turkey. Technovation. 31 (12), 666–678 (2011)