Six Years of Systematic Literature Reviews in Software Engineering: an Extended Tertiary Study

Fabio Q. B. da Silva, André L. M. Santos,
Sérgio Soares, A. César C. França, Cleviton V. F. Monteiro
Center ofInformatics – UFPE
Cidade Universitária – 50.540-740, Recife – PE – Brazil
{fabio,alms,scbs,accf2,cvfm}@cin.ufpe.br

ABSTRACT

After the introduction of evidence-based software engineering in 2004, systematic literature reviews (SLR) have been increasingly used as a method for conducting secondary studies in software engineering. Our goal is to analyze quality, coverage of software engineering topics, and potential impact of published SLRs, extending and updating two previous similar studies. We searched for SLRs published between July 2008 and December 2009, analyzed the relevant studies, compared and integrated our findings with previous studies. We found 67 SLRs, addressing 24 software engineering topics. The number of SLRs in software engineering is increasing and the overall quality of the studies is improving, suggesting that the software engineering research community is starting to consistently adopt SLRs as a research method. However, the majority of the SLRs did not evaluate the quality of primary studies and fail to provide guidelines to practitioners, thus decreasing their potential impact to influence software engineering practice.

Categories and Subject Descriptors

D.2.0 [**Software**]: Software Engineering – General;

General Terms

Experimentation

Keywords

Systematic reviews, Mapping Studies, Software Engineering, Tertiary Studies.

1. INTRODUCTION

In 2004, Kitchenham et al. [11] introduced the concept of evidence-based software engineering (EBSE) as a promising approach to integrate academic research and industrial practice in software engineering. Following this paper, Dybå et al.[7]presented EBSE from the point of view of the software engineering practitioner, and Jørgensen et al. [17] complemented with an account of the aspects of teaching EBSE to university students.

By analogy with Evidence-Based Medicine [21], five steps are needed to practice EBSE:

1. to convert the need for information (about the practice of software engineering) into answerable questions;
2. to track down, with maximum efficiency, the bestevidence with which to answer the questions;
3. to appraise the evidence critically, in order to assess its validity (closeness to the truth) and usefulness (practical applicability);
4. to implement the results of this appraisal in software engineering practice;
5. to evaluate the performance of this implementation.

The preferred method to implement steps 2 and 3 are systematic literature reviews (SLR). Kitchenham[12] adapted guidelines for performing literature reviews in medicine for SLRs in software engineering. Later, using concepts from social science [19], Kitchenham and Charters updated the guidelines [13]. The literature differentiates several types of systematic reviews [19], including:

- Conventional SLRs[19], which aggregate results about effectiveness of a treatment, intervention, or technology, and are related to specific research questions like: *Is intervention I on population P more effective in obtaining outcome O in context C than comparison treatment C?* (resulting in the PICOC structure [19]) When enough quantitative experiments are available to answer the research question, meta-analysis (MA) can be used to integrate effect results.
- Mapping (or scoping) Studies (MS) [1] aim to identify all research related to a specific topic, i.e. to answer broader questions related to trends in research. Typical questions are exploratory: *What do we know about topic T?*

Greenhalgh[8] emphasizes that evidence-based practice is not only about reading papers and summarizing their results in a comprehensive and unbiased way. It involves reading the *right* paper (valid and useful) and then to change our behavior in the practice of our discipline (software engineering, in our case). Therefore, EBSE is not only about performing good quality SLRs and making them publicly available (steps 2 and 3). All five steps should be performed for a practice to be considered evidence-based. Nevertheless, SLRs can play an important role in supporting research and informing practice about the impact or effect of technology. Therefore, information about how many SLRs are available in software engineering, where they can be found, which topic areas have been addressed, and the overall quality of available studies can greatly benefit the academic community as well as the practitioners.

In this article we perform a mapping study of SLRs in software engineering, published between July 2008 and December 2009. Our goal is to analyze available secondary studies, and integrate our findings with the results from two previous studies discussed in Section 2. Our work is classified as a tertiary study since we are performing a review of secondary studies.The study's protocol is presented in Section 3. Sections 4 and 5 discuss extracted data and analysis. Finally, the conclusions are presented in Section 6.

Conference'10, Month 1–2, 2010, City, State, Country.

Submitted to the IEEE/ACM
International conference on Software Engineering, ICSE'2010 – Please, do not distribute.

2. PREVIOUS STUDIES

Two previous tertiary studies have been performed with the goal of assessing the use of SLRs in software engineering research and, indirectly, to investigate the adoption of EBSE by the software engineering researches.

The first study developed by Kitchenham et al. [15] (called hereafter the Original Study – OS) found 20 unique studies reporting literature reviews that were considered systematic according to the authors.The study deployed a manual search of specific conference proceedings and journal paperslooking for peer-reviewed articles published between January 1st 2004 and June 30th 2007. The OS identified several problems or limitations of the existing SLRs:

- A relatively large number of studies (40%, 8/20) were investigating research methods or trends rather than technique evaluation, which should be the focus of a (conventional) systematic review [19].
- The spread of software engineering topics was limited. The majority of the SLRsconcerned with technical issues rather than research methods concentrated on cost estimation (58%, 7/12).
- The number of primary studies was much larger for mapping studies than forSLRs.
- Relatively few SLRs assessed the quality of primary studies.
- Relatively few papers provided advice oriented to practitioners.

The last two problems identified above are more concerning, since the actual purpose of evidence-based practice is to inform and advise practitioners about (good quality) empirical evidence that can be used to improve their practice. We shall investigate if these problems persist.

One limitation of the OS was that the search was manual and performed on a relevant but restricted set of sources. Therefore, relevant studies might have been missed, as turned out to be the case according to the findings of the second tertiary study performed byKitchenham et al. [16]. This study (hereafter called the First Extension Study – FE) deployed an automatic search on five search engines and indexing systems, and found 33 additional unique studies published between January 1st 2004 and June 30th 2008. The FE identified some improvement on the issues found in the OS: the number of SLRs was increasing as wellas the overall quality of the studies. However, still only few SLRsfollowed specific methodology, included practitioner guidelines or evaluated the quality of primary studies. The authors also emphasize that only a very small number of SLRswereauthored by researchers from the USA. Since the USA is the leading country in software engineering research, this could be interpreted as a sign of limitation on the adoption of evidence-based software engineering or that this adoption is mainly concentrated in European research groups.

3. METHOD

The research group that developed the OS and FE reviews, intended to repeat their study at the end of 2009 to "track the progress of SLRs and evidence-based software engineering" [15]. During the EASE Conference in May 2010, at Keele University, we discussed this extension with two members of the group, namely Pearl Brereton and David Budgen, and they said that the extension had not been performed yet. We manifested our intention of performing this extension and to integrate the results with the OS and FE findings. At that meeting, two methodological decisions were made. First, our extension would be performed independently of their work and with as little exchange of information as possible. Second, that we would use, as close as possible, the same protocol used in the FE. These decisions would assure that one study would not influence or bias the other one, and we would be able to compare the results, since the protocol would bethe same. Therefore, the method used in our study follows closely the protocol defined by Kitchenham et al. [14] and the structure of the presentation of [16].

Hereafter, we shall refer to the Original Study as OS, to the First Extension as FE, and to our study, the Second Extension, as SE. We shall use OS/FE to refer to the combination of the results related to the 53 SLRs from the OS (20) and FE (33). Finally, we use OS/FE+SE to refer to the combination of results of OS/FE with our findings, 67 SLRs, representing a total of 120 secondary studies.

3.1 Research Questions

The five research questions investigated in the SE were equivalent to theresearch questions used in the FE[16]. We performed minor adjustments and add sub-questions as follows.

RQ1: How many SLRs were published between 1st January 2004 and 31stDecember 2009?

RQ1.1: How many SLRs were published between 1st January 2004 and 30th June 2008?

RQ1.2: How many SLRs were published between 1stJuly 2008and 31stDecember 2009?

The sub-questions of RQ1 investigate the development of SLRs in two separate periods. To answer RQ1.1 we will use the results of OS/FE [15][16]. As for RQ1.2, we will perform the processes of search, selection, quality assessment, and data extraction defined in Sections 3.3-3.5. Similarly, we will address the next questions considering the two time periods as we explicitly did for RQ1, searching for new evidence, combining with the results of the previous studies, and integrating all findings.

RQ2: What research topics are being addressed?

RQ3: Which individuals and organizations are most active in SLR-based research?

The fourth question in the FE was "Are the limitations of SLRs, as observed in the originalstudy, still an issue?". We changed the fourth question to:

RQ4: Are the limitations of SLRs, as observed in the two previous studies, FE and OS, still an issue?

And we kept the fifth question unaltered:

RQ5: Is the quality of SLRs improving?

3.2 Research Team

This study was developed by a team of five researchers, who co-authored this article. Three of them, Fabio Silva, André Santos, and Sérgio Soares (referred to as R_1, R_2, and R_3) are full-time lectures. ClevitonMonteiro and César França (R_4 and R_5) are PhD students. All researchers are affiliated to the Center of Informatics, Federal University of Pernambuco, Brazil.

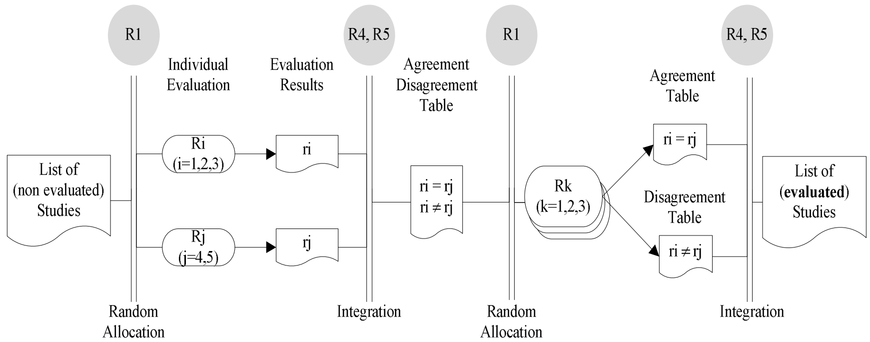

Figure 1: The Decision and Consensus Procedure (DCP)

3.3 Decision Procedure

Three important activities in a systematic review require decisions about possibly conflicting situations: study selection, quality evaluation, and data extraction. It is recommended that such activities to be performed by at least two researchers. Therefore, a process to support decision-making and consensus reaching is necessary. For this study, we defined a decision and consensus procedure (DCP) shown in Figure 1.

The procedure starts with a list of non-evaluated studies as the input for a decision process (study selection, quality assessment, or data extraction), which R_1 randomly allocates to two researchers ($R_i$and R_j). After individual evaluation, results ($r_i$and r_j) are integrated by R_4 and R_5 into an Agreement/Disagreement Table (ADT). Then, R_1 randomly allocates the results from the ADT to the researchers, making sure that a different researcher (R_k) will evaluate the results. The students do not participate in this stage. $R_k$judges the disagreements in the ADT and producesone of three results: an agreement for one of the previous decisions, a third result (r_k), or keep the original disagreement. The remaining disagreements are resolved in a consensus meeting with all five researchers. After consensus is reached, all results are integrated by R_4 and R_5 into the final list of evaluated studies.

3.4 Search Process

We perform our search looking for peer-reviewed articles published between 1st July 2008 and 31st December 2009. Differently from OS and FE, we combined automatic and manual searchto increase coverage.The Automatic searchwas performed by R_4 and R_5on 6 search engines and indexing systems: ACM Digital Library, IEEE Xplore Digital Library, Science Direct, CiteSeerX, ISI Web Of Science, and Scopus. All searches were performed on the entire paper, including title and abstract, except for the ISI Web of Science, wherethe search was based only on title and topic due to limitations imposed by the search engine. This is the string used in the automatic search:

("Software engineering") AND ("review of studies" OR "structured review" OR "systematic review" OR "literature review" OR "literature analysis" OR "in-depth survey" OR "literature survey" OR "meta analysis" OR "past studies" OR "subject matter expert" OR "analysis of research" OR "empirical body of knowledge" OR "overview of existing research" OR "body of published research" OR "Evidence-based" OR "evidence based" OR "study synthesis" OR "study aggregation")

The syntax was the same for all engines, except for the ISI Web of Science, which required minor syntax changes due to the characteristics of the engine. The semantics of the strings remainedunchanged. The search process was validated against the papers found in the OS and FE studies. Only three papers were not found by our search process. The study by Barcelos and Travassos[2] was obtained in the OS by directly consulting the authors. We searched the 6 engines looking for the article title directly and still did not find the study. The same happened with the study by Petersson et al. [18]. The study by Shaw and Clements [22] is indexed by the ACM digital library but the authors use the term survey instead of review, and our search failed to find this article. Overall, we missed only one indexed article in a total of 51, thus we concluded that our search process was robust.The automatic search on the 6 engines returned 1,389 documents. A first filter was applied by reading title and abstract to remove obviously irrelevant papers. This filter resulted in 157 papers.

The lecturers (R_1, R_2, and R_3) performed a Manual Searchon relevant journals and conference proceedings(Table 1). The researchers looked for title and abstract of all published articles in each source. This search happened in parallel with the automatic search and produced 66 potentially relevant articles.

Table 1 - Manual Search Sources

ACM Computer Surveys
ACM Transactions on Software Engineering Methodologies
Communications of the ACM
Empirical Software Engineering Journal
Evaluation and Assessment of Software Engineering
IEE Proceedings Software (now IET Software)
IEEE Software, Software Practice and Experience
IEEE Transactions on Software Engineering
Information and Software Technology
Information and Software Technology
Int. Conference on Software Engineering
Int. Symposium on Empirical Software Engineering and Measurement
Journal of Systems and Software

The lists from the automatic and manual search were merged and duplicates removed. The final list of potentially relevant studies contained 154 unique papers. This list was the input to the study selection activity.

3.5 Study Selection

Study selection was performed by fully reading 154 the potentially relevant articlesselected during the search process and excluding those articles that were not a SLR, i.e. literature review with defined research questions, search process, data extraction and data presentation, or were a SLR related to Information Systems, HCI or other Computer Science topics that were clearly not Software Engineering. When an SLR has been published in more than one journal or conference, both versions of the study were reviewed for purposes of data extraction, but only one (the first published, to be consistent with OS/FE) was accounted for in the final statistics. Study selection was performed following the decision and consensus procedure describe in Section3.3.

After finishing the study selection, we performed a manual search on the reference list of each selected study and found two new studies, [SE76] and [SE77]. The former was not found in the previous searches because the EASE Conference 2008 happened in June 26th and 27th, being out of the time period of our study. However, we decided to include the study since it has also been missed in the OS/FE. The latter was not found by the initial automatic search of IEEE Xplore. We tried the search again looking specifically for the title of the paper, and could not found it. In this case, the manual search of the references proved to be an effective strategysince, otherwise, we would have missed one article.At the end of this stage, 77 articles were selected for data extraction and quality assessment.

3.6 Quality Assessment

The OS and FE studies assessed the quality of the SLR using the set of criteria defined by the Centre forReviews and Dissemination (CDR) Database of Abstracts of Reviewsof Effects (DARE), of the York University[4]. This version of the DARE criteria was based on four questions:

- QA1: Are the review's inclusion and exclusion criteria describedand appropriate?
- QA2: Is the literature search likely to have covered all relevantstudies?
- QA3: Did the reviewers assess the quality/validity of the includedstudies?
- QA4: Were the basic data/studies adequately described?

A defined scoring procedure is used to assign scores to each question, which were then summed to produce the final quality score of the review. For instance, the answer for QA1 was obtained as follows:

- QA1: Y (yes), the inclusion criteria are explicitlydefined in the paper, P (Partly), the inclusion criteria areimplicit; N (no), the inclusion criteria are not defined and cannotbe readily inferred.

The scoring procedure was Y=1, P=0.5, and N=0. In the planning stage of our study, we noticed that the DARE criteria have changed and the current version have 5 questions[5]. Despite this change, we used the same quality criteria of OS and FEto allow for comparability of the results. The DCP was also used in the quality assessment producing the quality scores for all 77 papers.

3.7 Data Extraction Process

We extracted the following data to answer the research questions, from the 77 studies:

- The**Year**ofpublication.
- The **Quality Score** of the study.
- The **Review Type** related to whether the study is a conventional systematic literature review (SLR), a meta-analysis (MA) or a mapping study (MS).
- The **Review Scope** related to the whether the study focused on a detailed technical question (RQ), on (research) trends in a particular software engineering topic area (SERT), or on research methods in software engineering (RT).
- The software engineering **Topic Area** addressed by the study.
- Whether the study explicitly **Cited EBSE papers**([11], [7], and [17]) or **Cited Guidelines**([12] and [13]).
- The **Number of Primary** studies analyzed in the SLR, as stated in the paper either explicitly or as part of tabulations.
- Whether the study **Included Practitioners Guidelines** explicitly as an identifiable part (section, table,...) of the paper.
- The **Source Type** in which the study was first reported (J=journal, C=Conference, WS=Workshop, BS=Book Series).

After analyzing the results of the data extraction we decided to exclude 10 studies: four were not Software Engineering, three were reports of the results of two SLRs that appeared in the FE study, one was from 2010 (out of the time period of this study), one was a shorter version of SE01 published in another journal, and one received zero in the quality evaluation and did not have most of the required information.The DCP was used for data extraction, and at the end of this process 67 articles were selected for further analysis and to answer the research questions. These articles are listed in Appendix A.

Table2: Quality Assessment Scores (sub-set of studies)

Study Ref	QA1	QA2	QA3	QA4	Score	Quartiles
SE01	1	1	1	1	4	
SE02	1	1	1	1	4	
SE05	1	1	1	1	4	
SE18	1	1	1	1	4	
SE40	1	1	1	1	4	
SE46	1	1	0.5	1	4	
SE52	1	1	1	1	4	
SE55	1	1	1	1	4	4th
SE59	1	1	1	1	4	
SE63	1	1	1	1	4	
SE19	1	1	0.5	1	3.5	
SE36	1	1	0.5	1	3.5	
SE39	1	1	0	1	3.5	
SE48	1	1	0	0.5	3.5	
SE75	1	1	0	0.5	3.5	
SE08	0.5	0.5	1	1	2	
SE09	1	1	0	0	2	
SE22	1	0.5	0	0.5	2	
SE23	1	1	0	0	2	
SE24	1	1	0	0	2	
SE25	1	1	0	0	2	
SE26	1	1	0	0	2	
SE44	1	1	0	0	2	
SE45	1	1	0	0	2	
SE49	1	1	0	0	2	
SE56	0	1	0	1	2	
SE60	1	1	0	0	2	
SE64	0.5	1	0	0.5	2	
SE71	1	1	0	0	2	1st
SE72	0	0	0	1	2	
SE03	0.5	1	0	0	1.5	
SE12	1	0.5	0	0	1.5	
SE29	1	0.5	0	0	1.5	
SE32	0.5	1	0	0	1.5	
SE33	0.5	1	0	0	1.5	
SE66	0.5	1	0.5	0	1.5	
SE67	0	0.5	0	1	1.5	
SE04	0	0	0	1	1	
SE10	1	0	0	0	1	
SE13	1	0	1	0	1	
SE58	0	0	0.5	0	1	
SE74	0	0	0	0	0	

4. DATA EXTRACTION RESULTS

A summary of the data collected from the 67 SLRs resulting from the previous processes are shown in Table 3. Regarding the nature of the references to the EBSE papers and SLR Guidelines, similarly to the findings reported by Kitchenham et al. [15][16], all papers that cited the EBSE papers or the guidelines did so as a methodological justification for their study, so we consider all SLR to be EBSE-positioned.

Table2 shows the quality scores for each assessment question.

We ordered the studies by the final score and divided the set into quartiles. Due to space limitation we only present the results of the 4th (top performers) and 1st (bottom performers). This way it is easy to visualize how this sample of studies performed in the assessment. The implications of the quality assessment results are discussed in Section 5.5

Table 3: Systematic Literature Reviews in Software Engineering between July 2008 and December 2009

Study Ref (N=67)	Year	Quality Score	Review Type	Review Focus	Review Topic	Cited EBSE paper	Cited guidelines	Number Primary Studies	Practitioners Guidelines	Paper Type
SE01	2008	4	MS	SERT	Human Aspects	N	Y*	92	N	J
SE02	2008	4	SLR	RT	Knowledge Management	Y♥◊	Y*	68	Y	J
SE03	2008	1,5	MS	RT	Research Topics in Software	N	N	691	N	J
SE04	2008	1	MS	SERT	Software Project Management	N	N	48	N	C
SE05	2008	4	MS	SERT	Agile Software Development	N	Y#	36	Y	J
SE08	2008	2	MS	SERT	Software Testing	N	YO	14	Y	C
SE09	2008	2	MS	SERT	Requirements Engineering	N	Y+	240	N	C
SE10	2008	1	MS	SERT	Usability	N	Y#	51	Y	C
SE11	2008	2,5	MS	SERT	Software Process Improvement	N	Y#	50	Y	C
SE12	2008	1,5	MS	SERT	UML	N	Y#	33	N	CS
SE13	2008	1	SLR	RT	Distributed Software Development	N	N	12	N	C
SE14	2008	3	SLR	RQ	Usability	N	Y+O	63	Y	J
SE18	2009	4	MS	SERT	Software Testing	N	Y#	35	N	J
SE19	2009	3,5	SLR	RT	Software Testing	Y♥◊	Y#O	64	N	J
SE20	2009	2,5	MS	SERT	Software Maintenance and Evolution	N	Y+#	34	N	J
SE21	2009	2,5	MS	SERT	Requirements Engineering	N	Y*	58	N	C
SE22	2009	2	SLR	RQ	Agile Software Development	Y♥	Y*#	9	N	C
SE23	2009	2	MS	RQ	Design Patterns	Y♥	Y#	4	N	C
SE24	2009	2	MS	SERT	Software Maintenance and Evolution	N	Y*	12	Y	C
SE25	2009	2	MS	SERT	Risk Management	N	Y+C	80	N	J
SE26	2009	2	MS	SERT	Software Fault Prediction	N	N	74	N	J
SE27	2009	3	MS	SERT	Software Product Line	N	Y#	34	N	C
SE28	2009	3	MS	SERT	Software Product Line	Y♥	Y#	97	N	C
SE29	2009	1,5	MS	SERT	Requirements Engineering	N	N	46	N	CS
SE30	2009	3	MS	SERT	Software Maintenance and Evolution	N	Y*	176	N	J
SE32	2009	1,5	SLR	RT	Empirical Research Methods	Y♥	Y*	16	N	CS
SE33	2009	1,5	SLR	RQ	Software Security	N	Y#	64	N	C
SE34	2009	2,5	MS	SERT	Empirical Research Methods	N	N	8	N	J
SE35	2008	3	SLR	RQ	Software Testing	N	Y#O	28	N	C
SE36	2009	3,5	MS	SERT	Human Aspects	Y♥	Y*	92	N	J
SE37	2009	3	MA	RQ	Agile Software Development	Y◊	Y#	18	Y	J
SE38	2009	3	MS	SERT	Context Aware Systems	N	N	237	N	J
SE39	2009	3,5	SLR	RQ	Software Maintenance and Evolution	N	Y*	18	Y	C
SE40	2009	4	MS	SERT	Distributed Software Development	N	Y#	20	Y	C
SE42	2009	2,5	SLR	RT	Requirements Engineering	N	Y#	97	Y	J
SE43	2009	2,5	MS	SERT	Distributed Software Development	N	Y#	78	Y	J
SE44	2009	2	MS	SERT	Distributed Software Development	N	Y#	98	Y	C
SE45	2009	2	MS	SERT	Distributed Software Development	N	Y*	122	Y	C
SE46	2009	4	MS	SERT	Software Product Line	N	Y+	89	N	J
SE47	2009	3	MS	SERT	Software Product Line	N	Y*	23	N	C
SE48	2009	3,5	MS	SERT	Risk Management	Y♥	N	27	N	C
SE49	2009	2	MS	SERT	Requirements Engineering	N	N	36	Y	C
SE50	2009	3	MS	SERT	UML	N	Y+C	44	N	J
SE51	2009	3	SLR	RQ	Software Cost Estimation	N	N	12	N	C
SE52	2009	4	SLR	RQ	Software Development	N	Y*	5	Y	CS
SE53	2009	3	MS	SERT	Software Development	N	Y+	40	Y	J
SE54	2009	2,5	MS	SERT	Software Product Line	N	Y*	39	N	C
SE55	2009	4	MS	SERT	Requirements Engineering	N	Y+C	24	Y	J
SE56	2009	2	MS	SERT	Distributed Software Development	N	YO	72	Y	J
SE57	2009	3	MS	SERT	Agile Software Development	Y◊	Y*	50	N	C
SE58	2009	1	MS	SERT	Requirements Engineering	Y■	N	22	N	C
SE59	2009	4	SLR	RQ	Software Maintenance and Evolution	N	Y*+	15	Y	C
SE60	2009	2	MS	SERT	Software Architecture	N	Y*	11	N	CS
SE62	2009	2,5	MS	SERT	Empirical Research Methods	N	Y+	63	N	WS
SE63	2009	4	MS	SERT	Requirements Engineering	Y◊	Y*C	149	Y	J
SE64	2009	2	MS	SERT	Software Testing	N	YO	27	N	C
SE65	2008	2,5	MS	SERT	Software Product Line	N	Y*	17	Y	C
SE66	2009	1,5	MS	SERT	Software Testing	N	Y*	78	Y	J
SE67	2009	1,5	MS	SERT	Distributed Software Development	N	Y*C	12	N	CS
SE68	2009	2,5	MS	SERT	Service Oriented Systems Engineering	N	Y+	51	N	J
SE70	2009	3	MS	SERT	Software Evaluation and Selection	N	N	60	N	J
SE71	2009	2	SLR	RT	Empirical Research Methods	N	Y*	103	N	J
SE72	2009	2	SLR	RQ	Software Development	N	Y*	122	Y	BS
SE74	2009	0	MS	SERT	Empirical Research Methods	N	N	299	N	J
SE75	2009	3,5	MS	SERT	Human Aspects	N	Y*	92	N	J
SE76	2008	3	MS	SERT	Distributed Software Development	N	Y*	26	N	C
SE77	2008	3	MS	SERT	Software Product Line	N	Y#	19	N	C

♥[11], ◊[7], ■[20], *[12], +[10]. #[13], C[3], O[9], S Short Paper

5. DISCUSSIONS

In this section, we address the research questions presented in Section 3.1. We show the results of our study (SE), compare with the findings of OS/FE, and integrate the results (OS/FE+SE).

5.1 RQ1 - The Number of SLRs

Table 4shows the growth in published SLRs since 2004. The OS/FE studies found 53 studies between 2004 and June 2008 (4.5 years), and our extension (SE) found 67 studies between July 2008 and December 2009 (1.5 year). The studies published in 2009 account for 43% (51/120) of the total.

Table 4: Number of SLRs per Year

Year	Number of SLR			Number of EBSE Positioned SLR			
	OS/FE	SE	Total	OS/FE	SE	Total	%[1]
2004	6		6	1		1	17%
2005	11		11	5		5	45%
2006	9		9	6		6	67%
2007	15		15	9		9	60%
2008	12	16	28	10	12	22	79%
2009		51	51		41	41	80%
Total	**53**	**67**	**120**	**31**	**53**	**84**	**70%**

[1] (Total EBSE Positioned SLR/Total SLR) in the same year

Table 4 also shows that the number of SLR that cites either the EBSE papers or the SLR guidelines is also increasing in absolute number and also as a percentage of the studies in a given year. In fact, in SE 80% (53/67) of the SLR cited the EBSE paper, the SLR Guidelines, or both.

5.2 RQ2 - Research Topics Covered by SLRs

As shown in Table 3, the 67 reviews in SE addressed 24 different software engineering topics, 14 of them have not been addressed in OS/FE. The most frequent topics in our SLRs are: Requirements Engineering (8 studies), Distributed Software Development (8), Software Product Line (7), Software Testing (6), Empirical Research Methods (5), Software Maintenance and Evaluation, and Agile Software Development (4). The studies addressing these 6 topics represent 54% (36/67) of the total. In contrast, the 53 reviews of OS/FE addressed 18 topics, and 55% (29/53) of the studies addressed only 3 topics: Software Cost Estimation (12), Empirical Research Methods (11), and Software Development (in general) (6).

These figures indicate that, as the use of SLR becomes more widespread in the software engineering community, not only the number of SLRs increases but also more software engineering topics are covered by systematic reviews. In fact, the120 SLRs of OS/FE+SE addressed 38 distinct software engineering topics.

5.3 RQ3 - Individuals and Organizations Developing SLRs

In the Original Study (OS), a single researcher, MagneJørgensen, from the Simula Lab, Norway, who was involved in 8 studies, dominated the publications of SLRs. At the organizational level, Simula researchers contributed to 11 studies, just over half of the total. The First Extension (FE) showed a tendency of reducing this concentration, as more researchers from different organizations and from other parts of the world started to adopt SLRs as a research method.In total, 103 researchers, from 17 countries, and 46 organizations were involved in the development of SLRs in OS/FE.

In our study, the trend of reduction on the concentration in researchers, organizations, and countries continued to be observed. The number of researchers grew to 159, representing a 50% increase, which can be seen as high if we consider that there was a 26% increase in the number of SLRs compared to OS/FE. This indicates that the number of authors perstudy is increasing. In fact, we found that the percentage of studies co-authored by 3 or more researchers grew from 58% (31/53) to 67% (45/67), and for studies authored by a single researcher went down from 13% (7/53) to 1% (1/67). This indicates an evolution since the SLR guidelines and literature emphasize the need for at least two researchers to perform a SLR to assure higher levels of quality, reduce bias, and increase reliability of the results.

Another finding is that the number of researchers involved in more than one SLR is also increasing. In the OS, only five researchers were involved in more than three studies, in the FE another seven researchers co-authored three or more studies, and in the SE 10 new researchers entered this group. Furthermore, considering OS/FE+SE, there are 24 researchers that co-authored two studies and 125 were involved in only one study. Table 5 shows the 21 researchers that have co-authored three or more SLRs since 2004. This is an indication that, at least for this group of researchers, the use of SLR has gone from being a one off activity to become part of the research methods employed by these researchers.

Table 5: Authors with 3 or more Studies

Authors	Country	OS/FE	SE	OS/FE + SE
Jørgensen	Norway	9		9
GuilhermeTravassos	Brazil	3	3	*6*
Shepperd	UK	6		6
Tore Dyba	Norway	3	3	6
Muhammad Ali Babar	Ireland		5	5
Hannay	Norway	4		4
Sarah Beecham	UK		4	4
Sjøberg	Spain	4		4
Tony Gorschek	Sweden		4	4
AmbrosioToval	Spain		3	3
Helen Sharp	UK		3	3
Hugh Robinson	UK		3	3
Juristo	Spain	3		3
Kampenes	Norway	3		3
Kitchenham	UK	3		3
Maya Daneva	The Netherlands		3	3
Moløkken-Østvold	Norway	3		3
Moreno	Spain	3		3
Nathan Baddoo	UK		3	3
Thelin	Sweden	3		3
Tracy Hall	UK		3	3

In terms of affiliations, in the SE we found 55 organizations with researchers involved in developing SLRs, which combined with the 43 organizations from OS/FE reach the number of 90 distinct organizations since 2004. The countries where these organizations are located also increased in number and become more spread in the various regions of the world. The number of countries grew from 17 in the OS/FE to 25 considering OS/FE+SE, with 8 new countries in the SE study. Asian countries,which did not appear in the OS/FE, contributed to 10 studies (15%, 10/67) in the SE. Only 2 countries that appeared in the OS/FE did not show in SE (Israel and Colombia).As shown in Table 6, researchers affiliated with European organizations still concentrate the vast majority of studies, a tendency that remains since the Original Study. The participation of USA researchers can still be considered low, accounting for fewer than 12% (14/120) of the studies.

Altogether, this seems to indicate that SLRs are becoming more widespread in the scientific community. More researchers are using SLRs as a research method, and this use is spreading beyond Europe, where the majority of the promoters of EBSE and Systematic Reviews reside.

Table 6: SLRs per Country

Region	OS/FE	% (N=53)	SE	% (N=67)	OS/FE + SE	% (N=121)
N. America	9	17%	7	10%	16	13%
S. America	5	9%	8	12%	13	11%
Europe	45	85%	56	84%	101	83%
Asia	0	0%	10	15%	10	8%
M. East	1	2%	0	0%	1	1%
Oceania	5	9%	2	3%	7	6%

5.4 Limitations of the SLRs

Some limitations of the SLRs identified in the OS and FE are discussed in this section.

5.4.1 Review Topics and Extent of Evidence

As discussed in Section 5.2, the number of topics in software engineering covered by SLR and MS has increased since the OS and FE studies. There is no longer a concentration on a single topic (Software Cost Estimation), but there is still a concentration on 6 topics that have been addressed by 55% of the reviews: Empirical Research Methods (16 studies), Software Cost (13), Requirements Engineering (10), Distributed Software Development (9), Software Development(in general) (9),Software Testing (9), and Software Maintenance and Evolution (7).

The FE study reported a reduction on the proportion of papers directed at research methods between the OS (40%) and the new studies found in the FE (18%). In our study, this trend was not observed, as we identified 27% of the studies (18/67) that were directed at research methods or primarily aimed at researchers. In fact, considering the combination of the results of OS/FE and our study, reviews of Empirical Research Methods are the most frequent topic of study, been addressed by over 13% (16/120) of the SLRs.

Consistently with OS and FE studies, mapping studies (MS) analyze more primary studies than conventional systematic reviews (Table 7).

Table 7: Median of Primary Study per SLR&MA and MS

Statistic	2004	2005	2006	2007	2008	2009
Median	26,5	19,5	32	21	26,5	20
# SLRs&MA	6	8	7	8	8	11
Median	-	119	403,5	137	49	54,5
# MS	-	3	2	7	20	40

We found proportionally more MSs(82%, 55/67) than in the combination of OS/FE (32%, 17/53). Conversely, we found proportionally less conventional SLRs (18%, 12/67) than the OS/FE (68%, 36/53). Two reasons might account for this difference. First, we classified the studies using the method presented by Da Silva et al. [6], and the researchers in the OS and FE used an unreported method. In fact, using the results ofDa Silva et al. [6], the proportions in the OS/FE studies change to 72% of MS and 38% of SLR, being closer to the proportions found n SE.Moreover, the OS study did not distinguish between MSs and SLRs, classifying all studies as SLR, which could have increased the number of SLRs in the OS/FE studies

Second, we have found, as shown in sections 5.2 and 5.3, that an increasing number of newcomers (59), that is, researchers that performed an systematic review for the first time, have performed reviews in new topic areas. Performing a MS of a topic area is a natural first step in research, in particular if the area is more recent, for instance, agile development or distributed software development.

5.4.2 Orientation towards the Practice

Twenty reviews in the SE study addressed research questions that might be of interest to practitioners, including 11 that directly addressed technical evaluation questions (RQ). However, only 18%(12/67) of these studies explicitly provided guidelines for practitioners. These figures are consistent with the OS and FE studies, showing that there was no increase in the orientation of the SLRs towards the practice of software engineering (Table 8).

Table 8: Practitioners Guidelines in the SLRs

Practitioners Guidelines	OS/FE	SE	OS/FE + SE
N	44	55	99
Y	9	12	21
Y%	17%	18%	18%

Consistently with the low direct orientation towards providing guidelines to practitioners, 58% (39/67) of the reviewsin SE addressed trends in software engineering research that could have only an indirect interest to practitioners and 8 studies investigated research methods, with no interest to practice.

5.4.3 Quality Evaluation of Primary Studies

The proportion of the SLRs that undertake evaluation of the quality of primary studies increased when comparing SE and OS/FE studies, as shown in Table 9. Although this indicates an improvement, the numberof reviews performing a full and explicit quality evaluation is still very low, amounting to only 21% (14/67) of the reviews in the SE study.

Table 9: Evolution of Quality Evaluation of Primary Studies

Evaluate Quality of Primary Studies?	OS/FE	SE	OS/FE+SE
N	37	22	59
Y[1]	16	45	61
Y%	30%	67%	51%

[1]Includes full evaluation (score=1) and implicit evaluation (score=0.5)

Three situations that might explain the low numbers of quality assessment of primary reviewswere found. First, some researchers seemed to have confused quality assessment with explicitly stating the inclusion/exclusion criteria of primary studies, and therefore might have thought that no further quality assessment was necessary[SE42]. Second, in some cases the quality assessment was thought to be unnecessary because the primary studies were retrieved from “trustworthy sources” (e.g., peer reviewed journals) and this was considered to be enough to guarantee the quality of primary studies [SE43]. Third, in other cases, the search process found so few relevant studies that the researchers might have feared that applying quality criteria would let them with no studies to analyze [SE39].

Table 10: Use of Guidelines

Citation	OS/FE	% (N=53)	SE	% (n=67)	OS/FE+ SE	% (N=120)
EBSE	7	13%	12	18%	19	16%
Guidelines	27	51%	51[1]	76%	81	68%
EBSE & Guidelines	3	6%	10	15%	13	11%

[1]Excluding 3 studies that cited non EBSE related review guidelines.

5.4.4 Use of Guidelines

The use of guidelines and citations to the EBSE papers in the reviews increased in the SE with respect to OS/FE studies, as shown in Table 10.The increase in the use of guidelines was significantly correlated with quality of the SLRs by Kitchenham et al.[16]. However, the regression test using Cited Guidelines as the factor and quality score as the dependent variable show no statistical significance for the entire set of SLR (N=120).

5.5 Quality of the SLRs

Kitchenham et al. [16] observed that the quality of the SLRsincreased from the OS to the FE study. This trend continued in our study with a steady increase in the mean of the quality scores of the studies every year, except in 2007 (Table 11). Considering the six years period between 2004 and 2009, the increase in quality is 12.5%.

Table2 presents two extracts of the SLR with respect to the quality score. A closer look at Table2 shows that almost all studies in both extracts performed well in both QA1 and QA2, which are related to inclusion and exclusion criteria, and coverage of the search process. This is probably due to the increasing number of studies using the SLR guidelines to plan the studies. Studies in the 4th quartile performed well in all questions. Also, most studies in the 1st quartile failed on QA3 or QA4, which are related to quality assessment of primary studies and synthesis and presentation of findings related to individual primary studies.

We then compared the mean of the quality scores of the SLR with respect to three other factors. First, the SLRs that explicitly provide guidelines for practitioners have higher mean quality score (Mean = 2.85, σ = 0.91) than those that did not provide (Mean = 2.38, σ = 0.83). Second, SLRs published in Journals have more quality (Mean = 2.69, σ = 0.94) than the studies published in Conferences (Mean = 2.44, σ = 0.81). Third, the SLRs with scope RQ performed better (Mean = 2.88, σ = 0.76), than SERT (Mean = 2.41, σ = 0.91) and RT (Mean = 2.28, σ = 0.79).

We performed a regression analysis using these three factors and the result was statistically significant for a 95% confidence level as follows: Guidelines for Practitioners (B = 0.183,std error = 0.038, p = 0.000), Journal (B = 0.117,std error = 0.041, p = 0.005) and RQ (B = 0.081, std error = 0.036, p = 0.025).

Finally, we correlated the number of primary studies in an SLR with the quality score using Pearson coefficient and found that the inverse correlation is significant (*r*=-0.204, N=120, p=0.05). As discussed by Kitchenham et al. [16], SLRs addressing larger numbers of primary studies have lower quality scores than those with fewer primary studies. A possible explanation is that by being faced with too many studies to analyze, researchers may opt not to perform quality assessment and they also have more difficulties in presenting good synthesis and summary of evidence for each paper, thus scoring low on quality questions QA3 and QA4.

Table 11: SLR Quality

Cited Guidelines?	Statistics	2004	2005	2006	2007	2008	2009
No	# SLRS	6	6	4	7	6	10
	Mean	2,08	2,33	2,00	1,79	1,50	2,15
	σ	1,07	0,52	1,08	0,81	0,55	1,08
Yes	#SLRs	0	5	5	8	22	41
	Mean	-	2,20	3,10	3,00	2,80	2,72
	σ	-	0,27	0,65	0,60	0,78	0,81
All	# SLR	6	11	6	15	28	51
	Mean	2,08	2,27	2,61	2,43	2,50	2,61
	σ	1,07	0,41	0,99	0,92	0,92	0,89
	Increase		5%	7,5%	-5%	2,5%	2,5%

6. CONCLUSIONS

Two major problems with SLRs are to find all relevant studies and to assess their quality. In our study, we employed a mixed process approach to find relevant studies that combined automatic search in search engines, manual search on relevant journals and conference proceedings, and backward search, that is, searching for relevant studies in the references of previously selected studies. We checked the coverage of our automatic search and only failed to recover one study in a set of 51, which can be considered good if the automatic search is complemented by manual procedures.

Quality assessment of the SLRs was performed by at least two researchers and conflicts were resolved by a third researcher or by a consensus meeting in the cases in which the third point of way was also conflicting. This multi-evaluator procedure increases the confidence on the reliability of our quality assessment. However, we found the scoring procedure to be too subjective for question QA4 and inconsistent for QA2. We solved the inconsistency problem by consulting the researchers that performedthe OS/FE studies. The problems with question QA4 caused many disagreements between evaluators that were only solve in the consensus meeting.

Our findings show three major limitations with the current use of SLRs in software engineering. First, a large number of SLRs do not assess the quality of their primary studies. For conventional SLRs, and in particular those conducting meta-analysis to combine the effect size of some treatment, the quality of the primary studies is vital. However, this is a minor problem for MSs, since the goal in this case is not to present a combined effect size, but a broader overview of the research trends. Furthermore, since meta-analysis is very rare, this problem has not been an issue in practice.

Second, the integration of results of the primary studies was poorly conducted by many SRLs. We believe the actual problem is that these SLRs, in particular the MSs, are trying to combine and synthesize results from too diverse set of primary studies. This may be caused by the scarcity of empirical replications in software engineering.

Third, as has been identified in the OS/FE studies, the number of SLRs providing guidelines to practitioners is still small. Furthermore, we could not identify from the reported data in the SLRs, whether the origin of the investigated problem was the practice of software engineering in industry, or it was an academic problem. Since the practical origin of the problem and the practitioners guidelines are essential for developing steps 1, 4, and 5 of the EBSE approached discussed in Section 1, we should conclude that EBSE is not being fully realized in practice. On the other hand, the number of SLRs is increasing, along with the number of researchers and organizations performing them. This might indicate that SLRs are being adopted as a research method for the discovery of gaps and trends that could guide academic research in software engineering. This is corroborated by the increase in the proportion of mapping studies, since these are typically directed towards exploratory investigation of research trends.

As future work, we will investigate the extent to which the EBSE is being realized regarding the development of all steps defined in Section 1. One research approach will be to conduct a broad field survey with the researchers involved in the 120 SLRs to investigate the origin and motivation of their problems, and the application of the results of their SLRs. We also intend to make constant updates to this tertiary study at least on an early basis. Finally, we started to investigate the methods used by the researchers to integrate qualitative data. This is relevant due to the increasing incidence of case studies and other forms of qualitative research in software engineering. We expect to produce, from the best approaches of qualitative data analysis employed, guidelines for researchers performing SLRs of qualitative studies, as has been investigated in other fields.

7. REFERENCES

[1] Arksey, H.; O´Malley, L. (2005). Scoping studies: towards a methodological framework, International Journal of Social Research Methodology, 8(1), 19-32.

[2] Barcelos, R. F.; Travassos, G. H. (2006). Evaluation Approaches for Software Architectural Documents : a Systematic Review. *IDEAS*, p. 14. La Plata, Argentina.

[3] Biolchini, J. et al. (2005). Systematic Review in Software Engineering. Technical Report RT-ES 679-05. PESC, COPPE/UFRJ.

[4] Centre for Reviews and Dissemination. (2007). What are the criteria for the inclusion of reviews on DARE? Available at <http://www.york.ac.uk/inst/crd/faq4.htm>, (accessed 24.07.07, no longer working, use [5])

[5] Centre for Reviews and Dissemination. (2010). About DARE. Available at <http://www.york.ac.uk/inst/crd/darefaq.htm> (accessed 20.08.2010)

[6] Da Silva, F. Q. B. et al. (2010). Critical Appraisal of Systematic Reviews in Software Engineering: a Tertiary Study. ESEM'2010, Bolzano-Bolzen, Italy.

[7] Dybå, T.; et al. (2005). Evidence-based software engineering for practitioners, IEEE Software 22 (1) 58–65.

[8] Greenhalgh, T. (2007). How to Read a Paper: the Basics of Evidence-Based Medicine. Backwell Publishing, London.

[9] Khan, R. et al. (2003).Systematic Reviews to Support Evidence-Based Medicine: How to Review and Apply Findings of Healthcare Research. Royal Society of Medicine Press Ltd.

[10] Kitchenham, B. et al. (2002). Preliminary Guidelines for Empirical Research in Software Engineering. IEEE TSE, 28(8), 721–734.

[11] Kitchenham, B. et al. (2004). Evidence-based Software Engineering. 26th International Conference on Software Engineering, (ICSE), IEEE, Washington DC, pp 273 – 281.

[12] Kitchenham, B.A. (2004). Procedures for Undertaking Systematic Reviews. Joint Technical Report, Computer Science Department, 2004, Keele University (TR/SE-0401) and National ICT Australia Ltd (0400011T.1).

[13] Kitchenham, B.; Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering, Technical Report EBSE-2007-01, School of Computer Science and Mathematics, Keele University.

[14] Kitchenham, B.A. et al. (2008). Protocol for extending a Tertiary Study of Systematic Literature Reviews in Software Engineering. EPIC Technical Report, EBSE-2008-006.

[15] Kitchenham, B. et al. (2009). Systematic literature reviews in software engineering – A systematic literature review. Information and Software Technology, 51 7–15.

[16] Kitchenham, B. et al. (2010).Systematic Literature Reviews in Software Engineering– A Tertiary Study. Information and Software Technology, 52(8), pp. 792–805.

[17] Jørgensen, M.et al. (2005). Teaching evidence-based software engineering to university students. METRICS'05.

[18] Petersson, H. et al. (2004). Capture-recapture in Software Inspections after 10 Years Research – Theory, Evaluation and Application. JSS, 72(2), 249-264.

[19] Petticrew, M.; Roberts, H. (2006). Systematic Reviews in the Social Sciences. Blackwell Publishing.

[20] Rainer, A.; Beecham, S. (2008). Supplementary Guidelines and Assessment Scheme for the use of Evidence Based Software Engineering. Technical Report CS-TR-469, University of Hertfordshire: Hatfield, U.K.

[21] Sackett, D.L. et al. (1996). Evidence-Based Medicine: what it is and what it isn't. BMJ 1996;312:71-2.

[22] Shaw, M.; Clements, P. (2006). The Golden Age of Software Architecture: A Comprehensive Survey. IEEE Software, 23(2), 31-39.

Appendix A –THE SYSTEMATIC REVIEWS

[SE01] Beecham, S. et al. Motivation in Software Engineering: A systematic literature review. IST, 50(9-10), 860-878, 2008.

[SE02] Bjornson, F. O.; Dingsoyr, T. Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used. Knowledge Management, 50(11), 1055-1068, 2008.

[SE03] Cai, K.; Card, D.An analysis of research topics in software engineering – 2006.JSS, 81(6), 1051-1058, 2008.

[SE04] Dong, F. et al. Software Multi-project Resource Scheduling.ICSP, 63-75, 2008.

[SE05] Dybå, T.et al. Empirical studies of agile software development: A systematic review. IST, 50(9-10), 833-859, 2008.

[SE08] Haugset, B.; Hanssen, G. K. Automated Acceptance Testing: A Literature Review and an Industrial Case Study. Agile, 27-38.IEEE, 2008.

[SE09] Herrmann, A.; Daneva, M. Requirements Prioritization Based on Benefit and Cost Prediction: An Agenda for Future Research, 16th RE, 125-134, IEEE, 2008.

[SE10] Insfran, E. et al. A Systematic Review of Usability Evaluation in Web Development *.Workshops on Web Information Systems Engineering, 81-91, 2008.

[SE11] Kalinowski, M. et al. Towards a Defect Prevention Based Process Improvement Approach.34th Euromicro, 199-206.IEEE, 2008.

[SE12] Pretorius, R.; Budgen, D. A mapping study on empirical evidence related to the models and forms used in the uml. IEEE-ACM ESEM. IEEE, 2008.

[SE13] Smite, D. et al. Reporting Empirical Research in Global Software Engineering: A Classification Scheme. ICGSE, 173-181.IEEE, 2008.

[SE14] Van Velsen, L. et al. User-centered evaluation of adaptive and adaptable systems: a literature review. The Knowledge Engineering Review, 23(03), 261-281, 2008.

[SE18] Afzal, W et al. A systematic review of search-based testing for non-functional system properties.IST, 51(6), 957-976.Elsevier, 2009.

[SE19] Ali, S. et al.A Systematic Review of the Application and Empirical Investigation of Search-Based Test Case Generation. IEEE TSE, PP(99), 1-22, 2009.

[SE20] Benestad, H. C. Empirical Assessment of Cost Factors and Productivity during Software Evolution through the Analysis of Software Change Effort.Sciences-New York, 2009.

[SE21] Blanes, D. et al. Requirements Engineering in the Development of Multi-Agent Systems: A Systematic Review. 10th IDEAL, 510-517, 2009.

[SE22] Brereton, P. et al. Pair programming as a teaching tool: a student review of empirical studies. 22nd CSEE&T, 2009.
[SE23] Budgen, D.; Zhang, C. Preliminary Reporting Guidelines for Experience Papers. EASE, 1-10, 2009.
[SE24] Burrows, R. et al. Coupling Metrics for Aspect-Oriented Programming: A Systematic Review of Maintainability Studies (Extended Version). ENASE, 2009.
[SE25] Calvo-ManzanoVillalón, J. a. et al. State of the art for risk management in software acquisition. ACM SIGSOFT Software Engineering Notes, 34(4), 1, 2009.
[SE26] Catal, C.; Diri, B.A systematic review of software fault prediction studies.Expert Systems with Applications, 36(4), 7346-7354. Elsevier, 2009.
[SE27] Chen, L. et al. Variability Management in Software Product Lines: A Systematic Review. 13th SPLC, 81-90, 2009.
[SE28] Chen, L. et al. A Status Report on the Evaluation of Variability Management Approaches. 13th EASE, 2009.
[SE29] Condori-Fernandez, N. et al.A Systematic Mapping Study on Empirical Evaluation of Software Requirements Specifications Techniques.ESEM, 502-505, IEEE, 2009.
[SE30] Cornelissen, B. et al. A Systematic Survey of Program Comprehension through Dynamic Analysis. IEEE TSE, 35(5), 684-702, 2009.
[SE32] dos Santos, P. S. et al. Action research use in software engineering: An initial survey. ESEM, 414-417. IEEE, 2009
[SE33] Du, J. et al.An Analysis for Understanding Software Security Requirement Methodologies. SSIRI, 141-149. IEEE, 2009.
[SE34] Edwards, H. et al. The repertory grid technique: Its place in empirical software engineering research. IST, 51(4), 785-798. Elsevier, 2009.
[SE35] Engström, E. et al. A systematic review on regression test selection techniques.IST, 52(1), 14-30. Elsevier, 2010.
[SE36] Hall, T. et al.A Systematic Review of Theory Use in Studies Investigating the Motivations of Software Engineers.TOSEM, 18(3). ACM 2009.
[SE37] Hannay, J. E. et al. The effectiveness of pair programming: A meta-analysis. IST, 51(7), 1110-1122. Elsevier, 2009.
[SE38] Hong, J. et al. Context-aware systems: A literature review and classification. Expert Systems with Applications, 36(4), 8509-8522. Elsevier, 2009.
[SE39] Hordijk, W. et al. Harmfulness of Code Duplication - A Structured Review of the Evidence. 13th EASE, 2009.
[SE40] Hossain, E.; Babar, M. A. Using Scrum in Global Software Development: A Systematic Literature Review. ICGSE, 175-184. IEEE 2009.
[SE41] Excluded.
[SE42] Ivarsson, M.; Gorschek, T. Technology transfer decision support in requirements engineering research: a systematic review of REj. RE 2009, 155-175, 2009.
[SE43] Jiménez, M. et al. Challenges and Improvements in Distributed Software Development: A Systematic Review. Advances in Software Engineering, 1-15, 2009.
[SE44] Khan, S. U. et al. Critical Barriers for Offshore Software Development Outsourcing Vendors: A Systematic Literature Review. 16th APSEC, 79-86. IEEE, 2009.
[SE45] Khan, S. U. et al. Critical Success Factors for Offshore Software Development Outsourcing Vendors: A Systematic Literature Review. Fourth ICGSE, 207-216.IEEE, 2009.
[SE46] Khurum, M. et al.A systematic review of domain analysis solutions for product lines. JSS, 82(12), 1982-2003. Elsevier, 2009.
[SE47] Lamancha, B. P. et al. Software Product Line Testing A Systematic Review.ICSOFT, 23-30, 2009.
[SE48] Liu, D. et al. The Role of Software Process Simulation Modeling in Software Risk Management: a Systematic Review. Third ESEM, 302-311.IEEE, 2009.
[SE49] Lopez, A. et al. Risks and Safeguards for the Requirements Engineering Process in Global Software Development. Fourth ICGSE, 394-399.IEEE, 2009.
[SE50] Lucas, F. J. et al.A systematic review of UML model consistency management.IST, 51(12), 1631-1645.Elsevier, 2009.
[SE51] Mair, C.; Shepperd, M. A Literature Review of Expert Problem Solving using Analogy. 13th EASE, 2009.
[SE52] Mitchell, S. M. et al. A comparison of software cost, duration, and quality for waterfall vs. iterative and incremental development: A systematic review. ESEM, 511-515. IEEE, 2009.
[SE53] Mohagheghi, P. et al. Definitions and approaches to model quality in model-based software development – A review of literature.IST, 51(12), 1646-1669. Elsevier 209.
[SE54] Montagud, S. et al. Gathering Current Knowledge about Quality Evaluation in Software Product Lines.SPLC, 91-100, 2009.
[SE55] Nicolás, J. et al. On the generation of requirements specifications from software engineering models: A systematic literature review. IST, 51(9), 1291-1307. Elsevier, 2009.
[SE56] Persson, J. S. et al. Managing Risks in Distributed Software Projects: An Integrative Framework. IEEE TOSEM, 56(3), 508-532, 2009.
[SE57] Racheva, Z. et al. Value Creation by Agile Projects: Methodology or Mystery? LNBIP, 32(4), 141-155, 2009.
[SE58] Rainer, A. et al. An assessment of published evaluations of requirements management tools. 13th EASE, 209.
[SE59] Riaz, M. A systematic review of software maintainability prediction and metrics.3rd ESEM, 367-377. IEEE, 2009.
[SE60] Savolainen, J.; Myllarniemi, V. Layered architecture revisited — Comparison of research and practice.WICSA/ECSA, 317-320. IEEE, 2009.
[SE62] Stol, K. et al. The use of empirical methods in Open Source Software research: Facts, trends and future directions. Workshop on Emerging Trends in FLOSS Research and Development, 19-24. IEEE, 2009.
[SE63] Walia, G. S.; Carver, J. C.A systematic literature review to identify and classify software requirement errors.IST, 51(7), 1087-1109. Elsevier, 2009.
[SE64] Zakaria, Z. et al. Unit Testing Approaches for BPEL: A Systematic Review. 16th APSEC, 316-322. IEEE, 2009.
[SE65] de Souza Filho, E. D. et al. Evaluating Domain Design Approaches Using Systematic Review. ECSA, 50-65, 2008.
[SE66] Dias-Neto, A. C.; Travassos, G. H. Model-based testing approaches selection for software projects. IST, 51(11), 1487-1504. Elsevier, 2009.
[SE67] Ebling, T. et al.A Systematic Literature Review of Requirements Engineering in Distributed Software Development Environments.11th ICEIS, 363-366, 2009.
[SE68] Gu, Q. et al. Exploring service-oriented system engineering challenges: a systematic literature review. Service Oriented Computing and Applications, 3(3), 171-188, 2009.
[SE69] Excluded.
[SE70] Jadhav, A. S. et al. Evaluating and selecting software packages: A review. IST, 51(3), 555-563. Elsevier 2009.
[SE71] Kampenes, V. B., et al. A systematic review of quasi-experiments in software engineering.IST, 51(1), 71-82. Elsevier, 2008.
[SE72] Trendowicz, A. Factors Influencing Software Development Productivity – State-of-the-Art and Industrial Experiences.Advances in Computers, 77(09), 185-241, 2009.
[SE74] Zelkowitz, M. V.An update to experimental models for validating computer technology.JSS, 82(3), 373-376.Elsevier, 2009.
[SE75] Sharp, H., et al.Models of motivation in software engineering.IST, 51(1), 219-233.Elsevier, 2009.
[SE76] Prikladnicki, R.et al. Patterns of Distributed Software Development Evolution: A systematic review of the literature. 12th EASE, 2008.
[SE77] Khurum, M. et al. Systematic Review of Solutions Proposed for Product Line Economics.2nd MESPUL, 2008, pp.386-393.