Mechanisms to Characterize Context of Empirical Studies in Software Engineering

Bruno Cartaxo, Adauto Almeida, Emanoel Barreiros, Juliana Saraiva, Waldemar Ferreira, and Sergio Soares

Informatics Center - Federal University of Pernambuco - (CIn/UFPE)
{bfsc,ataf,efsb,jags2,wpfn,scbs}@cin.ufpe.br

Abstract. *Background:* It has become evident that empirical studies in software engineering (SE) have problems related to context characterization. This situation jeopardizes studies replication, result interpretation, knowledge transfer between academia and industry, and evidence integration of secondary studies. *Goals:* Our goals in this research are to identify and classify the mechanisms that support context characterization of empirical studies in SE. *Method:* A systematic mapping study with exhaustive coverage was conducted in accordance with the guidelines of evidence-based software engineering. *Results:* Out of 13,355 studies, 13 studies published between 1999 and 2012 were selected. Only one mechanism adopts the omnibus context approach, against 12 that follow the discrete approach. Ten studies present mechanisms to support context characterization of experiments. Only four out of the ten software engineering topics are covered by the found mechanisms. *Conclusions:* We found few mechanisms that support context characterization in SE. Besides, these mechanisms do not cover the specificities of many software engineering topics and empirical methods. Thus, we believe that more research to define mechanisms focused on these specificities is needed.

Keywords: Context, Mechanisms, Empirical Studies, Systematic Mapping Study, Software Engineering

1 Introduction

Software Engineering (SE) research presents new techniques, process, tools, and practices to increase the efficiency and quality of software systems [44]. However, some studies fail to present rigorous empirical evidences about their findings. Some authors argue that it is due to the lack of information about the context in which the studies were conducted [31, 4, 40]. On one hand, some researchers claim that the study's context is a central concept in Empirical Software Engineering (ESE) [15]. On the other hand, Sjøberg et al. say that frequently papers report context in a generic and implicit way, usually only at the external validity section [40].

The lack of contextual information is one of the main obstacles to replicate experiments [32]. Besides, some studies showed that the lack of context characterization inhibit the knowledge transfer between academia and industry [24,

4]. Additionally, a poor context characterization can cause problems to those researches that perform evidence integration through secondary studies [31]. Moreover, Basili et al. reported that context information is important to enable the creation of families of experiments [7]. Finally, when the context is explicit it is possible to conduct other studies to investigate any contextual aspect that could have interfered in the original research [34].

Once it is known the role of context in empirical studies, this work presents the results of a Systematic Mapping Study (SMS) that intends to identify and classify the mechanisms to support context characterization in SE. Moreover, we also indicate some gaps in the research area, as well as we present a summary of context characterization state-of-art. This summary can help the researchers in the finding of a mechanism that best fits their research characteristics.

This paper is organized as follows: Section 2 introduces the most important concepts. Section 3 shows the research method followed by this research. Section 4 presents the results. Section 5 summarizes the findings of this research. Section 6 depicts the threats to validity. Finally, Section 7 concludes.

2 Core Concepts

The notion of context can be used to indicate that a phenomenon, event, action or discourse needs to be studied in relationship with its environment, surrounding conditions and consequences [14]. Brézillon presented a similar definition, where context should be viewed as a set of relevant conditions and its influences that make a situation understandable and unique [8]. Additionally, Dey [13] defined context in ubiquitous computing as any information that can be used to characterize the situation of an entity.

This research focuses on context of empirical studies in SE. Thus, Dybå et al. [15] presented two types, or perspectives, of context:

Discrete: It can be understood as a set of variables that influence or moderate the relationship between the variables of interest [7, 5]. It also can be seen as a model to describe the environment in which measurements are made to evaluate software techniques [7]. Pfleeger [37] presents a similar concept, where context are state variables that influence the implementation of treatments, and indirectly affect the results of an experiment.

Omnibus: It is an approach that uses a broad perspective to structure and report context similarly to journalistic practices. The stories are described by five questions: who, what, when, where, and why, also known as the "5Ws" approach. Furthermore, Vieira et al. added the question "how" and expanded the framework to "5Ws+1H" [42].

In fact, there are various definitions and interpretations of context in different areas of knowledge. Dijk [14] mentioned that a theory on context could easily become a theory of everything. Moreover, Brézillon stated that context is heavily dependent on the investigated area, and it cannot be treated abstractly [9]. For these reasons, we define context as: *The set of information that are not the*

main interest of an empirical research (in other words, they are not the study's interventions), but they have an influence in the study's results.

Some key terms in the scope of this study have to be clearly defined, they are: mechanism, empirical method, and software engineering topic.

Mechanism is a generic term that comprises work that help the researchers to carry out their studies. In this study, we are following the mechanism classification proposed by Almeida et al. [2]: framework, guidelines, template, checklist, method, paradigm, process, technique, and lessons learned.

Empirical methods are systematic and rigorous structures to conduct scientific research, and they can be one of the following: experiment (controlled or quasi-experiments), case study, survey, ethnography, and action research [16].

Software engineering topics are the areas of interest inside SE body of knowledge, like software testing, management, among others. We are considering the ten topics presented in the SWEBOK [1], which are: software requirements, design, construction, testing, maintenance, configuration management, management, process, method and tools, and quality.

3 Research Method

The general objective of this research is to improve the quality of the empirical studies in SE by aiding their context characterization. Aiming at working toward this objective, we conducted an SMS following Kitchenham et al. guidelines [27]. This SMS addresses the following research questions:

RQ1: What are the mechanisms to support context characterization of empirical studies in SE?
RQ2: What are the types of mechanisms to characterize context of empirical studies in SE?
RQ3: What are the types of context covered by the mechanisms?
RQ4: What are the empirical methods covered by the mechanisms?
RQ5: What are the SE topics covered by the mechanisms?

The research question RQ1 is our central question. The answers of this question provide the foundations for the research as whole. The goal of RQ2 is to discover which types of mechanisms have been applied to support context characterization (such as, guidelines, frameworks, etc.). Moreover, RQ3 has the goal of classifying the found mechanisms in discrete or omnibus context specification. Furthermore, the answers of RQ4 provide some insights regarding how to characterize context of studies focusing on their empirical methods. Finally, the goal of RQ5 is to find mechanisms that present details about how to characterize context of studies focusing on their topics of interest.

The following sections describe each phase of our SMS. We adopted both manual and automatic search, and the searches were conducted on December of 2012.

3.1 Automatic Search Phase

We used the following search string structure: <mechanism and synonyms>AND <context and synonyms>AND <empirical software engineering and synonyms>. The terms and their synonyms were defined by interviewing experts and theme-related sources, as recommended by Kitchenham et al. [28]. The resulting search string was:

(mechanism OR framework OR method OR methodology OR guideline OR checklist OR process OR technique)
AND
(context OR contextual OR scope OR environment OR environmental OR settings OR circumstances OR situations OR situational OR population OR variables OR factors OR setup)
AND
("empirical software engineering" OR "experimental software engineering" OR "software engineering experimentation" OR "evidence based software engineering")

The automatic search sources were: IEEE, ACM, ScienceDirect (SD), Springer Link (SL), Scopus (Sc) and Engineering Village (EV). No restrictions about the period of studies on automatic sources were made.

3.2 Manual Search Phase

In the following sources, manual search was performed: ESEM (Empirical Software Engineering and Measurement), EASE (Evaluation and Assessment in Software Engineering), ESEJ (Empirical Software Engineering Journal), IST (Information and Software Technology). Eventually independent studies were added and the snowballing technique was applied [23]. As the automatic search, no restrictions about the venue edition were made.

3.3 Selection Phase

The **exclusion criteria** were defined based on the research objective and reviewed by an experienced researcher. Studies that matched any of the following exclusion criteria were removed:

1. The study is a slideshow or an extended abstract.
2. The study is duplicated.
3. The study is **not** related to SE.
4. The study does **not** present or only mentions a mechanism to support context characterization of empirical studies in SE.
5. Another justified reason.

In the *first round* of the study selection, we analyzed only the title and abstract. The studies that were clearly out of scope were excluded, and in case of doubt, the study remained to be analyzed on the *second round* [27]. In the *second round*, four pairs of reviewers analyzed the remaining studies. We analyzed the selected papers by screening whole of them [27]. In case of disagreement about inclusion/exclusion, a set of meetings was organized with an experienced researcher in order to resolve the conflicts.

3.4 Data Extraction Phase

Initially, the following meta-data were collected: evaluation date, title, authors, institutions, countries, source, and publication year. All selected studies were detailed read. In this phase, we excluded studies that were out of the scope of this research, but they were not excluded in the *second round*. The remaining studies correspond to the Selected Primary Studies (SPS) of this research.

4 Results

This section presents the results of our SMS. First, Section 4.1 shows some data about the selection process. Then, Section 4.2 presents some general information about the SPS. The remaining sections present our results regarding each research question (respectively, Section 4.3, 4.4, 4.5, 4.6, and 4.7).

4.1 Selection Procedure Results

Initially, 13,355 studies were collected, 9,405 from automatic search and 3,950 from manual search. After the *first round* of the selection phase only 527 studies remained. After the *second round*, this value was reduced to 60 studies. Additionally, during the data extraction phase more 47 studies were excluded, remaining the 13 selected studies. The high amount of excluded studies was expected, since the terms in search string are commonly found in various studies related to SE. That was a conscientious choice, since we decided to conduct this SMS with an exhaustive coverage [11]. The entire selection phase is summarized on Table 1, where "Ind." means independent sources, "snow." means the snowballing technique, "1#" means the *first round* of selection phase, "2#" means the *second round*, the numbers below the "excluded" cell represent each exclusion criterion as shown in Section 3.3, "incl." means the included studies at the selection phase, and "sel." means the final set of selected studies.

We calculated the agreement index and Kappa value of our criteria application process. Both agreement index and Kappa statistics (K) were calculated in order to evaluate the consistency of the application of the exclusion criteria made by the reviewers in the *second round*. Kitchenham and Charters [27] and Edwards et al. [17] recommend this calculation. For the agreement, we got a value of 91.2% of agreement between the reviewers. However, our Kappa value was K=0.5873544 which means a moderate level of agreement. After a deeper investigation, it was possible to observe that this was caused by an anomaly already described in the statistical literature as a paradoxical result that produces low values of Kappa even with high levels of agreement [18]. This paradox occurs in presence of rare findings [43] or categories with prevalent proportion of individuals [3], such as the case of this research where the "inclusion phenomenon" was rare and the "exclusion phenomenon" was prevalent. Therefore, we believe the exclusion criteria are consistent, since we obtained 91.2% of agreement. Besides, Vieira and Garrett [43] reported a similar problem with the Kappa test.

Table 1. Summary of SPS process.

Source	Initial	1#	2#						Selected
			Excluded					Included	
			Crit. 1	Crit. 2	Crit. 3	Crit. 4	Crit. 5		
IEEE	312	96	5	0	0	80	0	11	5
ACM	1,744	168	12	1	0	132	6	17	2
SD	996	51	1	1	1	43	0	5	1
SL	2,061	51	14	1	0	31	0	5	1
EV	292	34	10	0	0	19	0	5	1
Sc	4,000	86	2	1	0	69	1	13	0
ESEM	373	1	0	0	0	1	0	0	0
EASE	137	6	0	0	0	6	0	0	0
ESEJ	431	3	2	0	0	0	1	0	0
IST	3,009	31	2	0	0	25	0	4	0
Ind.	-	-	-	-	-	-	-	-	3
Snow.	-	-	-	-	-	-	-	-	0
TOTAL	13,355	527	48	4	1	406	8	60	13

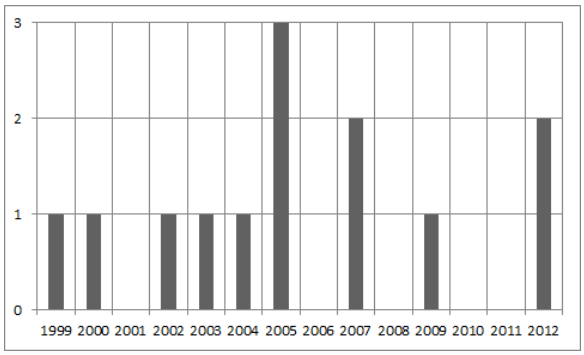

Fig. 1. Temporal Distribution of the selected studies.

4.2 General Information of the Studies

Figure 1 presents the temporal distribution of the studies related to context characterization. The first study dates from 1999 and in mean one new study related to context characterization is published by year.

Other relevant data is that 40 researchers are involved in studies related to context characterization. Claes Wohlin has the higher number of published studies (three). After him, ten researchers are authors of two studies each, and 29 others published only one. Moreover, 24 institutions are involved in research on this subject, and seven of them published two studies each, while 17 have only one publication. The complete list of authors and institutions are not available due to space limitations.

4.3 Answering Research Question 1

RQ1: What are the mechanisms to support context characterization of empirical studies in SE?

Table 7 present all the 13 mechanism found by our SMS. The first column presents the mechanism ID. Throughout this article, we refer the mechanisms based on this ID. The second column is a small description of the mechanisms.

4.4 Answering Research Question 2

RQ2: What are the types of mechanisms to support context characterization of empirical studies in SE?

Table 2 presents found mechanisms grouped by their mechanism types. It is possible to observe that framework is the most common mechanism, corresponding to seven of 13 found mechanisms. We believe this result is related to the high amount of mechanisms adopting the discrete context approach, once frameworks usually present a suite of contextual variables. While guidelines, methods, and paradigms present a detailed explanation about the context characterization [2].

Table 2. Mechanisms Grouped by Mechanism Types.

Mechanism	**Studies**	#
Framework	SPS01, SPS02, SPS04, SPS05, SPS06, SPS11, SPS12	7
Guidelines	SPS04, SPS10	2
Template	SPS07, SPS13	2
Checklist	SPS03, SPS04	2
Method	SPS09	1
Paradigm	SPS08	1

4.5 Answering Research Question 3

RQ3: What types of context are covered by the mechanisms?

We are categorizing the context types as discrete or omnibus (Section 2) as illustrated in Table 3. It is possible to observe that the discrete perspective is prevalent, since it is present in 12 of the 13 selected studies. Therefore, from this result we extrapolate conjectures related to SE community preferences, namely empirical method adoption and dominant philosophical stance. Thus, the strong focus on discrete context approaches could be related to the high adoption of quantitative methods, chiefly experiments, as the main empirical method in SE [21, 35]. Besides that, this fact brings with it the positivist philosophical stance as well as the comprehension of the world's phenomena by a reductionist point of view [33, 16].

Table 3. Distribution of mechanisms by context type.

Type	Studies	#
Discrete	SPS01, SPS02, SPS03, SPS04, SPS05, SPS06, SPS07, SPS09, SPS10, SPS11, SPS12, SPS13	12
Omnibus	SPS08	1

4.6 Answering Research Question 4

RQ4: What are the empirical methods covered by the mechanisms?

We classified the empirical methods as proposed by Easterbrook et al. [16] (details in Section 2). For instance, SPS07 is a template (mechanism type) to characterize context of experiments (empirical method). Table 4 presents the found mechanisms grouped by their empirical methods. We can note that experiment is the most mentioned empirical method. This prevalence is understandable since ESE community has applied more experiments than other empirical methods, as was already noticed by other researchers [35, 38]. Moreover, we can notice that sum of the quantity of mechanism by research method (Table 4 column 3) does not correspond to the quantity of found mechanisms. This is because some mechanism can be used by more then one empirical method. For instance, SPS08 can be used in case study, ethnography, and action research.

Table 4. Distribution of mechanisms by empirical method.

Method	Studies	#
Experiment	SPS01, SPS02, SPS04, SPS05, SPS07, SPS09, SPS10, SPS11, SPS12, SPS13	10
Case Study	SPS01, SPS03, SPS08, SPS10, SPS11, SPS12	6
Ethnography	SPS08	1
Action Research	SPS08	1

4.7 Answering Research Question 5

RQ5: What are the SE topics covered by the mechanisms?

We adopted the topics on the chapters of SWEBOK [1] (as presented in Section 2). Table 5 shows which topic is most suitable for each found mechanism. We can observe that only six selected studies presented a mechanism related to the topics aforementioned. This is an evidence of the lack of mechanisms to support context characterization of empirical studies focused on specific topics of SE. Consequently, researchers probably have been forced to adopt generic mechanisms to characterize context of their empirical studies.

Table 5. Distribution of mechanisms by SE Topic.

SE Topic	**Studies**	#
Software Construction	SPS01, SPS11	2
Software Process	SPS06, SPS12	2
Software Maintenance	SPS04	1
Software Engineering Management	SPS05	1

5 Summarizing Findings

In this section, we summarize all the data provided by answering our research questions. Table 6 presents this summary. It can help the researchers that intend to characterize context in their empirical studies. Once the researchers had defined the domain of their research topic and the empirical method, they can consult Table 6 in order to identify the most suitable mechanism. For instance, a researcher has to perform an experiment related to software evolution. Consulting this table, we can see that there are 11 mechanisms available to characterize context in experiments. According to the SWEBOK [1], the SE topic corresponding to software evolution is Software Maintenance. Therefore, SPS04 is the mechanism most suitable in this scenario, however SPS02, SPS07, SPS09, SPS10, SPS13, and SPS06 can also be adopted.

Table 6. Context Characterization Mechanism Summary.

Method	**SE Topic**	**Studies**
Experiment	Construction	SPS01; SPS11
	Process	SPS12
	Maintenance	SPS04
	Management	SPS05
	-	SPS02; SPS07; SPS09; SPS10; SPS13; SPS06
Case Study	Construction	SPS01; SPS11
	Process	SPS12
	-	SPS03; SPS08; SPS10
Ethnography	-	SPS08
Action Research	-	SPS08

Analyzing Table 6, on the one hand we can see that mostly the mechanisms are related to experiment or case study. On the other, there is no mechanism specific to many SE topics, such as, software requirements, design, testing, configuration management, management, method and tools, and quality. Moreover, there is only one mechanism suitable to ethnography and action research. Besides, none mechanism can be used by surveys.

Table 7. Mechanisms that Support Context Characterization.

Study	Study's Description
SPS01	**An Initial Framework for Research on Pair Programming [19]** — This study presents a framework that enumerates a set of useful variables for researches that aim to investigate phenomena related to pair programming through experiments. Furthermore, it provides a rich description of each of the listed variables as well as references of empirical studies that have used the described variables. This helps to understand the effects of each variable and increase the scientific credibility of the framework.
SPS02	**Building Knowledge through Families of Experiments [7]** — This study presents a discussion about the experience and motivation of subjects of experiments in SE. Moreover, exemplifies the implications and influences of these two variables on empirical studies. That helps researchers who want to characterize and understand the interactions related to experience and motivation.
SPS03	**Context in industrial software engineering research [36]** — This study presents a list of variables grouped into categories for characterize context of empirical studies focusing on industry research. The categories comprising variables related to organization and market are particularly interesting, once these aspects are often neglected in purely academic studies.
SPS04	**Empirical studies in reverse engineering: state of the art and future trends [41]** — This study presents a framework to guide research on reverse engineering listing sets of contextual variables that potentially influence experiments about this theme.
SPS05	**Experimental Context Classification: Incentives and Experience of Subjects [22]** — This study presents a classification based on the motivation and experience of experiments' subjects. The definition and discussion of classes of incentives that impact on subjects' motivation contribute to a researcher who wishes to classify and understand the motivational aspects that may impact an empirical study.
SPS06	**The situational factors that affect the software development process: Towards a comprehensive reference framework [10]** — This study presents a long and rich list of variables that affect the software development process, relying on rigorous qualitative methods to define such list of variables. If in one hand has no guides and in-depth discussions about each variable, on the other, strongly contributes with an initial catalog of variables that influence phenomena related to software development process.
SPS07	**Using Context Distance Measurement to Analyze Results across Studies [12]** — This study presents a template to collect contextual information of experiments in order to compare the context of different studies.
SPS08	**What Works for Whom, Where, When, and Why? On the Role of Context in Empirical Software Engineering [15]** — This study presents an approach to think and characterize context of empirical studies that apply qualitative methods. Replaces the notion of variables for five questions that encourage the characterization of context by means of journalistic techniques. Additionally, shows a rich discussion about the inherent problems with the traditional approach of context characterization by means of discrete variables.
SPS09	**Collecting Feedback During Software Engineering Experiments [25]** — This study presents an approach to collect contextual information about events that may occur during the execution of experiments and can impact their results. Besides that, contributes listing variables that are usually neglected, such as interruptions in activities during the execution of the experiment as well as stress level and various other information related to the participant's perception regarding the experiment itself.
SPS10	**Preliminary Guidelines for Empirical Research in Software Engineering [29]** — This study presents a discussion about context of empirical studies focusing more on guide by example how to characterize context than listing variables.
SPS11	**Extreme Programming Evaluation Framework for Object-Oriented Languages Version 1.4 [45]** — This study presents a framework encompassing a wide range of contextual variables categorized and described in detail. The main goal is to assist the process of characterization and evaluating of empirical studies focused on Extreme Programming (XP). Some of the variables definitions do not only show rich descriptions but also exhibit measurement scales.
SPS12	**Rational Unified Process Evaluation Framework Version 1.0 [30]** — This study presents the same framework of the SPS11, adapted for the characterization and evaluation of empirical studies focusing on Rational Unified Process (RUP).
SPS13	**Experimentation in Software Engineering: An Introduction [46]** — This study presents a brief discussion on the importance of characterizing some contextual variables of experiments as well as their impact.

6 Threats to Validity

This section shows some possible threats to validity of this study, and how each of them was mitigated. According to Sjøberg et al. [39], the main threats to validity that can occur in this kind of studies are: (1) publication bias, (2) inaccuracy in data extraction, and (3) classification errors. Moreover, Kitchenham [26] presented the following potential threats: (4) rely on just one search engine, (5) select studies only of a specific period of time and (6) conduct the selection procedure by only one person. Barreiros [6] also states that (7) the automated search strategy can fail to consider recent studies that have not been indexed yet.

The following actions were taken to try to mitigate the outlined threats: (1) the two major conferences (EASE and ESEM) and the two leading journals (ESEJ and IST) about Empirical Software Engineering (ESE) were considered; (2) a structured form was defined to extract relevant information from the studies; (3) the study classification was checked by another researcher; (4) the automatic search process was conducted on two different search engines, and four digital libraries; (5) no restriction was made for the period of publications; (6) the studies selection procedure was done in pairs, and conflict meetings were held; (7) the studies from automatic sources were complemented by studies obtained from manual sources through a systematic process, in order to include recent studies that possibly were not indexed by the search engines and digital libraries when the automatic search was performed.

7 Concluding Remarks

There are many claims about the need for more and better context characterization of empirical studies in SE [24, 36, 40, 15]. This research showed that there are few studies that support context characterization in SE. In our SMS, we analyzed 13,355 primary studies, and only 13 were selected. Among our results, we highlight: (i) Seven out of 13 studies presented are frameworks, (ii) only one mechanism adopts the omnibus context approach, against 12 which use the discrete approach, (iii) ten out of 13 selected studies show mechanisms to support context characterization in experiments, and (iv) only four of the ten SE topics are covered by the found mechanisms. Finally, we summarized all data collected in order to aid the researchers to choose a mechanism that supports the context characterization in their empirical studies.

As future work we plan to investigate in depth the context characterization in each specific empirical method, and confronting this new information with the one obtained by the current study. Furthermore, a theory about context of empirical studies in SE could be built based on the knowledge acquired from the combination of these researches. Towards a more dynamic context description, as proposed by Dijk [14], we plan to define models and ontologies for empirical studies based on context, like in Garcia et al. [20].

References

1. Abran, A., et al.: Guide to the Software Engineering Body of Knowledge - SWEBOK. IEEE Press, Piscataway, NJ, USA, 2004 version edn. (2004), http://www.swebok.org/ironman/pdf/SWEBOK_Guide_2004.pdf
2. Almeida, A.: Um Mapeamento Sistemático de Mecanismos para Guiar Estudos Empíricos em Engenharia de Software. Master's thesis, Federal University of Pernambuco (March 2011)
3. Altman, D.: Practical Statistics for Medical Research. Chapman & Hall/CRC Texts in Statistical Science, Taylor & Francis (1990), http://books.google.com.br/books?id=v-walRnRxWQC
4. Babar, M.A., Zhang, H.: Systematic literature reviews in software engineering: Preliminary results from interviews with researchers. pp. 346–355. ESEM '09, IEEE Computer Society, Washington, DC, USA (2009), http://dx.doi.org/10.1109/ESEM.2009.5314235
5. Baron, R.M., Kenny, D.A.: The moderator–mediator variable distinction in social psychological research: Conceptual, strategic, and statistical considerations. Journal of personality and social psychology 51(6), 1173 (1986)
6. Barreiros, E.F.S.: A Systematic Mapping Study on Software Engineering Testbeds. Master's thesis, Federal University of Pernambuco (February 2011)
7. Basili, V., et al.: Building knowledge through families of experiments. IEEE Transactions on Software Engineering 25(4), 456–473 (1999)
8. Brézillon, P.: Context in artificial intelligence: I. a survey of the literature. Computers and Artificial Intelligence 18(4) (1999)
9. Brezillon, P.: Context modeling: Task model and practice model. pp. 122–135. CONTEXT'07, Springer-Verlag, Berlin, Heidelberg (2007), http://dl.acm.org/citation.cfm?id=1770806.1770816
10. Clarke, P., O'Connor, R.V.: The situational factors that affect the software development process: Towards a comprehensive reference framework. IST 54(5), 433–447 (May 2012), http://dx.doi.org/10.1016/j.infsof.2011.12.003
11. Cooper, H.: Organizing knowledge syntheses: A taxonomy of literature reviews. Knowledge in Society 1(1), 104–126 (1988), http://dx.doi.org/10.1007/BF03177550
12. Cruzes, D., et al.: Using context distance measurement to analyze results across studies. In: ESEM. pp. 235–244 (2007)
13. Dey, A.K.: Understanding and using context. Personal Ubiquitous Comput. 5(1), 4–7 (Jan 2001), http://dx.doi.org/10.1007/s007790170019
14. Dijk, T.v.: Discourse and Context: A Sociocognitive Approach. Cambridge University Press (2010), http://books.google.com.br/books?id=P3N9QgAACAAJ
15. Dybå, T., et al.: What works for whom, where, when, and why?: On the role of context in empirical software engineering. pp. 19–28. ESEM '12, ACM, New York, NY, USA (2012), http://doi.acm.org/10.1145/2372251.2372256
16. Easterbrook, S., et al.: Selecting empirical methods for software engineering research. In: Shull, F., et al. (eds.) Guide to Advanced Empirical Software Engineering, pp. 285–311. Springer London (2008), http://dx.doi.org/10.1007/978-1-84800-044-5_11
17. Edwards, P., et al.: Identification of randomized controlled trials in systematic reviews: accuracy and reliability of screening records. Statistics in Medicine 21(11), 1635–1640 (2002), http://dx.doi.org/10.1002/sim.1190

18. Feinstein, A.R., Cicchetti, D.V.: High agreement but low kappa: I. the problems of two paradoxes. Journal of Clinical Epidemiology 43(6), 543 – 549 (1990), http://www.sciencedirect.com/science/article/pii/089543569090158L
19. Gallis, H., et al.: An initial framework for research on pair programming. pp. 132–. ISESE '03, IEEE Computer Society, Washington, DC, USA (2003), http://dl.acm.org/citation.cfm?id=942801.943643
20. Garcia, R.E., et al.: An ontology for controlled experiments on software engineering. In: SEKE. pp. 685–690. Knowledge Systems Institute Graduate School (2008), http://dblp.uni-trier.de/db/conf/seke/seke2008.html#GarciaHBM08
21. Glass, R., et al.: Research in software engineering: an analysis of the literature. IST 44(8), 491 – 506 (2002), http://www.sciencedirect.com/science/article/pii/S0950584902000496
22. Höst, M., et al.: Experimental context classification: Incentives and experience of subjects. pp. 470–478. ICSE '05, ACM, New York, NY, USA (2005), http://doi.acm.org/10.1145/1062455.1062539
23. Jalali, S., Wohlin, C.: Systematic literature studies: Database searches vs. backward snowballing. pp. 29–38. ESEM '12, ACM, New York, NY, USA (2012), http://doi.acm.org/10.1145/2372251.2372257
24. Jedlitschka, A., Ciolkowski, M.: Towards evidence in software engineering. In: ISESE. pp. 261–270 (2004)
25. Karahasanoviæ, A., et al.: Collecting feedback during software engineering experiments. ESEJ 10(2), 113–147 (Apr 2005), http://dx.doi.org/10.1007/s10664-004-6189-4
26. Kitchenham, B.: What's up with software metrics? - a preliminary mapping study. JSS 83(1), 37–51 (Jan 2010), http://dx.doi.org/10.1016/j.jss.2009.06.041
27. Kitchenham, B., Charters, S.: Guidelines for performing systematic literature reviews in software engineering. Tech. Rep. EBSE 2007-001, Keele University and Durham University Joint Report (2007), http://www.dur.ac.uk/ebse/resources/Systematic-reviews-5-8.pdf
28. Kitchenham, B., et al.: Mapping study completeness and reliability - a case study. In: EASE. pp. 126–135 (2012)
29. Kitchenham, B.A., et al.: Preliminary guidelines for empirical research in software engineering. IEEE Trans. Softw. Eng. 28(8), 721–734 (Aug 2002), http://dx.doi.org/10.1109/TSE.2002.1027796
30. Krebs, W., et al.: Rational unified process evaluation framework version 1.0. Tech. rep. (2005), ftp://ftp.ncsu.edu/pub/unity/lockers/ftp/csc_anon/tech/2005/TR-2005-46.pdf
31. Lopes, E.C., et al.: A conceptual framework for the development of applications centred on context and evidence-based practice. In: Filipe, J., Cordeiro, J. (eds.) ICEIS (3). pp. 60–69. SciTePress (2010), http://dblp.uni-trier.de/db/conf/iceis/iceis2010-3.html#LopesSVS10
32. Lung, J., et al.: On the difficulty of replicating human subjects studies in software engineering. pp. 191–200. ICSE '08, ACM, New York, NY, USA (2008), http://doi.acm.org/10.1145/1368088.1368115
33. McNeill, P., Chapman, S.: Research Methods. Routledge (2005), http://books.google.com.br/books?id=3iAjarmi0ykC
34. Montesi, M., Lago, P.: Software engineering article types: An analysis of the literature. JSS 81(10), 1694–1714 (Oct 2008), http://dx.doi.org/10.1016/j.jss.2007.11.723
35. O'Brien, M., et al.: Empirically studying software practitioners - bridging the gap between theory and practice. In: ICSM. pp. 433–442 (2005)

36. Petersen, K., Wohlin, C.: Context in industrial software engineering research. In: ESEM. pp. 401–404 (2009)
37. Pfleeger, S.L.: Experimental design and analysis in software engineering. Annals of Software Engineering 1(1), 219–253 (1995), http://dx.doi.org/10.1007/BF02249052
38. Segal, J.: The nature of evidence in empirical software engineering. In: Software Technology and Engineering Practice, 2003. Eleventh Annual International Workshop on. pp. 40–47 (2003)
39. Sjøberg, D.I.K., et al.: A survey of controlled experiments in software engineering. IEEE Trans. Softw. Eng. 31(9), 733–753 (Sep 2005), http://dx.doi.org/10.1109/TSE.2005.97
40. Sjøberg, D.I.K., et al.: The future of empirical methods in software engineering research. pp. 358–378. FOSE '07, IEEE Computer Society, Washington, DC, USA (2007), http://dx.doi.org/10.1109/FOSE.2007.30
41. Tonella, P., et al.: Empirical studies in reverse engineering: State of the art and future trends. ESEJ 12(5), 551–571 (Oct 2007), http://dx.doi.org/10.1007/s10664-007-9037-5
42. Vieira, V., et al.: A context-oriented model for domain-independent context management. Revue d'Intelligence Artificielle 22(5), 609–627 (2008), http://dblp.uni-trier.de/db/journals/ria/ria22.html#VieiraBST08
43. Viera, A.J., Garrett, J.M.: Understanding interobserver agreement: The kappa statistic. Journal of the Society of Teachers of Family Medicine 37(5) (2005), http://www.stfm.org/fmhub/fm2005/May/Anthony360.pdf
44. Weyuker, E.J.: Empirical software engineering research - the good, the bad, the ugly. pp. 1–9. ESEM '11, IEEE Computer Society, Washington, DC, USA (2011), http://dx.doi.org/10.1109/ESEM.2011.66
45. Williams, L., Krebs, W., Layman, L.: Extreme Programming Evaluation Framework for Object-Oriented Languages Version 1.4. Tech. rep., NCSU (Jun 2004), ftp://ftp.ncsu.edu/pub/unity/lockers/ftp/csc_anon/tech/2004/TR-2004-18.pdf
46. Wohlin, C., et al.: Experimentation in Software Engineering: An Introduction. Kluwer Academic Publishers, Norwell, MA, USA (2000)