A Mapping Study on Method Engineering — First Results

Marco Kuhrmann, Daniel Méndez Fernández, Michaela Tiessler
Technische Universität München – Software & Systems Engineering
Munich, Germany
{kuhrmann, mendezfe, tiessler}@in.tum.de

ABSTRACT

Context: Software processes have become inherently complex to cope with the various situations we face in industrial project environments. In response to this problem, the research area of Method Engineering arose in the 1990s aiming at the systematization of process construction. **Objective:** Although the research area has gained much attention and offered a plethora of contributions so far, we still have little knowledge about the feasibility of Method Engineering. To overcome this shortcoming, necessary is a systematic investigation of the respective publication flora. **Method:** We conduct a systematic mapping study and investigate, inter alia, which contributions were made over time and which research type facet they address to distill a common understanding of the state-of-the-art. **Results:** Based on the review of 64 publications, our results show that most of those contributions only repeat and discuss formerly introduced concepts, whereas empirically sound evidence on the feasibility of Method Engineering, is still missing. **Conclusion:** Although the research area constitutes many contributions, yet missing are empirically sound investigations that would allow for practical application and experience extraction.

Categories and Subject Descriptors

D.2.9 [**Software Engineering Management**]: Software process models

General Terms

Exprimentation

Keywords

Situational Method Engineering, Mapping Study, Systematic Literature Review

1. INTRODUCTION

Method Engineering arose as a research area in direct response to the problem that industrial complex processes needed systematic adaptation. As there is no silver bullet in software processes that matches all possible needs of projects, a number of authors voted for a flexible adaptation approach [3, 18]. However, flexibility in software process design is, in general, a frequently discussed topic with a yet missing common agreement.

During the past decades, a number of contributions on (Situational) Method Engineering (SME, we use *Method Engineering* in this paper as a synonym) were published. Some authors consider Method Engineering to be the "current most optimistic route" to create flexible and adaptable software processes [15]. Until today, however, it is still unknown which of the available approaches has which practical impact. We can observe selected studies on the application of Method Engineering approaches, or discussions on the feasibility of Method Engineering, e.g., [18, 15]. Yet, it remains unclear what the exact state-of-the-art is w.r.t. the practical application and the feasibility of Method Engineering.

Problem Statement. Although many contributions on Method Engineering where proposed so far, it remains unclear which approaches are established in general and which approaches are disseminated in practice. In other domains, e.g., software process metamodels [33], there is reproducible research. Comparable studies in the area of Method Engineering are not yet available. If at all, studies stay on a comparative level and do not allow for practical application nor knowledge extraction. In summary, we still have little knowledge about the finally established state-of-the-art.

Research Objective. To overcome the shortcoming stated above, we aim at conducting a systematic investigation of the publication flora in Method Engineering to paint a big picture of the state-of-the-art.

Contribution. We contribute a systematic mapping study and analyze which contributions were made over time and of which research type facet those contributions are. This analysis allows us to distill an initial understanding of the maturity and the state of application of SME.

Outline. The remainder of the paper is organized as follows. In Sect. 2, we discuss work related. We introduce the study design in Sect. 3, before discussing the results in Sect. 4. In Sect. 5, finally, we conclude the paper.

2. RELATED WORK

Method Engineering (SME) is a paradigm, which addresses the need for flexible and situation-specific methods and their composition. In 1987, Basili and Rombach [3] fostered the

EASE '13, April 14–16 2013, Porto de Galinhas, Brazil

discussion on more flexibility of software processes. The tailoring of a software process should address project goals and environments. SME was proposed as a paradigm to allow for more flexibility. Especially Brinkkemper [5, 4] and Harmsen [11] provided fundamental research int this area. Based on their research, e.g., Gonzales-Perez worked on the adaptation of SME in software process metamodels [9]. Furthermore, several approaches using SME ideas were proposed, e.g., to support methods' authoring and design [27, 7, 26]. Today, several process frameworks, such as ISO 24744 [19] claim to implement basic SME concepts.

However, although SME was proposed in the mid 1990's and there is a considerable number of contributions based on SME ideas, there are only few studies dealing with analyzing the state-of-the-art and, in particular, the feasibility when applying SME concepts in practice. In 1997 Hofstede and Verhof [18] collected all information available for this domain at the time, provided the first study, and stated that "Much more empirical research is needed to substantiate the claims associated with the potential benefits of situational method engineering." They discussed the definition of methods and method fragments, the selection of method fragments, storage, formalisms, and the retrieval and the assembly of method fragments. Taking into account that SME was a rather "young" concept at this time, Hofstede and Verhof provided a comprehensive collection of relevant concepts and terms. However, their contribution is more of philosophical nature as they discussed available concepts rather than providing any research type classification for those concepts. In 2009, Rolland [31] reviewed the state-of-the-art and compared SME-related concepts and the terminology used. This work should provide "a survey of the main results obtained for the two issues of defining and assembling [reusable method] components." The survey stays, however, philosophical and is similar to the one of Hofstede and Verhof, which had no systematic literature review and/or classification of available contributions according to their research type facets in scope. This also holds for Henderson-Sellers and Ralyté [15] who continued the investigation of the state-of-the-art analyses in 2010.

Existing studies do not provide reproducible reviews or a classifications scheme. There is no study available that critically discusses the state-of-the-art in terms of which concepts are defined and applied, and what are the experiences in applying those concepts to allow for evaluating the feasibility and the degree of dissemination of SME.

In this paper, we close the gap in the analyses of SME-related contributions by providing a systematic mapping study to revise the evolution of the respective publication flora over time w.r.t. considered research type facets [35]. This allows us to draw a first picture about the maturity and the practical feasibility of Method Engineering.

3. STUDY DESIGN

We design the study as a combination of methods used for a systematic mapping study to structure the publication flora and ones used for a systematic literature review (SLR) to conduct an in-depth analysis of the publications (see also Peterson et al. [25] and Kitchenham et al. [21]). The following study design itself is structured according to Runeson et al. [34]. After defining the research questions, we describe the case selection. Finally, we describe how we collect and analyze the data, before discussing the validity procedures.

3.1 Research Questions

Our overall goal is to elaborate the state-of-the-art in SME. To this end, we formulate two research questions:

RQ 1 How many papers were published over the years?

RQ 2 Which research type facets address the contributions?

The first research question aims at investigating which publications were contributed in which year. This gives us the opportunity to analyze particular trends in a quantitative manner. The second research question aims at structuring the contributions according to the research type facets proposed by Wieringa et al. [35] to investigate whether the contributions where of more conceptual nature or more empirical nature. The classification of the research type in combination with the year of publication rounds out the trend analysis and needs an in-depth analysis whereby we consider our study to be not exclusively a mapping study where we classify the publications according to the abstracts and the keywords, but need deeper insights to analyze the state of evidence. One reason is that many contributions classified by the authors as, for example, a "study" need more clarification regarding the type of study, e.g., validation research or evaluation research.

3.2 Case Selection

Peterson et al. [25] propose to initiate a mapping study by (1) constructing the repository via a search of primary (known) papers, (2) screen those papers for inclusion and exclusion according to their relevance to the research questions, and (3) construct the classification scheme of the maps according to the keywords and the abstracts. However, we need a deviation from the standard procedure for two reasons. First, inherent in the research area is that many contributions cannot be allocated to a common area Method Engineering; for instance, many publications arise from other research communities that investigate concepts of software processes and tailoring of any facet, e.g., "organizational tailoring", or "dynamic tailoring". Those exemplary terms already show how the various interpretations of Method Engineering hamper the definition of the search strings and the inclusion and exclusion criteria in advance. Second, we deviate from the classic construction of the maps according to the keywords proposed in the publications as we are especially interested in the research type facets of the papers while following a pre-defined classification scheme that needs to be analyzed independently from the given keywords.

For this reason, we refer to the case selection by following a more pragmatic, yet more time-intensive procedure. We first structure the publications and, thus, lay the foundation for the search strings by following the principles of snow-balling [21]. We use a primary set of publications and manually search for secondary references that are based on the contributions' references sections to find further contributions. This first step results in a set of standard contributions used for testing research questions, search strings, and structuring the publications. For this primary search, we refer to the selected authors and publications (Brinkkemper [5, 4], Harmsen [11], Henderson-Sellers [16, 15], and Hofstede [18]), which later on also serve as control values (the automated search result set has to contain the contributions of those authors, see the previous section). The second step is the automated search in several literature databases, which we introduce in the following.

3.3 Data Collection Procedures

The queries are built based on the keyword lists given by the primary sources and most common terminology in the area of software processes. As main data sources, we rely on established literature databases, which we consider most appropriate for a search. The internal discussion about which databases to select is based on our experiences in the process engineering domain. We select the following five databases: *ACM Digital Library, SpringerLink, IEEE Digital Library, Wiley,* and *Elsevier*. If there is a paper listed in one of those databases, but is only referred, we count it for the database that generates the item, regardless of the actual publication location. In addition to those databases, we also take papers into account that are not referred by the databases, but have to be considered as key contributions, e.g., the PhD thesis of Harmsen [11]. For such contributions, we add a category "misc".

Query Definition. To define our queries, we take a sample of relevant papers, analyze them in order to identify and iteratively refine the search strings, and validate them against a pre-defined list reference authors (see Sect. 3.2). The initial set of key words is: {*software, development, process, tailoring, method, methodology, customization, customisation, adaption, adaptation, ISO, CMMI, SPICE, standard, compliance, study, experience, weaving, situational, engineering, practice*}.

Table 1: Final search strings used for query.

S1	(process **or** method **or** methodology) **and** (tailoring **or** adaption **or** customization **or** customisation)
S2	process tailoring **and** (practice **or** experience **or** study)
S3	method **and** (engineering **or** weaving) **or** situational method engineering

Based on the primary searches and the analysis of the primary sources via snow-balling, we conclude the search strings shown in Tab. 1. We use the search strings and aforementioned literature databases for the data collection. Each result set is transferred to a spreadsheet. Having the single result sets available, all results are combined and used as basis for the data analysis. For each data source, at most the top 160 search results are taken into account.

3.4 Analysis Procedures

In the following, we describe our analysis procedure.

3.4.1 Analysis Preparation

To get the initial set of data to be analyzed, we perform an automated search that requires us to filter and prepare the result set. The data analysis is prepared by harmonizing the data and performing a 3-staged voting process.

Harmonization. Since many contributions occur multiple times or are out of scope, we first clean the result set by eliminating multiple occurrences and eliminating contributions that not deal with computer science.

Voting. We perform a 3-staged voting process to classify the papers as relevant or irrelevant and to build a set of contributions for further investigation. The result sheet therefore contains three columns (attribute "relevance"). The first two columns are used in the first voting stage (one column per researcher). A cell in the column is filled either with *1* (the contribution is relevant) or *0*. If a contribution is finally rated with *2*, it is automatically in the set of contributions for further investigation. However, if a contribution is rated with *0*, it is excluded from further investigation. Only if a contribution is rated with *1*, it is marked to be judged in the secondary voting. The criteria for the voting were (1) the title of the contribution and (2) the abstract. In the second voting stage, we only consider contributions that are not finally decided and call in a third reviewer. This third reviewer also works with the integrated table and votes by following the same criteria as in the first voting stage. In the third and last voting stage, we analyze the results of the second stage, but extend the evaluation to the complete contribution by further conducting an in-depth analysis of the paper going beyond the title and the abstract. The goal of this final stage is to figure out the key contributions on SME that are relevant for the in-depth analyses.

3.4.2 In-depth Analysis

In the following, we summarize the analysis procedures used to answer our research questions.

RQ 1 – Contributions over Time. To analyze which contributions were made over time, we count the contributions and aggregate the results to clusters structured according to the year of publication.

RQ 2 – Research Type Facets. We classify the contributions according to the classification scheme proposed by Wieringa et al. [35] and further applied by Peterson et al. [25] to the context of systematic mapping studies. We refer to the same structure as proposed by Peterson et al. [25]. We classify the papers strictly according to the criteria given by the scheme while allocating literature reviews and further contributions in which existing concepts of SME are re-organized, structured, and/or re-classified to the category *philosophical papers*.

3.5 Validity Procedures

To increase the validity of our study, we refer to two particular procedures. First, we analyze the area of investigation in advance and conduct a snow-balling procedure to infer and iteratively re-adjust the search strings. This increases the construct and the external validity, since we perform our analysis on a small, but representative set of publications. Second, we refer to researcher triangulation within a rigorous multi-staged voting procedure (Sect. 3.4.1) and during the classification of the contributions according to the research type facets. The voting procedure allows us to select the relevant papers from the irrelevant ones and to classify them appropriately. This procedure is accompanied by an in-depth analysis of the contents of the papers going beyond the abstracts, which we see as necessary as SME still remains a multi-facetted area with various interpretations in the provided concepts and the used terminology.

4. RESULTS

In the following, we summarize our results and structure them according to the research questions. For each result set, we conclude with a short interpretation. Table 2 summarizes the set of the papers resulting from the collection and preparation phases. We summarize the databases, the total number of results, the cleaned number of results after removing duplicates, and after the multi-staged voting of the papers for their relevance. Due to space limitations, the reference section of this short paper only lists the contributions used in the following discussion.

Table 2: Summarized search results.

Database	Total	Clean	Voting	Relevant
ACM	210	210	44	14
Springer	60	60	18	18
IEEE	210	210	22	11
Wiley	1120	381	43	5
Elsevier	50	50	23	12
Misc				4
$\sum$	1650	911	150	64

4.1 Contributions over Time (RQ 1)

To answer research question 1 we analyze the contributions classified in Tab. 2 as relevant for the years of publication. Figure 1 (upper part) shows the plot of the distribution from 1985 until 2012. The plot shows the first contribution by Basili and Rombach [3] in 1987, a first peak in the time slot 1995-1998, a second peak in 2001 that is followed by a stable number of contributions, and, finally, a third peak in 2007. The first peak is represented by the initial contributions, e.g., of Brinkkemper or Hamsen. The second peak is mainly given by contributions of Ralyté et al. (e.g., [30, 28]), and contributions that discuss opportunities for particular applications of SME concepts, e.g., the ones of [17, 2, 8]. We also find contributions that directly discussed the emerging agile methods and how SME could serve agile development (e.g., [12, 20]). The third and largest peak was in 2007 and points to a stronger effort of the SME community, reflected in publications like [1, 9, 24]), while we could find no publication at all in 2011 and so far in 2012.

Interpretation. We interpret the series of publications over time as follows: The initiation of first discussions about the need for adoption of software processes seems to be triggered by Basili and Rombach in 1987. Seven years later, Brinkkemper initiated first conceptual contributions on this area in 1995 and baptized the research area with the term *method engineering*. His conceptual work was followed by Harmsen's contributions related to his PhD work. The next serious peak in the number of publications is in 2001, which seems to be triggered by the release of the agile manifesto being signed the same year. Also, the release of the *OPEN process framework* [10, 14] as well as the release of the *Software Engineering Metamodel for Development Methodologies* (SEMDM, ISO 24744 [19, 9]) further seems to have fostered further discussions and work in the area of method engineering summarized in a joint proceeding [29], after which, finally, the number of publications abruptly ended indicating the problem domain to be sufficiently explored and practically applicable concepts to be disseminated.

4.2 Research Type Facets (RQ 2)

To answer research question 2, we create, as a first step, a tag cloud to get a first impression about the concepts and terms most frequently used. As a second step, we create a map of the number of contributions over time and their classification according to the research type facets (see Fig. 1, lower part). The tag cloud, for reasons of space limitations not included in the paper, reveals the terms *study* and *practice* to be encountered 14 times each, the term *approach* 38 times, and the term *propose* 35 times.

In the contribution set were, however, only three experience papers, two evaluation papers, four validation papers, and one opinion paper. We classified the majority of the contributions as philosophical papers (18 papers) or solution proposals (36).

Interpretation. During the classification of the publications according to the research type facets, we expected the terms *study*, *practice*, and *experiences* to indicate to a certain evolution of the concepts also reflected in the research type facets, i.e. we first expect opinion papers, followed by solution proposals, validation and evaluation research, and finally experience reports. This expectation furthermore was manifested by the last peak in the publications shown in Fig. 1 after which only few contributions were made, thus, suggesting an increased knowledge about the practical suitability of the proposed concepts made during validation and evaluation research and the dissemination of the concepts into practice.

The result set, however, does not contain the amount of empirical studies we would expect. Our classification reveals that most of the work remained to be of conceptual and philosophical nature where existing concepts were discussed from different angles. Starting from 2001—about 5 years after the first conceptual contributions were made by Brinkkemper and Harmsen—we see a large number of solution papers in which same or similar concepts are transferred to different domains, e.g., for the domain of agile methods [12] or requirements engineering [6]. Further contributions "rethink" the notion of SME [28, 32] or propose the implementation of SME concepts into software process metamodels [12, 9], but still remain without empirical evidence on the feasibility of the contributed concepts.

Even those contributions on the evaluation of SME [18, 13, 31, 15] do not provide any empirical evaluation going beyond philosophical discussions or, to some extent, isolated controlled experiments that are not generalizable nor representative due to the sensitive context in which software processes are applied. Therefore, our conclusion is that the research area remains at the level of solution proposals.

5. CONCLUSION

This paper closes a gap in the SME literature by conducting a systematic mapping study to develop a notion of the state-of-the-art in SME without going into conceptual, methodical, and technical details and discussions. Our results already show that SME is still an emerging field with many ideas and concepts competing for the favor of process engineers and process users. However, our data also shows missing empirical evidence on the feasibility of SME. Among 64 rigorously selected papers, only 9 papers could be classified as experience reports (3), validation research (4), or evaluation research (2). The majority of the analyzed contributions was classified as papers of philosophical nature (18 out of 64) or as pure solution proposal (36 out of 64).

Relation to existing Evidence. Existing studies by Hofstede and Verhof [18], Rolland [31], and Henderson-Sellers

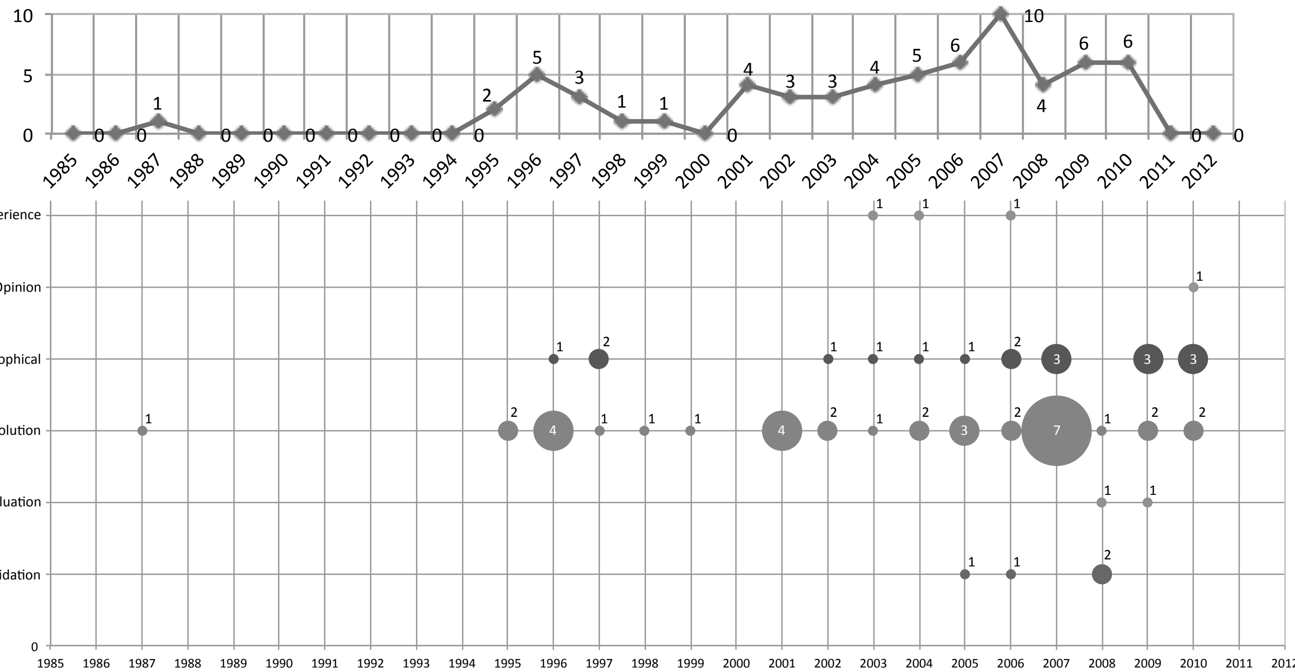

Figure 1: Number of contributions over time and structured according to research type facets.

and Ralyté [15] focus on collecting and summarizing SME knowledge. The systematization in this study shows a gap between the contributors' self-conception and the classification that is based on the research type facets.

The study at hands does not aim at collecting detailed information about SME concepts and techniques. In contrast to the existing studies, the study at hands aims at providing a first systematization of the SME domain. Therefore, this study combines techniques for conducting mapping studies and systematic literature reviews to investigate the entire domain and to systematize the SME publication flora in general.

Impact/Implications. At first, variability in software processes—in terms of design, implementation, and management as well as of their project-specific tailoring—remains an open issue. Although the agile community propagates minimalistic approaches, practice shows that there is a need for rich and structured software processes. Method Engineering is a promising approach to allow for more flexibility over the entire software process life cycle. Furthermore, enactment of software processes [22] remains an only partially solved issue for which SME provides promising concepts. Our research shows the need for further discussions on SME, especially the need for building a knowledge base comprising concrete experiences. When developing/refining SME concepts, a sound data/knowledge base is required to align SME with concrete requirements.

Second, our research also shows that although SME concepts were contributed to standards for software process metamodels, those SME parts stayed, to a large extent, neglected. The SME community thus needs to foster a critical discourse on the appropriateness of the initial SME concepts and whether there is potential for improvement to highlight the advantages.

Limitations. This paper aims at creating a first big picture and, thus, has some limitations. Deeper insights and analyses w.r.t. conceptual, methodical, and technical aspects of SME are not part of this study. Furthermore, this study does not aim at creating taxonomies or generalized concepts. This study does also not provide any solution or improvement proposal, as the scope was to systematize the SME publication flora and to illustrate a picture of the state-of-the-art of the SME domain.

Future Work. The findings of this study show the need for further investigation: Although existing studies collected much knowledge in the area of SME and several standards (partially) implement SME concepts, an agreed taxonomy, and consequently an agreed modeling approach, of SME is still missing. Especially concepts such as artifact orientation [23] are, if at all, only rudimentary noted. Further research needs to construct a sound theory of Method Engineering w.r.t. state-of-the-art concepts in Software Engineering[1] and synthesize those concepts with method engineering, e.g., by implementing an *artifact-oriented method engineering* in software process metamodels such as SPEM. Based on such a theory, we can evidently evaluate the feasibility—in software process design as well as in software process use—and provide an empirically sound basis to conduct proper validation research and infer valid and relevant improvement goals.

Acknowledgment

We want to thank Olena Stute for her work on the initial investigation to figure out the key contributions on SME. We also want to thank Georg Kalus for fruitful discussions on previous versions of this paper and for the support during the analyses of the publications.

[1]For instance the SEMAT initiative: `http://semat.org`

6. REFERENCES

[1] P. J. Ågerfalk, S. Brinkkemper, C. Gonzalez-Perez, B. Henderson-Sellers, F. Karlsson, S. Kelly, and J. Ralyté. Modularization Constructs in Method Engineering: Towards Common Ground? In *IFIP WG 8.1 Working Conference*, 2007.

[2] M. N. Aydin and F. Harmsen. Making a Method Work for a Project Situation in the Context of CMM. In *Proceedings of 4th PROFES*, 2002.

[3] V. R. Basili and H. D. Rombach. Tailoring the software process to project goals and environments. In *Proceedings of 9th ICSE*, 1987.

[4] S. Brinkkemper. Method engineering: engineering of information systems development methods and tools. *Information and Software Technology*, 38(4), 1996.

[5] Brinkkemper, S. and Saeki, M. Meta-modelling based assembly techniques for situational method engineering. *Information Systems*, 1999.

[6] C. Coulin, D. Zowghi, and A.-E.-K. Sahraoui. A situational method engineering approach to requirements elicitation workshops in the software development process. *Software Process: Improvement and Practice*, 11(5), 2006.

[7] E. Dominguez and M. A. Zapata. Noesis: Towards a situational method engineering technique. *Information Systems*, 32(2), 2007.

[8] B. Fitzgerald, N. L. Russo, and T. O'Kane. Software development method tailoring at Motorola. *Communications of the ACM*, 46(4), 2003.

[9] Gonzalez-Perez, C. Supporting Situational Method Engineering with ISO/IEC 24744 and the Work Product Pool Approach. In *IFIP WG 8.1 Working Conference*, 2007.

[10] Graham, I., Henderson-Sellers, B., and Younessi, H. *The OPEN Process Specification*. Addison-Wesl., 1997.

[11] A. F. Harmsen. *Situational Method Engineering*. PhD thesis, Universiteit Twente, 1997.

[12] B. Henderson-Sellers. Method engineering for OO systems development. *Communications of the ACM*, 46(10):73, 2003.

[13] B. Henderson-Sellers, C. Gonzalez-Perez, and J. Ralyte. Comparison of Method Chunks and Method Fragments for Situational Method Engineering. In *19th Australian Conference on Software Engineering*, 2008.

[14] B. Henderson-Sellers, C. Gonzalez-Perez, M. K. Serour, and D. G. Firesmith. Method engineering and COTS evaluation. In *ACM SIGSOFT Software Engineering Notes*. ACM, 2005.

[15] B. Henderson-Sellers and J. Ralyte. Situational method Engineering: State-of-the-Art Review. *Journal of Universal Computer Science*, 2010.

[16] Henderson-Sellers, B. Method engineering: Theory and practice. In *Information Systems Technology and its Applications*, 2006.

[17] S. Henninger and K. Baumgarten. A Case-Based Approach to Tailoring Software Processes. In *Proceedings of 4th ICCBR*, 2001.

[18] A. Hofstede and T. Verhoef. On the Feasibility of Situational Method Engineering. *Information Systems*, 1997.

[19] ISO/IEC JTC 1, SC 7. Software Engineering – Metamodel for Development Methodologies. Technical Report ISO/IEC 24744:2007, ISO, 2007.

[20] F. Keenan. Agile process tailoring and problem analysis (APTLY). In *Proceedings of 26th ICSE*, 2004.

[21] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman. Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*, 51(1), 2009.

[22] M. Kuhrmann, G. Kalus, and M. Then. The Process Enactment Tool Framework – Transformation of Software Process Models to Prepare Enactment. *Science of Computer Programming*, 2012.

[23] D. Mendez Fernandez, B. Penzenstadler, M. Kuhrmann, and M. Broy. A Meta Model for Artefact-Orientation: Fundamentals and Lessons Learned in Requirements Engineering. In *Proceedings of 13th Models*, 2010.

[24] Y.-R. Nehan and R. Deneckere. Component-based Situational Methods. In *IFIP WG 8.1 Working Conference*, 2007.

[25] Peterson, K. and Feldt, R. and Mujtaba, S. and Mattsson, M. Systematic mapping studies in software engineering. In *Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering*, pages 68–77, 2008.

[26] V. Plihon. MENTOR: An Environment Supporting the Construction of Methods. In *Asia Pacific Software Engineering Conference*, 1996.

[27] T. Punter and K. Lemmen. The MEMA-model: towards a new approach for Method Engineering. *Information and Software Technology*, 1996.

[28] J. Ralyte. Requirements Definition for the Situational Method Engineering. In *Working Conference on Engineering Information Systems in the Internet Context*, 2002.

[29] J. Ralyté, S. Brinkkemper, and B. Henderson-Sellers, editors. *Situational Method Engineering: Fundamentals and Experiences*. Springer, 2007.

[30] J. Ralyte and C. Rolland. An Assembly Process Model for Method Engineering. In *Advanced Information Systems Engineering*, 2001.

[31] C. Rolland. Method Engineering: State-of-the-Art Survey and Research Proposal. In *Conference on New Trends in Software Methodologies, Tools and Techniques*. IOS Press, 2009.

[32] C. Rolland. Method engineering: towards methods as services. *Software Process: Improvement and Practice*, 14(3), 2009.

[33] I. Ruiz-Rube, J. M. Dodero, M. Palomo-Duarte, M. Ruiz, and D. Gawn. Uses and Applications of SPEM Process Models. A Systematic Mapping Study. *Journal of Software Maintenance and Evolution: Research and Practice*, 1(32), 2012.

[34] P. Runeson and M. Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 2009.

[35] R. Wieringa, N. Maiden, N. Mead, and R. Colette. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. *Requirements Engineering*, 11(1):102–107, 2005.