Systematic Mapping Studies in Software Engineering

Kai Petersen[1,2], Robert Feldt[1], Shahid Mujtaba[1,2], Michael Mattsson[1]
[1]School of Engineering, Blekinge Institute of Technology, Box 520
SE-372 25 Ronneby
(kai.petersen | robert.feldt | shahid.mujtaba | michael.mattsson)@bth.se

[2]Ericsson AB, Box 518
SE-371 23 Karlskrona
(kai.petersen | shahid.mujtaba)@ericsson.com

Abstract

BACKGROUND: **A software engineering systematic map is a defined method to build a classification scheme and structure a software engineering field of interest. The analysis of results focuses on frequencies of publications for categories within the scheme. Thereby, the coverage of the research field can be determined. Different facets of the scheme can also be combined to answer more specific research questions.**
OBJECTIVE: **We describe how to conduct a systematic mapping study in software engineering and provide guidelines. We also compare systematic maps and systematic reviews to clarify how to chose between them. This comparison leads to a set of guidelines for systematic maps.**
METHOD: **We have defined a systematic mapping process and applied it to complete a systematic mapping study. Furthermore, we compare systematic maps with systematic reviews by systematically analyzing existing systematic reviews.**
RESULTS: **We describe a process for software engineering systematic mapping studies and compare it to systematic reviews. Based on this, guidelines for conducting systematic maps are defined.**
CONCLUSIONS: **Systematic maps and reviews are different in terms of goals, breadth, validity issues and implications. Thus, they should be used complementarily and require different methods (e.g., for analysis).**

Keywords: Systematic Mapping Studies, Systematic Reviews, Evidence Based Software Engineering

1. INTRODUCTION

As a research area matures there is often a sharp increase in the number of reports and results made available, and it becomes important to summarize and provide overview. Many research fields have specific methodologies for such secondary studies, and they have been extensively used in for example evidence based medicine. Until recently this has not been the case in Software Engineering (SE). However, a general trend toward more evidence based software engineering (Kitchenham et al. 2004) has lead to an increased focus on new, empirical and systematic research methods. There have also been proposals for more structured reporting of results, using for example structured abstracts (Budgen et al. 2007).

The systematic literature review is one secondary study method that has gotten much attention lately in SE (Kitchenham & Charters 2007, Dybå et al. 2006, Hannay et al. 2007, Kampenes et al. 2007) and is inspired from medical research. Briefly, a systematic review (SR) goes through existing primary reports, reviews them in-depth and describes their methodology and results. Compared to literature reviews common in any research project, a SR has several benefits: a well-defined methodology reduces bias, a wider range of situations and contexts can allow more general conclusions, and use of statistical meta-analysis can detect more than individual studies in isolation (Kitchenham & Charters 2007). However, SRs also have several drawbacks, the

main one being the considerable effort required. In software engineering the systematic reviews have focused on quantitative and empirical studies, but a large set of methods for synthesizing qualitative research results exists (Dixon-Woods et al. 2005).

Systematic mapping is a methodology that is frequently used in medical research, but that have largely been neglected in SE. To the best of our knowledge there is only one clear example of a systematic mapping study within SE (Bailey et al. 2007). This may be due to that systematic maps have not yet been discovered as a method for aggregating software engineering research. A systematic mapping study provides a structure of the type of research reports and results that have been published by categorizing them. It often gives a visual summary, the map, of its results. It requires less effort while providing a more coarse-grained overview. Previously, systematic mapping studies in software engineering have been recommended mostly for research areas where there is a lack of relevant, high-quality primary studies (Kitchenham & Charters 2007).

In this paper we analyze the differences between systematic review and systematic mapping studies and argue for a broader set of situations where the latter is appropriate. In Section 2 we describe a detailed process for systematic maps. Section 3 summarizes the existing SE systematic reviews and contrasts them with systematic maps. Section 4 then discusses additional guidelines for systematic maps before we conclude in Section 5.

2. THE SYSTEMATIC MAPPING PROCESS

We have adapted and applied systematic mapping to software engineering in a study focusing on software product line variability (Mujtaba et al. 2008). In the following, we detail the process we used. We also discuss some of the choices in the systematic map by (Bailey et al. 2007).
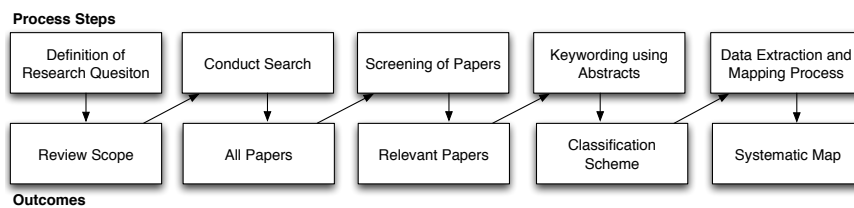

FIGURE 1: The Systematic Mapping Process

The essential process steps of our systematic mapping study are definition of research questions, conducting the search for relevant papers, screening of papers, keywording of abstracts and data extraction and mapping (see Figure 1). Each process steps has an outcome, the final outcome of the process being the systematic map.

2.1. Definition of Research Questions (Research Scope)

The main goal of a systematic mapping studies is to provide an overview of a research area, and identify the quantity and type of research and results available within it. Often one wants to map the frequencies of publication over time to see trends. A secondary goal can be to identify the forums in which research in the area has been published. These goals are reflected in both papers' research questions (RQs) which are similar, as shown in Table 1.

TABLE 1: Research Questions for Systematic Maps

Object Oriented Design Map (Bailey et al. 2007)	Software Product Line Variability Map (Mujtaba et al. 2008)
RQ1: Which journals include papers on software design? **RQ2:** What are the most investigated object oriented design topics and how have these changed over time? **RQ3:** What are the most frequently applied research methods, and in what study context?	**RQ1:** What areas in software product line variability are addressed and how many articles cover the different areas? **RQ2:** What types of papers are published in the area and in particular what type of evaluation and novelty do they constitute?

2.2. Conduct Search for Primary Studies (All Papers)

The primary studies are identified by using search strings on scientific databases or browsing manually through relevant conference proceedings or journal publications. A good way to create the search string is to structure them in terms of population, intervention, comparison, and outcome (Kitchenham & Charters 2007). The structure should of course be driven by the research questions. Keywords for the search string can be taken from each aspect of the structure. For example, the outcome of a study (e.g., accuracy of an estimation method) could lead to key words like "case study" or "experiment" which are research approaches to determine this accuracy.

The main difference between the studies is that we do not consider specific outcomes or experimental designs in our study (Mujtaba et al. 2008). We avoided this restriction since we wanted a broad overview of the research area as a whole. If we had only considered certain types of studies the overview could have been biased and the map incomplete. Some sub-topics might be over- or under-represented for certain study methods. This difference is also reflected in the search strings:

- *Object Oriented Design Map:* ("object oriented" AND "design" AND "empirical evidence") OR ("OO" AND "empirical" AND "design") OR ("software design" AND "OO" AND "experimental")
- *Software Product Line Variability Map:* "software" AND ("product line" OR "product family" OR "system family") AND ("variability" OR "variation")

The choice of databases was also different. For the object oriented design map, all results from relevant databases for computer science and software engineering were taken into consideration. However, we only consider the main forums for software product line research, namely the Software Product Line Conference (SPLC) [1] and the Workshop on Product Family Engineering (PFE). Furthermore, we considered journal articles in addition to that. As the SPLC is the main forum to publish product line research, this is a good starting point to determine the classification scheme and distribution of articles between identified categories.

2.3. Screening of Papers for Inclusion and Exclusion (Relevant Papers)

Inclusion and exclusion criteria are used to exclude studies that are not relevant to answer the research questions. The criteria in Table 2 show that the research questions influenced the inclusion and exclusion criteria, thus the empirical part is considered only for the object oriented design map. We found it useful to exclude papers which only mentioned our main focus, variability, in introductory sentences in the abstract. This was needed since it is a central concept in the area and thus is frequently used in abstracts without papers really addressing it any further. We prototyped this technique and did not find any misclassifications because of it.

TABLE 2: Inclusion and Exclusion Criteria

Object Oriented Design Map (Bailey et al. 2007)	Software Product Line Variability Map (Mujtaba et al. 2008)
Inclusion: books, papers, technical reports and grey literature describing empirical studies regarding object oriented software design. Where several papers reported the same study, only the most recent was included. Where several studies were reported in the same paper, each relevant study was treated separately.	**Inclusion:** The abstract explicitly mentions variability or variation in the context of software product line engineering. From the abstract, the researcher is able to deduce that the focus of the paper contributes to product line variability research.
Exclusion: Studies that did not report empirical findings or literature that was only available in the form of abstracts or Powerpoint presentations.	**Exclusion:** The paper lies outside the software engineering domain. Variability and variation are not part of the contributions of the paper, the terms are only mentioned in the general introductory sentences of the abstract.

2.4. Keywording of Abstracts (Classification Scheme)

In (Bailey et al. 2007) the process of how the classification scheme was created was not clearly described. For our study, we followed a systematic process shown in Figure 2. Here *Keywording*

---

[1] www.splc.net

is a way to reduce the time needed in developing the classificaiton scheme and ensuring that the scheme takes the existing studies into account. Keywording is done in two steps. First, the reviewers read abstracts and look for keywords and concepts that reflect the contribution of the paper. While doing so the reviewer also identifies the context of the research. When this is done, the set of keywords from different papers are combined together to develop a high level understanding about the nature and contribution of the research. This helps the reviewers defining a set of categories which is representative of the underlying population. When abstracts are of too poor quality to allow meaningful keywords to be chosen, reviewers can choose to study also the introduction or conclusion sections of the paper. When a final set of keywords has been chosen, they can be clustered and used to form the categories for the map.
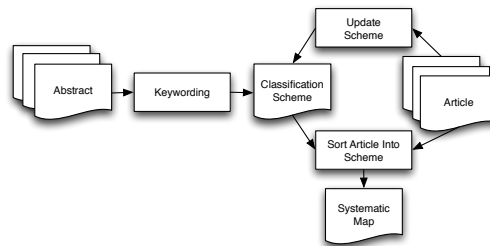

FIGURE 2: Building the Classification Scheme

In our study, three main facets were created. One facet structured the topic (i.e., software product line variability), for example in terms of architecture variability, requirements variability, implementation variability, variability management and so forth. Furthermore, the type of contribution was considered, which for example could be a process, method, tool etc. These categories were derived from the keywords. However, the research facet which reflects the research approach used in the papers is general and independent from a specific focus area. We choose an existing classification of research approaches by Wieringa et al. (Wieringa et al. 2006), summarized in Table 3.

TABLE 3: Research Type Facet

Category	Description
Validation Research	Techniques investigated are novel and have not yet been implemented in practice. Techniques used are for example experiments, i.e., work done in the lab.
Evaluation Research	Techniques are implemented in practice and an evaluaiton of the technique is conducted. That means, it is shown how the technique is implemented in practice (solution implementation) and what are the consequences of the implementation in terms of benefits and drawbacks (implementation evaluation). This also includes to identify problems in industry.
Solution Proposal	A solution for a problem is proposed, the solution can be either novel or a significant extension of an existing technique. The potential benefits and the applicability of the solution is shown by a small example or a good line of argumentation.
Philosophical Papers	These papers sketch a new way of looking at existing things by structuring the field in form of a taxonomy or conceptual framework.
Opinion Papers	These papers express the personal opinion of somebody whether a certain technique is good or bad, or how things should been done. They do not rely on related work and research methodologies.
Experience Papers	Experience papers explain on what and how something has been done in practice. It has to be the personal experience of the author.

We found these categories easy to interpret and use for classification without evaluating each paper in detail (as it is done for a systematic review). For example, evaluation research can be excluded if no industry cooperation or real world project is mentioned. Furthermore, validation research is easy to pinpoint by checking whether the paper states hypotheses, uses summary statistics (e.g., figures like scatter diagrams or histograms) and describes the main components of an experimental setup. Furthermore, the scheme allows to classify non-empirical research in the categories solution proposal, philosophical papers, opinion papers and experience papers. In

our review, the majority of papers was related to the non-empirical category. Other research type classifications have been proposed, and they are discussed in Section 4.

2.5. Data Extraction and Mapping of Studies (Systematic Map)

When having the classification scheme in place, the relevant articles are sorted into the scheme, i.e., the actual data extraction takes place. As shown in Figure 2 the classification scheme evolves while doing the data extraction, like adding new categories or merging and splitting existing categories. In this step, we used an Excel table to document the data extraction process. The table contained each category of the classification scheme. When the reviewers entered the data of a paper into the scheme, they provided a short rationale why the paper should be in a certain category (for example, why the paper applied evaluation research). From the final table, the frequencies of publications in each category can be calculated.

The analysis of the results focuses on presenting the frequencies of publications for each category. This makes it possible to see which categories have been emphasized in past research and thus to identify gaps and possibilities for future research. The two maps used different ways of presenting and analyzing the results.

The object oriented design map is illustrated using summary statistics in form of tables, showing the frequencies of publications in each category. The intervention type was used in the object oriented design map to structure the topic and counted the number of papers for each intervention type. In our study, we used a bubble plot to report the frequencies, shown in Figure 3. This is basically two x-y scatterplots with bubbles in category intersections. The size of a bubble is proportional to the number of articles that are in the pair of categories corresponding to the bubble coordinates. The same idea is used two times, in different quadrants of the same diagram to show the intersection with the third facet. If a systematic map has more facets than three additional bubble plots could be added either in the same diagram or by having multiple diagrams for different facet combinations. We think the bubble plot supports analysis better than frequency tables. It is easier to consider different facets simultaneously, and summary statistics can still be added for facets individually. It is also more powerful in giving a quick overview of a field, and thus to provide a map. Further visualization alternatives could be found in statistics, human computer interaction (HCI) and information visualisation fields.
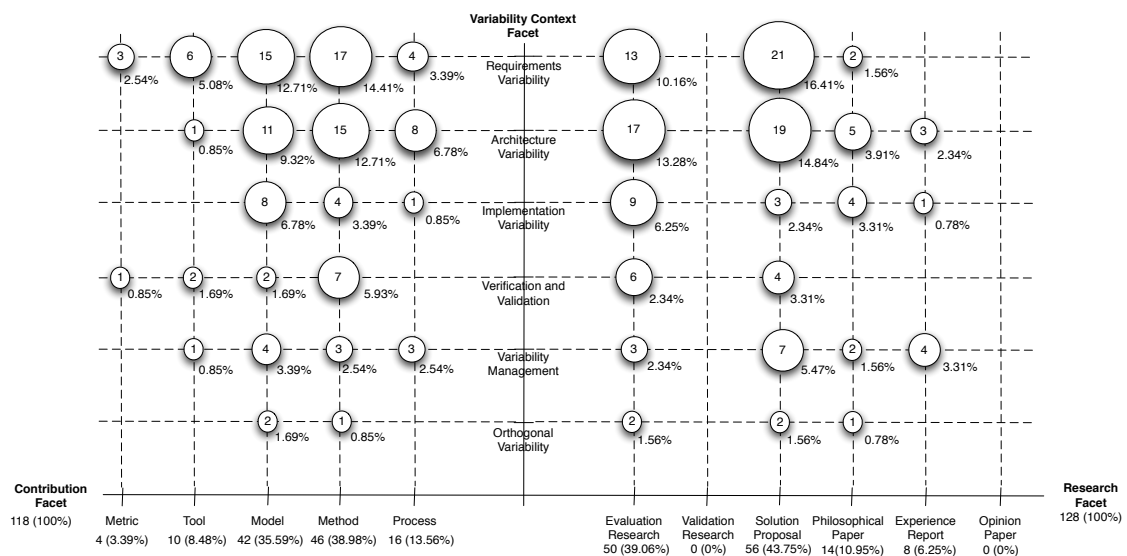

FIGURE 3: Visualization of a Systematic Map in the Form of a Bubble Plot

3. COMPARATIVE ANALYSIS AND DISCUSSION

We have studied existing systematic reviews in software engineering and characterized them. This serves as input to the comparison and discussion in this section. The systematic review studies were identified using the following search string: *"systematic review" AND "software engineering"*, and by searching Inspec & Compendex, IEEExplore and ACM Digital Library. The search resulted in a total of 21 papers. We excluded papers that were not in the area of software engineering, were not based on (Kitchenham & Charters 2007) or did not explicitly state in title or abstract that they were systematic reviews. This resulted in eight systematic reviews being included. We also included two further systematic reviews identified in (Kitchenham 2007) since they also match our criteria for inclusion. The reference are summarized in the following table.

TABLE 4: Systematic Reviews Included

ID	Reference	ID	Reference
1	(Dybå et al. 2006)	6	(Mendes 2005)
2	(Grimstad et al. 2006)	7	(Sjøberg et al. 2005)
3	(Hannay et al. 2007)	8	(Jørgensen & Shepperd 2007)
4	(Kampenes et al. 2007)	9	(MacDonell & Shepperd 2007)
5	(Kitchenham et al. 2007)	10	(Davis et al. 2006)

3.1. Characterizing Existing Systematic Reviews

For each of the ten included SE systematic review we characterized them based on their research goals, criteria for inclusion and exclusion, the number of inclusions and exclusions, classification scheme and means of analysis:

- **Research Goals:** A study that aims to 'Identify Best and Typical Practices' analyzes a set of empirical studies to determine which techniques are used and work in practice. For 'Classification and Taxonomy' a study creates a framework or classifies the existing research. 'Emphasis on Topic Categories' means that the study identifies how much research is published in different sub-topics in the field of interest. Finally, a study which 'Identify Publication Fora', identifies the journals, conferences and workshops relevant in the focus area.
- **Inclusion Requirements:** Two main inclusion requirements was found: 'Research is within focus area' and 'Empirical Methods Used'. In the latter category the included papers used empirical methods.
- **Number of Articles Included:** In this category we identify the number of 'Potentially relevant studies' (i.e., found in the search) and the number of 'Included articles' (after applying inclusion and exclusion criteria as well as quality checks).
- **Means of Analysis:** Four types of studies are used: 'Meta studies' integrate several studies through statistical analyzes of the studies' quantitative data. 'Comparative analysis' uses logical simplification and confidence assessment theories. 'Thematic analysis' counts papers related to specific themes or categories. 'Narrative summaries' focus on qualitative review and narrative explanations. Further means of analysis are described in (Dixon-Woods et al. 2005), but we have not found evidence of their use in software engineering systematic reviews.

The summary of our characterization is shown in Table 5, the reference IDs refer to the studies in Table 4. It shows that a majority of reviews aim at identifying best practices in software engineering (Studies 2, 5, 6, 8, 9, 10). Most of these focus on the use of empirical methods (see inclusion requirements). The remaining studies (1, 3, 4, and 7) put requirements on the empirical part as they are studying empirical methods in SE. Furthermore, all of the systematic reviews assess whether papers are related to the focus area. Only two reviews (7,8: (Sjøberg et al. 2005, Jørgensen & Shepperd 2007)) focused mainly on classifications and taxonomies and presented frequencies of papers in identified categories through thematic analysis. These two studies also aimed at identifying the relevant publication fora. What distinguishes them from systematic maps is their in-depth analysis in form of a detailed narrative summary.

In the table we can see that the number of potentially relevant studies is large compared to number of studies that were included in the analysis. It is worth noting that three of the reviews we found (Dybå et al. 2006, Hannay et al. 2007, Kampenes et al. 2007) are based on the 103 articles identified in a single of the other studies (Sjøberg et al. 2005). As means of analysis all studies used some form of narrative summary. Two studies used thematic analysis, two studies applied meta analysis and one study used comparative analysis.

TABLE 5: Systematic Review Characteristics

Reference ID	1	2	3	4	5	6	7	8	9	10
Research Goals										
Identify Best and Typical Practices	X	X	X	X	X	X			X	X
Classification and Taxonomy			X				X	X		
Emphasis on Topic Categories							X	X		
Identify Publication Fora							X	X		
Inclusion Requirements										
Research is Within Focus Area	X	X	X	X	X	X	X	X	X	X
Empirical Methods Used	X	X	X	X	X		X		X	X
Number of Included Articles										
Potentially Relevant Studies	5453	963	5453	5453	1344	353	5453	n.a.	185	564
Relevant Studies (Included)	78	24	24	78	10	173	103	304	10	26
Means of Analysis										
Meta Study	X		X							
Comparative Analysis										X
Thematic Analysis							X	X		
Narrative Summary	X	X	X	X	X	X	X	X	X	X

3.2. Comparison

A comparison of systematic maps and reviews was presented already in (Kitchenham & Charters 2007), focusing mainly on differences in breadth and depth. We extend on that based the overview of systematic reviews and on experience from conducting systematic maps.

Difference in Goals: When comparing systematic reviews and maps, it is clear that their goals can be different. As pointed out in (Kitchenham & Charters 2007) a systematic review aims at establishing the state of evidence, even though other goals like classification are mentioned. However, the systematic reviews we have found focus on identifying best practices based on empirical evidence (this is the case for the majority of systematic reviews, see Table 5). This is not a goal for systematic maps, and can not be since they do not study articles in enough detail. Instead, the main focus here is on classification, conducting thematic analysis and identifying publication fora. Both study types share the aim of identifying research gaps. In our product line variability map, we identified gaps by graphing and thus showing in which topic areas and for which research types there is a shortage of publications. The systematic reviews shows where particular evidence is missing or is insufficiently reported in existing studies. This is not possible with systematic maps.

Difference in Process: We see two main differences in the process. In maps, the articles are not evaluated regarding their quality as the main goal is not to establish the state of evidence. Secondly, data extraction methods differ. For the systematic mapping study, thematic analysis is an interesting analysis method, as it helps to see which categories are well covered in terms of number of publications. In systematic reviews, the method of meta analysis requires another level of data extraction in order to continue working with the quantitative data collected in primary studies (Kitchenham & Charters 2007). However, we see no reason for why not several different methods of analysis could be applied in the same study. A thematic summary leading to a map could be the first steps in a more detailed systematic review. We are doing just that based on (Mujtaba et al. 2008).

Difference in Breadth and Depth: In a systematic mapping study, more articles can be considered as they don't have to be evaluated in such detail. Therefore, a larger field can be

structured (e.g., the whole software product line area). This is also reflected in the search string and inclusion criteria that we used in product line variability map. That is, we only considered population and intervention thus introducing fewer limitations and, potentially, getting more search hits. On the other hand, the systematic review by Kitchenham et al. (Kitchenham et al. 2007) state the outcome and quality assessment of the articles as a major focus, which increases the depth and thus the effort required. This could require a more specific focus of the study and thus fewer studies being included. This difference was also recognized in (Kitchenham & Charters 2007).

Classifying the Topic Area: Many reviews have mentioned the lack of methodological rigor in primary studies, e.g. in (Mendes 2005) *"only 5 % of the studies are considered rigorous methodologically research"*. If we restrict our sample of papers to such a small portion of the available papers there is a risk that our overview of the topic area will be incomplete. It is likely that it is also relatively easier to do empirical research in some sub-areas than in others. Thus, a systematic review focusing on papers using some particular method might introduce a bias when it comes to presenting the overall research area. This is also supported by the fact that only a small number of potentially relevant articles is included in the systematic reviews we found above.

Classifying the Research Approach: In our systematic map on software product line variability we used very high level categories to assess the type of paper in terms of novelty and evaluation. Due to the argument before, this is valid as no detailed evaluation of articles can be done when structuring a large area, in consequence the classification has to be high level. On the other hand, a different classification scheme should be used for systematic reviews as the empirical research approach is evaluated in much more detail. Therefore, one review study (Sjøberg et al. 2005) applied the classification scheme proposed by Glass et al (Glass et al. 2002). The scheme is on a quite detailed level, as it distinguishes more than 22 research methods (like action research, conceptual analysis, ethnography, field study, etc.) and 13 research approaches (for example descriptive system, evaluative-deductive, evaluative-critical, etc.). In order to judge a paper regarding these categories requires a much more in-depth analysis of the paper. The high number of categories for systematic reviews and their detail level is particularly visible for the reviews (Hannay et al. 2007, Sjøberg et al. 2005).

Validity Consideration: As pointed out in (Mendes 2005) 73 % of the papers were designated incorrectly, i.e., they for example promised an experiment which was no experiment. The same problem was reported by (Jørgensen & Shepperd 2007) who found that the term experiment was not always used in line with the definition of controlled experiments. Consequently, when not evaluating the papers in such detail within systematic maps, there might be judgmental errors when classifying the papers into detailed categories. This threat is minimized in systematic reviews as here a detailed evaluation of the research methodology is conducted, including extracting data regarding the methodology (e.g., data collection procedures). This effect can be somewhat alleviated by the fact that systematic maps can consider more papers than a review (see above).

Industrial accessibility and relevance: In our contacts with industrial software engineers they often ask for papers that can give a good introduction to a specific software engineering area. Systematic reviews could be good papers to recommend them. When we have done so they often think the studies are too detailed and hard to access. When presenting the systematic map it was easier to spark interest. We think the visual appeal of systematic maps can summarize and help transfer results to practitioners. However, the focus on depth and empirically validated results that the systematic reviews uncover should be of higher importance for practitioners. Thus, systematic reviewers should think of ways of presenting and structuring their results in more accessible ways.

4. GUIDELINES FOR SYSTEMATIC MAPS AND REVIEWS IN SOFTWARE ENGINEERING

Based on the comparison above and our experience with systematic reviews and systematic maps we propose the following extensions to guidelines for these types of studies.

Use Methods Complementarity: We have seen that both methods have different goals that can also partly contradict each other. For example, a good structure of the topic area is hindered by excluding the majority of articles due to lack of empirical evidence. Therefore, different search strategies and inclusion and exclusion criteria have to be applied (as discussed before). A

systematic map should be used as a first step toward a systematic review, i.e., first the topic area is structured and thereafter a specific focus area is investigated with a systematic review. However, in this context it is important to mention that a systematic map without conducting a successive systematic review has a value in itself. It helps to identify research gap in an topic area and provides indications for lack of evaluation or validation research in certain areas with little effort.

Adaptive Reading Depth For Classification: A common view is that mapping studies are often conducted based on only the abstracts. However, we have noticed that abstracts are often misleading and lack important information. As shown in the study by (Budgen et al. 2007) structured abstracts considerably improve the understandability so we encourage them being proposed and mandated more widely in software engineering. When they are not available we propose an adaptive strategy towards the choice of level of detail: do not pre-specify that only certain parts of a paper can be read. Instead, allow more detailed study of papers for which it is not clear how they should be classified. The more parts of a paper one considers the more effort is required. However, the validity of the results also increases. A mapping study that goes deeper into the papers can become more like a systematic review. The two type of studies can be considered as different points on a continuum. Regardless of where on this continuum a study is designed to be we think that the more quantitative approach common for mapping studies can complement also systematic reviews.

Classify Papers Based on Evidence and Novelty: Even though one does not evaluate the research methods in detail, high level classification schemes can still be used to classify papers. The classification scheme should also provide categories for non-empirical research. These requirements are well fulfilled by the classification scheme presented by (Wieringa et al. 2006) which we recommend using future systematic maps. A future refinement could be to further divide it into different classes, e.g. based on evidence level and type of novelty.

Visualize Your Data: When counting the frequencies of publications in specific categories, one can determine how well the category is covered. Such information is usually summarized in tables or visualized using bar plots. However, as we found it interesting to combine different categories (e.g., map research methods against topic categories) the systematic map bubble plot is more useful. Bubble plots allow to combine categories with each other and thus the relative emphasis of research on categories is visible from the plot itself. Therefore, we recommend researchers doing systematic maps and reviews to investigate and make use of alternative ways of presenting and visualizing their results. For example, Google's Visualization Toolkit based on GapMinder[2] could be used to create bubble plots that vary over time to better show research trends.

5. CONCLUSION

In this paper we illustrated the systematic mapping process and compared it with systematic reviews. To do this we have characterized and summarized ten existing systematic reviews in software engineering. Our findings are that the study methods differ in terms of goals, breadth and depth. Furthermore, the use of the methods has different implications for the classification of the topic are and the research approach. As a consequence, both methods should and can be used complementary. A systematic map can be conducted first, to get an overview of the topic area. Then the state of evidence in specific topics can be investigated using a systematic review. Furthermore, based on the comparison and our experience with systematic maps we provided a set of extensions to guidelines for systematic maps. They specifically state the importance of visualizing results; a technique which should be more widely used also in systematic reviews. In future work more systematic maps should be conducted to gain further experience with our proposed mapping process and guidelines.

References

Bailey, J., Budgen, D., Turner, M., Kitchenham, B., Brereton, P. & Linkman, S. (2007), Evidence relating to object-oriented software design: A survey, *in* 'Proc. of the 1st Int. Symp. on Empirical Software Engineering and Measurement (ESEM 2007)', pp. 482–484.

---

[2]Examples of how GapMinder works can be found on http://www.gapminder.com

Budgen, D., Kitchenham, B., Charters, S., Turner, M., Brereton, P. & Linjkman, S. (2007), Preliminary results of a study of the completeness and clarity of structured abstracts, *in* 'Proc. of the 11th Int. Conf. on Evaluation and Assessment in Software Engineering 2007', pp. 64–72.

Davis, A. M., Tubío, Ó. D., Hickey, A. M., Juzgado, N. J. & Moreno, A. M. (2006), Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review, *in* 'Proc. of the 15th Int. Conf. on Requirements Engineering (RE 2006)', pp. 176–185.

Dixon-Woods, M., Agarwal, S., Jones, D., Young, B. & Sutton, A. (2005), 'Synthesising qualitative and quantitative evidence: a review of possible methods.', *Journal of Health Services Research and Policy* **10**(1), 45–53.

Dybå, T., Kampenes, V. B. & Sjøberg, D. I. K. (2006), 'A systematic review of statistical power in software engineering experiments', *Information & Software Technology* **48**(8), 745–755.

Glass, R. L., Vessey, I. & Ramesh, V. (2002), 'Research in software engineering: an analysis of the literature', *Information & Software Technology* **44**(8), 491–506.

Grimstad, S., Jørgensen, M. & Moløkken-Østvold, K. (2006), 'Software effort estimation terminology: The tower of babel', *Information & Software Technology* **48**(4), 302–310.

Hannay, J. E., Sjøberg, D. I. K. & Dybå, T. (2007), 'A systematic review of theory use in software engineering experiments', *IEEE Trans. Software Eng.* **33**(2), 87–107.

Jørgensen, M. & Shepperd, M. J. (2007), 'A systematic review of software development cost estimation studies', *IEEE Trans. Software Eng.* **33**(1), 33–53.

Kampenes, V. B., Dybå, T., Hannay, J. E. & Sjøberg, D. I. K. (2007), 'Systematic review: A systematic review of effect size in software engineering experiments', *Information & Software Technology* **49**(11-12), 1073–1086.

Kitchenham, B. (2007), 'The current state of evidence-based software engineering. keynote int. conf. on evaluation and assessment in software engineering (2007)'.

Kitchenham, B. A., Dyba, T. & Jorgensen, M. (2004), Evidence-based software engineering, *in* 'Proc. of the 26th Int. Conf. on Software Engineering (ICSE 2006)', IEEE Computer Society, pp. 273–281.

Kitchenham, B. A., Mendes, E. & Travassos, G. H. (2007), 'Cross versus within-company cost estimation studies: A systematic review', *IEEE Trans. Software Eng.* **33**(5), 316–329.

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.

MacDonell, S. G. & Shepperd, M. J. (2007), Comparing local and global software effort estimation models - reflections on a systematic review, *in* 'Proc. of the 1st Int. Symp. on Empirical Software Engineering and Measurement (ESEM 2007)', pp. 401–409.

Mendes, E. (2005), A systematic review of web engineering research, *in* 'Proc. of the Int. Symp. on Empirical Software Engineering (ISESE 2005)', pp. 498–507.

Mujtaba, S., Petersen, K., Feldt, R. & Mattsson, M. (2008), Software product line variability: A systematic mapping study, *in* 'submission'.

Sjøberg, D. I. K., Hannay, J. E., Hansen, O., Kampenes, V. B., Karahasanovic, A., Liborg, N.-K. & Rekdal, A. C. (2005), 'A survey of controlled experiments in software engineering', *IEEE Trans. Software Eng.* **31**(9), 733–753.

Wieringa, R., Maiden, N. A. M., Mead, N. R. & Rolland, C. (2006), 'Requirements engineering paper classification and evaluation criteria: a proposal and a discussion', *Requir. Eng.* **11**(1), 102–107.