Open Source Software Evaluation, Selection, and Adoption: a Systematic Literature Review

Valentina Lenarduzzi
LUT University
Lahti, Finland
valentina.lenarduzzi@lut.fi

Davide Taibi
Tampere University
Tampere, Finland
davide.taibi@tuni.fi

Davide Tosi, Luigi Lavazza, Sandro Morasca
University of Insubria
Varese, Italy
davide.tosi;luigi.lavazza;sandro.morasca@uninsubria.it

***Abstract*—*Background.* Open Source Software (OSS) is experiencing an increasing popularity both in industry and in academia. *Aim.* We investigated models for the selection, evaluation, and adoption of OSS, focusing on factors that affect most the evaluation of OSS. *Method.* We conducted a Systematic Literature Review of 262 studies published until the end of 2019,**

to understand whether OSS selection is still an interesting topic for researchers, and which factors are considered by stakeholders and are assessed by the available models. *Result.* We selected 60 primary studies: 20 surveys and 5 lessons learned studies elicited the motivations for OSS adoption; 35 papers proposed several OSS evaluation models focusing on different technical aspects. This Systematic Literature Review provides an overview of the available OSS evaluation methods, highlighting their limits and strengths, based on the wide range of technicalities and aspects explored by the selected primary studies. *Conclusion.* OSS producers can benefit from our results by checking if they are providing all the information commonly required by potential adopters. Users can learn how models work and which models cover the relevant characteristics of OSS they are most interested in.

***Index Terms*—Open-source software, software selection, software adoption, software quality models**

I. Introduction

The foundation of the Open Source Initiative dates back to 1998[1]. Since then, Open Source Software (OSS) has been gaining growing interest and popularity both in industry and in academia, but there are a few reasons that prevent OSS from being even more widespread. First, reasonably detailed and up-to-date user documentation is sorely lacking for a large number of OSS projects. So, many potential end-users get frustrated when they start evaluating an OSS product that they would like to adopt, because of the large degrees of uncertainly about the intent and technical aspects of the OS product and project [1], [2]. Second, many end-users do not trust OSS products, because they confuse OSS solutions with free software. Moreover, objective assessment of the quality of OSS solutions is quite difficult to accomplish [3]–[7], especially because collecting the information required for evaluation is difficult [8].

To support the adoption of OSS, several evaluation models have been proposed (e.g., [SP1], [SP2], [SP12] an [SP13]). These models aim at supplying potential users with the information needed for deciding whether to use a given OSS product or not. The goal of this Systematic Literature Review (SLR) is to identify, analyze, and classify the OSS models used during the selection, evaluation, and adoption of OSS. We consider with particular attention the evaluation areas and the factors addressed by the OSS evaluation models.

The ultimate goal of the SLR is to inform practitioner on the OSS evaluation models that are available, the kind of evaluations that can be achieved, and their effectiveness and extent. Our results will help practitioners make effective decisions about which models are more suitable in their context and how reliable and exhaustive the achieved indications can be. Our results will also enable researchers to identify gaps in the current OSS adoption literature.

We analyzed 262 studies published from the foundation of the Open Source Initiative (1998) up to December 2019. We consider both models targeted to end users, such as models to support end-users in selecting the most appropriate OSS application, and models for practitioners, including the selection of the most appropriate OSS components, libraries, frameworks, and other tools to be integrated in the software they deliver. The complete list of the 262 papers selected is available in the replication package [2].

Many papers concerning OSS adoption have been published in the last twenty years. These papers mainly introduce OSS quality models or adoption processes that allow users to select OSS products or evaluate some specified characteristics, based on a predefined process.

Hauge et al. [9] reported a SLR on the adoption of OSS in software-intensive organizations. They analyzed how companies select CASE tools and OSS components to be integrated in their development process, considering papers published between 1998 and 2008. The authors suggest an OSS adoption framework addressing Using OSS CASE tools, Integrating OSS Components, Participating in OSS Communities, Providing OSS Products, Using OSS Development Practices.

Sbai et al. [10] investigated the information that can be automatically extracted from OSS repositories to support the adoption process. They also reported a classification of the information provided in seven papers, considering seven adoption models and two surveys. Their purpose was different from ours, and they only reported the factors related to the infor-

[1]https://opensource.org

[2]www.taibi.it/raw-data/SLR_OSS_Adoption_RawData

mation that can be automatically extracted from repositories, without reporting on other factors. Moreover, the selection of the models was performed in a non-systematic manner.

Our work differs from previous reviews as follows.

Goal: We aim at comparing the existing OSS adoption and selection models and understanding the main factors influencing the adoption of OSS. We consider models for developers that need to select the most appropriate OSS component and models for end users that need to select an existing product. Instead, Hauge et al. [9] aimed at understanding how companies adopt OSS and, especially, how OSS CASE tools are adopted. Sbai et al. [10] aimed at identifying which information should be considered by an OSS selection tool.

Timeframe: Our review is more recent, as it includes studies published until 2019, while Hauge et al. reviewed papers between 1998 and 2008.

Comprehensiveness: Our search strategy encompassed eight search engines, as suggested by Kitchenham [11], [12] and uses the systematic snowballing process [13]. Moreover, we included papers from relevant journals and conferences and all papers referenced in our identified studies. As a result, we analyzed many more papers than surveied in [9] and [10].

Analysis: We provide a short summary and a detailed result of each paper (see Appendix A), which can be useful as a reference for future work or as a quick overview of the selected works.

We can summarize our contributions as follows:

- We identified a set of 262 studies addressing OSS selection models, which researchers can use as a basis for future investigations into OSS adoption models.
- A subset of 60 studies, of which 20 surveys and 5 lesson learned reporting the motivation for the adoption of OSS, and 35 studies reporting evaluation models. For each model identified, we considered the availability of support tools as well as empirical validations, and we extracted a set of factors, measures, and information that characterize the analyzed models.
- A synthesis of the evolution and the current motivation for OSS adoption.
- A synthesis of the existing OSS adoption and evaluation models.

Paper structure: The remainder of this paper is organized as follows: In Section 2, we present the SLR process and the methodology adopted in this work. We also define the research questions and describe the criteria for the selection of primary studies. In Section 3, we illustrate the information extracted from the reviewed papers. Section 4 discusses the results obtained. In Section 5, we explain the threats to validity of the SLR. Finally, in Section 6 we draw conclusions and provide an outlook on future work.

II. Research Method

We adopted the protocol proposed by Kitchenham et al. [11], [12], in combination with the systematic snowballing process proposed by Wohlin [13].

TABLE I: Inclusion and exclusion criteria

Criteria	Assessment Criteria	Step
Inclusion	Papers not peer-reviewed (i.e. blog, forum ...) referenced at least in two of the selected studies	T/A
Exclusion	Papers not fully written in English	T/A
	Duplicate paper (only consider the most recent version)	T/A
	Position papers and work-plan (i.e. paper that does not report results)	T/A
	Only the latest version of the papers (e.g., journal papers that extend conference papers will be excluded if they are referred to the same dataset)	T/A
	Papers where models, methods, or tools are not clearly reported in the abstract.	A

A. Goal and Research Questions

Based on the goals stated in the introduction, we defined two main research questions (RQs).

RQ1. What OSS evaluation, selection, and adoption models have been proposed so far?

RQ2. What are the common factors that are considered in the selection, evaluation, and adoption process?

B. Search Strategy

1) Bibliographic Sources Identification: We combined automatic and manual search activities to optimize results. The bibliographic sources we selected are *ACM digital Library, IEEEXplore Digital Library, Science Direct, Scopus, Google Scholar, Citeseer library, Inspec, and Springer link.*

2) Inclusion and Exclusion Criteria: We defined criteria to be applied to title and abstract (T/A) or to full-text (F), as reported in Table I. We considered also papers that may have had an impact in academic studies and practical application, even though they did not go through the usual reviewing process of scientific publications.

3) Primary Study Selection Query: We applied the following query:

```
(``evaluation'' OR ``selection'' OR
``adoption'' OR ``evaluation model*''
OR ``selection model*'' OR ``adoption
model*'') AND (``Open Source Software'' OR
``OSS'' OR ``FLOSS'' OR ``Libre Software''
OR ``Free Software'')
```

4) Search and Selection Process: The search was conducted in December 2019 including all the publications available up to that date. The application of the searching terms returned 2504 unique papers.

Applying Inclusion and Exclusion Criteria to title and abstract: After testing and refining the applicability of inclusion and exclusion criteria as suggested by Kitchenham [12], we applied the refined criteria to the retrieved papers. Each paper was read by two authors and in case of disagreement a third author was involved to decide whether to include or exclude the paper. Out of 2504 retrieved primary studies, we retained 262.

Full text reading: We read the 262 papers, applying the criteria defined in Table I: 68 papers were selected as relevant.

Snowballing: We performed the snowballing [13] iteratively, until no additional primary studies were identified.

Snowballing resulted in identifying 7 additional relevant papers, which were added to the selected set to be reviewed.

5) Assessing the Suitability of the Selected Papers: We aimed at selecting only papers that contain models or methods for OSS selection, evaluation, or adoption. To this end, we applied the assessment criteria given in Table III: each primary study was assigned a score on a four-point Likert scale (poor, fair, good, excellent), and only studies rated "fair" or better were selected. As a result, of the 75 primary studies previously selected, only 60 were finally retained.

TABLE II: Search and selection results

Step	# papers
Retrieval from bibliographic sources	2504
Inclusion and exclusion criteria	-2271
Fulfill reading	-165
Snowballing	7
Assessment criteria	-15
Papers identified	**60**

The 60 papers considered in this review—as well as the 15 finally discarded—are listed in the References.

C. Data Extraction

Seven pieces of information were extracted from each study as reported in Table IV.

TABLE III: Assessment criteria definition

Id	Assessment Criteria	Criteria Definitions
AC1	Evaluation purpose	The paper clearly states the evaluation areas
AC2	Model type	The paper clearly identifies the approach defined to evaluate OSS and the support provided by the methodology for the evaluation process (i.e., definition of checklists, guidelines, keywords, ...)
AC3	Is there a selection, evaluation, and/or adoption model/methodology reported in the paper?	The paper clearly shows evidence about models or methodologies for the selection, evaluation, and/or adoption process. We also took into account contributions not clearly performed for OSS, but with clear evidence of usage and validation in an OSS context.
AC4	Model Usability/Repeatability	The paper clearly describes repeatable and usable models, who describe all the steps and the measures needed to apply them.
AC5	Factors, Measures and Information to be analyzed	The paper clearly identifies set of factors, measures, and information (both standard or newly defined in the paper) to be analyzed.
AC6	Empirical validation	The paper clearly reports the availability of empirical validation for the proposed model/methodology, such as case studies, controlled experiments, and others.

III. Results

The 60 selected primary studies were published between 2004 and 2019, with a peak—accounting for 40% of the papers—between 2010 and 2011 (see Figure 1). The selected papers have an average of 15.3 citations per paper, a maximum of 142, and a minimum of 0. The most influential papers only address two topics: Motivations and Adoption Model. All other topics (such as Risk model, Reliability model, etc.) were not investigated widely.

TABLE IV: Data Extraction Form and Related Research Questions (if applicable)

Extracted Data	RQs
Publication Year	n.a.
Number of Citations extracted from Google Scholar. We adopted Scholar instead of other bibliographic sources since we believe it is the most comprehensive citation counter, also including citations from practitioners white-papers	n.a.
Goal of the paper. For instance, the paper goal can be to evaluate the OSS maturity or quality	RQ1
How it was built. Models can be built based on the experience of the authors, based on the results of a case study, of a set of interviews or a combination of the previous one.	RQ1
How it works. Models can be applied with different approaches such as checking the availability of a set of factors listed in a checklist without downloading or trialing the tool or downloading the OSS product and measuring some of its characteristics.	RQ1
Availability of tools to apply the models or methodologies	RQ1
Factors, measures, and information to be analyzed	RQ2

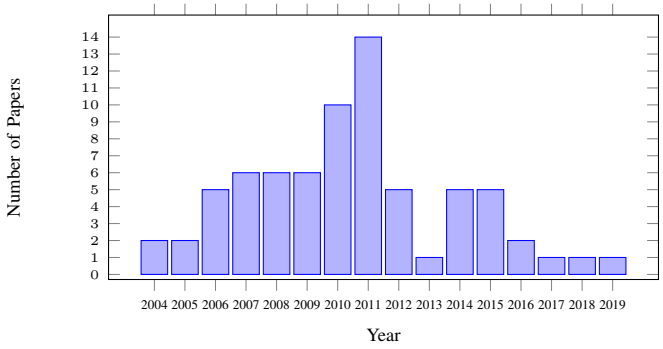

Fig. 1: Publications distribution

As for the research questions given in Section II, the primary studies analyzed in this work contribute as follow.

A. RQ1: What OSS evaluation, selection, and adoption models have been proposed so far?

To answer this RQ, we classify the selected papers in three sub-categories.

Models. Algorithms or methods to evaluate specific characteristics of the OSS under consideration.

Surveys. Collections of information from users that already adopted OSS; these are useful to provide potential adopters of OSS with insights form those who already use OSS.

Lessons Learned. Context-specific evaluation of OSS, derived from case studies, that could be valuable to users that need to adopt OSS in similar contexts.

35 primary studies (58.3%) propose models for different characteristics of OSS products such as their maturity or quality (see column "Scope" of Table V).

20 studies (33.3%) are surveys on the adoption of OSS, describing adoption processes, motivation and issues in different contexts and analyzing the characteristics or reported lessons learned by performing case studies or surveys on users that are using or that are adopting OSS, as reported in Table VI.

The remaining 5 papers (8.3%) derived lessons learned from case studies.

We classified the selected papers according to four characteristics, as reported in Table V and Table VI. The meaning of the tables' columns is given below.

Scope: The goal of the investigation. For instance, some surveys investigate the factors that affect the quality of OSS and describe models that support users to evaluate the quality of OSS products.

How it was built: The process adopted in primary studies:

- *Case Studies*. Single or multiple case studies performed in companies or on OSS projects.
- *Interviews* are often adopted to collect information from OSS users.
- *Experience*. OSS evaluations based on the experience of the authors.

How it works: The approach to evaluate OSS is classified in checklists, measurement and the combination of checklists and measurement:

- *Checklist*. This is an experience-based model, where the evaluation is commonly based on two steps. The user needs to extract information concerning an OSS product according to a checklist. In most cases, this is done by just crawling for information on the official website; in other cases, the product is also trialed.
- *Measurement*. Several papers propose the application of multiple measures that can be collected manually or from existing tools. As an example, some adoption models recommend to analyze the size of the source code in terms of lines of code, while other requires to identify the different development languages adopted in the tool.

As for the models' scope, adoption and evaluation are by far the most common ones. Other purposes, such as those related to evaluating process, testing, reliability, and security, are defined as extensions of closed source models by adding OSS community aspects and source code analysis as criteria, which are not available in closed source software. Less than half of the 20 selected surveys investigate adoption motivations. The other half investigate different OSS characteristics such as economic factors, testing processes, and quality characteristics.

The vast majority of models (21 out of 35) identified the factors to be evaluated based on a single case study, while 4 models combined the motivations collected from OSS user's interviews with a case study. Five models base their factors only on the experience of the authors and five models on the results of a set of interviews (VI, column "How it was built").

The vast majority of the models require that an importance weight be assigned to each factor. Models built based on case studies or on authors' experience commonly require the specification of the factors' importance based on the experience of the assessor, while models built based on interviews commonly propose to weigh the factors based on the importance indicated by the interviewees.

14 models are based on the combination of checklist and measurement, 13 require only the evaluation of the list of factors reported in a checklist, 8 propose the measurement of a set of factors.

As for surveys and lessons learned, 13 out of 20 aim at identifying the adoption motivations while the remaining 7 have different scopes such as reliability, trustworthiness, risk management and others.

Providing stakeholders with tools that support the application of OSS evaluation models would likely increase the applicability of these models and thus make the adoption of OSS solutions more widespread. However, only few of the papers analyzed in this SLR provide stakeholders with tool support: 22 papers list the availability of a set of tools, 4 papers provided a set of detailed checklists and guidelines to simplify the work of the stakeholders, and 8 papers reported the implementation of their proposed theoretical and statistical models to support the evaluation of OSS products, but did not publish the tools (Table VII).

Only two tools mentioned in the reviewed papers have a web page that is still available and up-to-date (Table VII). Once again, this highlights the need for tools that simplify the dissemination of selection/adoption/evaluation models in industry.

B. RQ2: What are the common factors that are considered in the selection, evaluation, and adoption process?

We analyzed the factors considered as important by models (Table VIII) and we compared them with those proposed as evaluation factors in the surveys and lesson learned (Table IX).

The factors reported in Table VIII and Table IX were classified in eight main groups: *Community and Adoption, Development process, Economic, Functionality, Licence, Operational software characteristics, Quality, Support and Service.* In the Tables, we also report the sub-factors that are considered by the primary studies.

In Figure 2, we plot the percentage of primary studies dealing with models and primary studies reporting Surveys or Lessons learned that supported each factor and sub-factor. For instance, the factor "Community Size" has been mentioned as important by 6 out of 35 (17%) primary studies describing models and by 3 out of 25 (12%) primary studies reporting surveys or lessons learned.

As shown in Figure 2, not all the factors considered important by users are taken into account in the evaluation models.

Functional suitability is considered much more important by models than by surveys and lesson learned.

Economic factors are considered at least five times more frequently by surveys than models. However, Licence characteristics are by far more important in models. This could explain the lower importance of economic in models, since Licence could be a proxy-factor for acquisition costs.

Community and Adoption-related information are considered both by surveys and models, with some small variance in one factor (coordination considered only by surveys).

Support and Service is by far more frequently considered by models. This could be explain because of the easiness of col-

TABLE V: RQ1: Reviewed papers dealing with Evaluation Models.

ID	Scope	How it was built	How it works
[SP1]	Maturity	Experience	Checklist
[SP2]	Maturity	Experience	Checklist
[SP4]	Security	Case Study	Checklist Measurement
[SP7]	Maintenance	Experience	Checklist
[SP8]	Risk	Case Study	Checklist Measurement
[SP9]	Quality	Case Study	Checklist Measurement
[SP11]	Testing Trustworthiness	Case Study	Checklist Measurement
[SP12]	Adoption	Case Study	Checklist Measurement
[SP13]	Adoption	Case Study	Checklist Measurement
[SP15]	Trustworthiness	Experience	Checklist Measurement
[SP16]	Trustworthiness	Case study Interviews	Measurement
[SP17]	Selection	Case Study	Checklist Measurement
[SP18]	Adoption Cost	Case Study	Checklist
[SP19]	Evaluation	Case Study Interviews	Checklist
[SP21]	Testing process	Case Study	Measurement
[SP22]	Trustworthiness	Case Study Interviews	Checklist Measurement
[SP23]	Community	Case Study	Checklist
[SP24]	Community	Case Study	Checklist Measurement
[SP25]	Maturity	Case Study Interviews	Checklist Measurement
[SP28]	Adoption Cost	Case Study	Checklist
[SP30]	Adoption Motivations	Case Study	Checklist
[SP32]	Trustworthiness	Interviews	Measurement
[SP33]	Testing	Case Study	Measurement
[SP34]	Quality	Interviews	Checklist Measurement
[SP38]	Web portal information	Interviews	Checklist
[SP41]	Web portal information	Interviews	Checklist
[SP43]	Reliability	Case Study	Measurement
[SP48]	Testing	Case Study	Checklist
[SP49]	Adoption Motivations	Case Study	Checklist
[SP51]	Quality	Interviews	Checklist
[SP52]	Risk	Experience	Measurement
[SP55]	Reliability	Case Study	Measurement
[SP56]	Reliability	Case Study	Measurement
[SP59]	Adoption Motivations	Case Study	Checklist Measurement
[SP60]	Resilience	Case Study	Checklist Measurement

TABLE VI: RQ1: Reviewed papers reporting Surveys and Lesson Learned.

ID	Scope	How it was built	How it works
[SP3]	Adoption Motivations	Interviews	
[SP5]	Adoption Motivations	Interviews	
[SP6]	Adoption Motivations	Interviews	Checklist
[SP10]	Selection process	Interviews	
[SP14]	Adoption Motivations	Interviews	
[SP20]	Adoption Motivations	Interviews	
[SP26]	Development process	Case Study	
[SP27]	Economic factors	Interviews	
[SP29]	Quality	Interviews	
[SP31]	Adoption patterns	Case Study	
[SP35]	Trustworthiness	Interviews	
[SP36]	Adoption Motivations	Interviews	
[SP37]	Adoption Motivations	Case Study	
[SP39]	Adoption Motivations	Interviews	
[SP40]	Cloud Evaluation	Interviews	Checklist
[SP42]	Adoption Motivations	Interviews	
[SP44]	Adoption Motivations	Interviews	
[SP45]	Marketing and Communication	Interviews	
[SP46]	Reliability	Interviews Case Study	Measurement
[SP47]	Testing	Interviews	
[SP50]	Quality	Interviews	
[SP53]	Adoption Motivations	Interviews	
[SP54]	Adoption Motivations	Interviews	
[SP57]	Adoption Motivations	Case Study	
[SP58]	Risk	Case Study	

TABLE VII: RQ1: The tools proposed by the evaluation models

Paper ID	Tool Name	Tool URL
[SP4]	ROSEN (real-time OpenSSL execution monitoring system)	Not available
[SP11]	RAP Tool	Not available
[SP13]	QSOS	www.qsos.org
[SP15]	SQO-OSS	Not available
[SP22]	OMM Tool	Not available
[SP31]	T-Doc Tool	Not available
[SP32]	MAcXim	Not available
[SP33]	QualiPSo Trustworthiness Checklist	Not available
[SP34],[SP35],[SP42] [SP45],[SP46],[SP48] [SP49],[SP50],[SP56]	Theoretical and statistical model	Not available
[SP27],[SP30],[SP44]	MOSST	Not available
[SP36][SP37]	OP2A Checklist	Not available
[SP59]	OSSPal	www.osspal.org

lecting information about number of contributors, professional support, ecc.

Different *Operational software characteristics*, such as Maturity, Usability and Compliance with standards, are also considered more important by models, with the exception of multi-platform support.

The *Availability of development process documentation* is considered more frequently by models.

Quality-related factors are considered with a similar frequency both in models and surveys.

IV. Discussions

The analysis of the literature shows a discrepancy between the information proposed in evaluation models and the information considered useful by practitioners. As also highlighted in [8], the collection of the factors required to evaluate an OSS product is very time-consuming, mainly because most of the information required is not commonly available on the OSS product portals.

The result of this work could be highly beneficial for OSS producers, since they could check if they are providing all the information commonly required by who is evaluating their products, and maximize the likelihood of being selected. The result can also be useful to potential OSS adopters, who will speed-up the collection of the information needed for the evaluation of the product.

Even in case OSS producers do not enhance their portals by providing the information listed in Tables VIII and IX, the results of this work could be useful for practitioners that need to evaluate an OSS product. The list of factors can be effectively used as checklist to verify if all the potentially important characteristics of OSS have been duly evaluated. For instance, a practitioner could have forgotten to evaluate the trend of the community activity and he/she could adopt an OSS

TABLE VIII: The factors proposed by the evaluation models

Factor	Sub-factors
Functionality ([SP18], [SP49], [SP55],[SP56])	Functional Suitability ([SP9],[SP22],[SP34])
Economic ([SP19])	Cost ([SP1],[SP2],[SP12],[SP18]) Total Cost of Ownership (TCO) ([SP34]) Return On Investment (ROI) ([SP34]) Differentiate from competitors ([SP1],[SP2][SP12]) Innovativeness ([SP34]) Clear project management ([SP1],[SP2],[SP13], [SP17],[SP59])
Licence	Licence type ([SP1], [SP2],[SP12],[SP13],[SP17], ([SP18],[SP19], [SP24],[SP34],[SP59]) Law conformance ([SP34])
Community and Adoption ([SP11],[SP19],[SP49],[SP55], [SP56])	Community size ([SP2],[SP8],[SP13],[SP17],[SP24], [SP59]) # developers ([SP2],[SP12],[SP13],[SP15],[SP17], [SP24],[SP52], [SP54],[SP59]) Availability of forum ([SP34])
Support and Service ([SP1], [SP2],[SP12],[SP13],[SP17], [SP24],[SP15],[SP17],[SP18], [SP19],[SP24],[SP43],[SP49], [SP52],[SP59])	Contributors ([SP1],[SP2],[SP12],[SP13],[SP15], [SP34],[SP52],[SP59]) Quality of professional support ([SP1],[SP2],[SP12], [SP13],[SP17],[SP24],[SP34]) Training ([SP1],[SP2],[SP13],[SP17],[SP34])
Operational software characteristics ([SP55], [SP56])	Maturity ([SP1], [SP2],[SP12],[SP13],[SP15],[SP17], [SP24],[SP34],[SP59]) "Trialability" ([SP34]) Independence from other sw ([SP12],[SP34]) Adopted sw architecture ([SP59]) Development language ([SP1],[SP2],[SP12],[SP59]) Multiplatform support ([SP1],[SP2],[SP59]) Standard compliance ([SP1],[SP2],[SP12],[SP13], [SP17])
Development Process [SP19],[SP49],[SP52],[SP55], [SP56]	Clear Roadmap ([SP34]) Availability of development process doc. ([SP1],[SP2], [SP12],[SP13],[SP15],[SP17],[SP24],[SP34],[SP59]) Books/Online ([SP12],[SP13],[SP15],[SP17],[SP24], [SP34]) Code Documentation ([SP34]) Architectural Documentation ([SP17], [SP34])
Quality ([SP19],[SP21], [SP43],[SP48],[SP49],[SP55], [SP56])	Code quality([SP12],[SP15],[SP17],[SP21], [SP24],[SP32],[SP33],[SP34],[SP38],[SP41],[SP59]) Reliability ([SP1],[SP2],[SP11],[SP12],[SP13],[SP15], [SP17],[SP43],[SP59]) Interoperability / Compatibility ([SP34]) Customization Easiness ([SP34]) Flexibility ([SP1],[SP2],[SP13],[SP17]) Maintainability ([SP12],[SP13],[SP15],[SP17],[SP24], [SP32],[SP34]) Security ([SP1],[SP2],[SP15],[SP17],[SP24],[SP59]) Performance ([SP1],[SP2],[SP15],[SP17],[SP34],[SP59]) Modularity ([SP1],[SP2],[SP13],[SP17],[SP21],[SP32], [SP34]) Usability ([SP1],[SP2],[SP17],[SP34],[SP49]) Portability ([SP1],[SP2],[SP8],[SP17],[SP34]) Adaptability ([SP1],[SP2],[SP13]) Defect-proneness ([SP7],[SP32],[SP34],[SP38],[SP41]) Change-proneness ([SP7],[SP32],[SP34],[SP38],[SP41]) Testability ([SP21],[SP32]) [SP33],[SP34],[SP38], [SP41])

TABLE IX: The factors proposed by the evaluation surveys and lessons learned

Factor	Sub-factors
Functionality [SP6]	Lack of Drivers in CSS ([SP36])
Economic	Cost ([SP6],[SP3],[SP27],[SP31],[SP35],[SP36],[SP37], [SP39], [SP44],[SP54]) TCO ([SP27],[SP35],[SP36] ROI ([SP27], [SP35]) Differentiate from competitors ([SP3],[SP5],[SP27], [SP35],[SP54]) Innovation ([SP3],[SP20],[SP36], [SP54]) Clear project management ([SP6],[SP40])
Licence	Licence type ([SP6],[SP35],[SP54]) Law conformance ([SP6],[SP35])
Community and Adoption ([SP35],[SP42],[SP44], [SP45],[SP54])	Community size ([SP6], Coordination ([SP31],[SP35],[SP54]) # developers ([SP6],[SP54]) Availability of forum ([SP35],[SP54])
Support and Service ([SP6],[SP36],[SP37], [SP44],[SP53],[SP54])	Contributors ([SP35])
Operational sw characteristics	Maturity([SP54]) "Trialability" ([SP3],[SP5]) Independance from other sw ([SP54]) Development language ([SP35]) Multiplatform support ([SP6],[SP40],[SP54]) Standard compliance ([SP35])
Development Process ([SP6],[SP31])	Clear Roadmap ([SP6],[SP20]) Documentation ([SP29],[SP35],[SP54] Books/Online ([SP35],[SP54] Architectural Documentation ([SP35],[SP54])
Quality	Code quality ([SP6],[SP35],[SP42],[SP44],[SP54]) Reliability ([SP3],[SP6],[SP35],[SP37],[SP53],[SP54]) Interoperability/Compatibility ([SP3],[SP29],[SP39], [SP54]) Customization Easiness ([SP37],[SP54]) Flexibility ([SP54]) Maintainability ([SP6],[SP35],[SP42],[SP44],[SP54]) Security ([SP35],[SP40],[SP44],[SP54]) Performance ([SP35]) Usability ([SP29],[SP35]) Portability([SP35]) Adaptability ([SP54]) Testing ([SP6],[SP47]) Efficiency ([SP29])
Other	Personal Interest ([SP5],[SP14],[SP42],[SP54]) Regulations and Political Influence ([SP6],[SP44]) Accomplishment ([SP14]) Ethical Reasons([SP42],[SP54]) Experience Stimulation ([SP14],[SP20])

product that has a "dissolving" community: this could create problems in the future because of the lack of maintenance and updates. The usage of checklist would allow practitioners to double check if they considered all factors, thus reducing the potential unexpected issues that could come up after the adoption.

Future Research Directions. As a result of our findings, we propose the following directions for future research in this area. Focus on the definition of a common model (which may be obtained by merging multiple available approaches) and favor its adoption through rigorous and extensive validation in industrial settings. This could increase the validity of the model and thus its dissemination in industry, where OSS is still not widely adopted. Several models already exist but, according to the results of our SLR, they have not been strongly validated and, as a consequence, adoption has been limited.

Try to target the models at quality factors that are of real interest for stakeholders. Most of the available models focus on the overall quality of the product, but few of them are able to assess each single factor that composes the overall quality of the OSS product. This can complicate the assessment of OSS products by stakeholder, who are interested in specific quality factors: e.g., developers are likely more interested in reliability or testability aspects, while business people may be more interested in cost or maintenance factors, etc..

Develop tools that support the research directions listed above (i.e., tools able to support and simplify the applicability of the proposed models during the evaluation of OSS products). Most of the tools mentioned in the primary studies are prototypes and most of them are not available or maintained anymore.

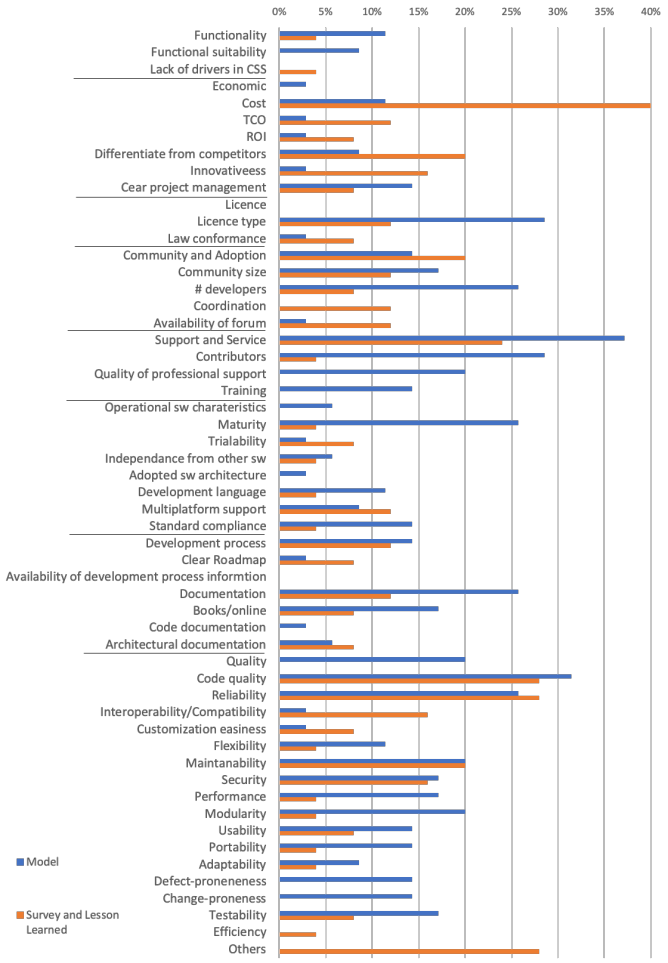

Fig. 2: Percentage of papers reporting the different factors

V. Threats to Validity

Construct validity. The terms adopted in our study are sufficiently stable to be used as search strings. In order to assure the retrieval of all papers on the selected topics, we searched broadly in general publication databases that index most well-reputed publications. Moreover, we also included papers from the gray literature when referenced by the primary studies. *Reliability*. We defined search terms and applied procedures that can be replicated by others. In this SLR, the results were evaluated by three different researchers in order to minimize errors during the data collection process. We only used descriptive statistics, so threats to *Internal validity* are minimal. *External validity* Our SLR is generalizable as far as the primary studies cover completely and accurately the topics of interest. We believe that the selected primary studies provide a good overview of the selection, evaluation, and adoption of OSS.

VI. Conclusion

The SLR reported in this work was carried out to acquire knowledge on the state of the art in the area of models for quality evaluation during the selection process of OSS products. Specifically, we focused our study on two research questions to understand (1) what factors are mainly discussed by stakeholders during the selection process; (2) what factors are actually assessed by the available models.

We identified 60 relevant primary studies. The relatively small number of primary studies identified can be explained by the fact that the definition of models for OSS analysis emerged only around the year 2004. However, the topic appears to be of interest both for the research community (the research field is still active with several primary studies in the second decade of this century) and for industry, where the interest for OSS products has been growing over time but where the adoption of OSS is still limited due to the difficulty of assessing its quality.

We believe that the identified primary studies present a good deal of material that can be used to provide answers to the research questions under focus.

References

[1] L. Lavazza, S. Morasca, D. Taibi, and D. Tosi, "Predicting OSS Trustworthiness on the Basis of Elementary Code Assessment," in *International Symposium on Empirical Software Engineering and Measurement*, ser. ESEM '10, 2010, pp. 36:1–36:4.

[2] ——, "OP2A: How to Improve the Quality of the Web Portal of Open Source Software Products," in *Web Information Systems and Technologies*, J. Filipe and J. Cordeiro, Eds., 2012, pp. 149–162.

[3] V. del Bianco, L. Lavazza, S. Morasca, D. Taibi, and D. Tosi, "An Investigation of the Users' Perception of OSS Quality," in *Open Source Software: New Horizons*, 2010, pp. 15–28.

[4] S. Morasca, D. Taibi, and D. Tosi, "T-DOC: A Tool for the Automatic Generation of Testing Documentation for OSS Products," in *Open Source Software: New Horizons*, 2010, pp. 200–213.

[5] V. Del Bianco, L. Lavazza, S. Morasca, D. Taibi, and D. Tosi, "The QualiSPo Approach to OSS Product Quality Evaluation," in *Int. Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development*, 2010, pp. 23–28.

[6] S. Morasca, V. del Bianco, D. Taibi, and L. Lavazza, "A Survey on Open Source Software Trustworthiness," *IEEE Software*, pp. 67–75, 2011.

[7] L. Lavazza, S. Morasca, D. Taibi, and D. Tosi, "An Empirical Investigation of Perceived Reliability of Open Source Java Programs," in *Annual ACM Symposium on Applied Computing*, 2012, pp. 1109–1114.

[8] Y. Kamei, T. Matsumoto, K. Yamashita, N. Ubayashi, T. Iwasaki, and T. Shuichi, "Studying the Cost and Effectiveness of OSS Quality Assessment Models: An Experience Report of Fujitsu QNET," *IEICE Transactions on Information and Systems*, pp. 2744–2753, 11 2018.

[9] O. Hauge, C. Ayala, and R. Conradi, "Adoption of Open Source Software in Software-intensive Organizations - A Systematic Literature Review," *Inf. Softw. Technol.*, vol. 52, no. 11, pp. 1133–1154, Nov. 2010.

[10] N. Sbai, V. Lenarduzzi, D. Taibi, S. B. Sassi, and H. H. B. Ghezala, "Exploring information from OSS repositories and platforms to support OSS selection decisions," *Information and Software Technology*, vol. 104, pp. 104 – 108, 2018.

[11] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," 2007.

[12] B. Kitchenham and P. Brereton, "A systematic review of systematic review process research in software engineering," *Information & Software Technology*, vol. 55, no. 12, pp. 2049–2075, 2013.

[13] C. Wohlin, "Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering," in *International Conference on Evaluation and Assessment in Software Engineering*, ser. EASE '14, 2014, pp. 38:1–38:10.

Appendix A: The Selected Papers

[SP1] F.W. Duijnhouwer, C. Widdows, Open Source Maturity Model. Capgemini Expert Letter. 2003.

[SP2] B.Golden. The Open Source Maturity Model (OSMM). In Succeeding with Open Source 2004. Addison-Wesley Information Technology Series.

[SP3] J.Dedrick and J. West. An exploratory study into open source platform adoption, 37th Annual Hawaii International Conference. 2004. pp. 1-10,

[SP4] S.Choi, Y.Kang and G. Lee. A security evaluation and testing methodology for open source software embedded information security system, ICCSA International Conference. 2005. Vol 3481, pp. 215-224.

[SP5] E. Glynn, B. Fitzgerald and C.Exton. Commercial adoption of open source software: an empirical study, International Symposium on Empirical Software Engineering. 2005. pp. 17-18.

[SP6] D. Cruz, T. Wieland and A. Ziegler. Evaluation criteria for free/open source software products based on project analysis, Software Process Improvement and Practice. 2006. Vol. 11(2), pp. 107-122.

[SP7] T. Koponen. Evaluation Framework for Open Source Software Maintenance, International Conference on Software Engineering Advances. 2006. pp. 52.

[SP8] S. Kumar and L. Wang. Metrics to Support Open Source Software Adoption Decisions, Twelfth Americas Conference on Information Systems. 2006. pp. 839-847.

[SP9] W. Sung, J. Kim and S. Rhew. A Quality Model for Open Source Software Selection, International Conference on Advanced Language Processing and Web Information Technology. 2007. pp. 515-519.

[SP10] Y. Lee, J. Kim, W. Choi and S. Rhew. A Study on Selection Process of Open Source Software, International Conference on Advanced Language Processing and Web Information Technology. 2007. pp. 568-571.

[SP11] A. Immonen and M. Palviainen. Trustworthiness Evaluation and Testing of Open Source Components, International Conference on Quality Software (QSIC). 2007. pp. 326-321.

[SP12] D. Taibi, L. Lavazza and S. Morasca. OpenBQR: a framework for the assessment of OSS. Open Source Development, Adoption and Innovation. 2007. pp. 173-186

[SP13] Atos Origin. Method for Qualification and Selection of Open Source Software (QSOS), from http://www.qsos.org (Last visited: August, 2016)

[SP14] L. Yan; T. Chuan-Hoo and T. Hock-Hai. Open Source Software Adoption: An Investigation Into Motivations and Amotivations of Users,Proceedings of the Fourteenth Americas Conference on Information Systems. 2008. pp. 1-11,

[SP15] I. Samoladas, G. Gousios, D. Spinellis and I. Stamelos. The SQO-OSS Quality Model: Measurement Based Open Source Software Evaluation, Open Source Development, Communities and Quality. 2008. Vol. 275, pp. 237-248.

[SP16] D.Taibi, V. Del Bianco, D. Dalle Carbonare, L. Lavazza and S. Morasca. Towards The Evaluation Of OSS Trustworthiness: Lessons Learned From The Observation Of Relevant OSS Projects, Open Source Development, Communities and Quality. 2008. Vol. 275, pp. 389-395.

[SP17] A.I. Wasserman, M.Pal, C.Chan. The Business Readiness Rating: a framework for evaluating open source Technical Report 2006

[SP18] B. Russo and G. Succi. A Cost Model of Open Source Software Adoption. IJOSSP. 2009. Vol 1.3, pp. 60-82.

[SP19] K. van den Berg. Open source software evaluation Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives. Chapter 1.6. pp. 197-210.

[SP20] J. Heili , S. Assar. An empirical enquiry into the adoption of open source software by individual users in France ADIS. International Conference Information Systems Barcelona, 2009.

[SP21] S. Morasca, D.Taibi, D. Tosi. Towards Certifying the Testing Process of Open-Source Software: New Challenges or Old Methodologies? ICSE Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development. 2009.

[SP22] V.Del Bianco, L. Lavazza, S. Morasca, D.Taibi. Quality of Open Source Software: The QualiPSo Trustworthiness Model 5th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2009. 2009. pp. 3-6.

[SP23] V. Lenarduzzi, D. Tosi, L. Lavazza, S. Morasca. Why Do Developers Adopt Open Source Software? Past, Present and Future, International Conference on Open Source Systems. 2019. pp. 104-115.

[SP24] M. Soto and M. Ciolkowski. The QualOSS open source assessment model measuring the performance of open source communities. 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM '09). 2009. pp. 498-501.

[SP25] E. Petrinja, R. Nambakam, and A. Sillitti. Introducing the OpenSource Maturity Model. Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development (FLOSS '09). 2009.

[SP26] L. Lavazza, S. Morasca, D. Taibi, D. Tosi. Applying SCRUM in an OSS Development Process: An Empirical Evaluation 11th International Conference, XP 2010. 2010.

[SP27] V. Del Bianco, Lavazza, L. , Morasca, S. , Taibi, D. , and Tosi, D. , A Survey on the Importance of Some Economic Factors in the Adoption of Open Source Software. SERA. 2010, pp. 151-162.

[SP28] S. Liang and L. Ren-wang, AHP software pricing evaluation model and its application in the open source software, International Conference on Computer Application and System Modeling (ICCASM 2010). 2010. pp. V1-346-V1-352.

[SP29] V. Del Bianco, Lavazza, L. , Morasca, S. , Taibi, D. , and Tosi, D. , An Investigation of the Users' Perception of OSS Quality, in OSS, 2010, pp. 15-28.

[SP30] U. Laila, Syed Faisal Ahmed Bukhari: Open Source Software (OSS) Adoption Framework for Local Environment and its Comparison. SCSS. 2009. pp.13-16.

[SP31] T. Yamakami, Open source software adoption patterns and organizational transition stages for software vendors, International Conference on Information Sciences and Interaction Sciences (ICIS). 2010, pp. 531-535.

[SP32] L. Lavazza, Morasca, S., Taibi, D. , and Tosi, D. , Predicting OSS trustworthiness on the basis of elementary code assessment, in ESEM, 2010.

[SP33] S. Morasca, Taibi, D., and Tosi, D. , T-DOC: A Tool for the Automatic Generation of Testing Documentation for OSS Products, in OSS, 2010, pp. 200-213.

[SP34] V. Del Bianco, Lavazza, L., Morasca, S. , Taibi, D. , and Tosi, D. , The QualiSPo approach to OSS product quality evaluation. 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development. 2010, pp. 23-28.

[SP35] V. Del Bianco, Lavazza, L., Morasca, S. , and Taibi, D. A survey on Open Source Software Trustworthiness, IEEE Software. 2010, vol. 28.

[SP36] K. Mijinyawa. A theory-grounded framework of Open Source Software adoption in SMEs, European Journal of Information Systems. Vol. 20(2), 2011.

[SP37] Benlian, A. and Hess, T. Comparing the relative importance of evaluation criteria in proprietary and open-source enterprise application software selection - a conjoint study of ERP and Office systems. Information Systems Journal, 2011, vol. 21, pp. 503-525.

[SP38] L. Lavazza, Morasca, S., Taibi, D. , and Tosi, D. , OP2A: How to Improve the Quality of the Web Portal of Open Source Software Products, in Lecture Notes in Business Information Processing, Springer-Verlag, 2011.

[SP39] E. Luoma, N. Helander, L. Frank. Adoption of open source software and software-as-a-service models in the telecommunication industry ICSOB 2011

[SP40] I. Voras, B. Mihaljević and M. Orlić, Criteria for evaluation of open source cloud computing solutions, Information Technology Interfaces (ITI). 33rd International Conference on. 2011, pp. 137-142.

[SP41] G. Basilico, Lavazza, L., Morasca, S. , Taibi, D. , and Tosi, D. , OP2A: Assessing the Quality of the Portal of Open Source Software Products. International Conference on Web Information Systems and Technologies. 2011.

[SP42] L. Yan, T. Chuan-Hoo, X. Heng, and T. Hock-Hai. Open source software adoption: motivations of adopters and amotivations of non-adopters. SIGMIS Database. 2011, Vol. 42, pp. 76-94.

[SP43] W. Lee, J. K. Lee and J. Baik, Software Reliability Prediction for Open Source Software Adoption Systems Based on Early Lifecycle Measurements. 35th Annual Computer Software and Applications Conference. 2011, pp. 366-371.

[SP44] K. Gurusamy, J. Campbell. Enablers of open source software adoption: A case study of APS organisations. Australasian Journal of Information Systems. 2012, Vol 17 No 2.

[SP45] V. Del Bianco, Lavazza, L. , Lenarduzzi, V. , Morasca, S. , Taibi, D. , and Tosi, D. , A Study on OSS Marketing and Communication Strategies. IFIP International Conference on Open Source Software (OSS2012). 2012.

[SP46] Lavazza, L. , Morasca, S. , D. Taibi, and Tosi, D. , An Empirical Investigation of Perceived Reliability of Open Source Java Programs, Annual ACM Symposium on Applied Computing (SAC'12). 2012, pp. 1109-1114.

[SP47] P. S. Kochhar, T. F. Bissyandé, D. Lo and L. Jiang, An Empirical Study of Adoption of Software Testing in Open Source Projects, 13th International Conference on Quality Software, Najing. 2013, pp. 103-112.

[SP48] S. Morasca, Taibi, D. and Tosi, D. , OSS-TMM Guidelines for improving the testing process of open source software, International Journal of Open Source Software and Processes (IJOSSP), 2011.

[SP49] M.K. Mijinyawa and L. Abdulwahab. An extended framework for evaluation of open source software adoption in small businesses Research Journal of Information Technology, Vol. 6, pp. 248-269.

[SP50] M. Sarrab and O. M. Hussain Rehman. Empirical study of open source software selection for adoption, based on software quality characteristics. Adv. Eng. Softw. 2014, Vol. 69, pp. 1-11.

[SP51] J. Roy, C. Contini, F. Brodeur, N. Diouf, and W. Suryn. Method for the Evaluation of Open Source Software Quality from an IT Untrained User Perspective. International C* Conference on Computer Science and Software Engineering. 2014

[SP52] A. Siena, M. Morandini, A. Susi. Modelling risks in open source software component selection. International Conference on Conceptual Modeling Atlanta (GA). 2014

[SP53] L. Ramanathan, S. Krishnan. An empirical investigation into the adoption of open source software in Information Technology outsourcing organizations, Journal of Systems and Information Technology. Vol. 17, pp.167 - 192

[SP54] D. Taibi, An Empirical Investigation on the Motivations for the Adoption of Open Source Software, International Conference on Software Engineering Advances. 2015.

[SP55] Y. Tamura, S. Yamada Reliability Modeling and Assessment for Open Source Cloud Software: A Stochastic Approach. In Open Source Technology: Concepts, Methodologies, Tools, and Applications (pp. 1069-1090).

[SP56] H. Okamura and T. Dohi. Towards comprehensive software reliability evaluation in open source software. International Symposium on Software Reliability Engineering (ISSRE). 2015. pp. 121-129.

[SP57] D. Taibi. Can Opinion Mining Techniques Help to Select Open Source Software?, International Journal of Computer and Software Engineering. 2016. pp. 1-7.

[SP58] M. Silic and A. Back. The Influence of Risk Factors in Decision-Making Process for Open Source Software Adoption, International Journal of Information Technology and Decision Making. 2016. Vol. 15(1), pp. 151-185.

[SP59] A.I. Wasserman, X. Guo, B. McMillian, K. Qian, M. Wei and Q. Xu. OSSpal: Finding and Evaluating Open Source Software, Open Source Systems: Towards Robust Practices. 2017. pp. 193-203.

[SP60] A. Kritikos and I. Stamelos. Open Source Software Resilience Framework, Open Source Systems: Enterprise Software and Solutions. 2018. pp. 39-49.