A Flexible and Pragmatic Requirements Engineering Framework for SME

Thomas Olsson[a], Joerg Doerr[a], Tom Koenig[a] and Michael Ehresmann [b]

[a] Fraunhofer Institute for Experimental Software Engineering, Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany

[b] insiders technologies, Brüsseler Strasse 1, 67657 Kaiserslautern, Germany

Abstract

Convincing organizations to improve their software development processes is difficult. It becomes even more difficult for smaller firms, as they have smaller margins for improvement work and most likely lack the in-house competence to make improvements. Based on the experience from smaller and medium-sized companies, a flexible and pragmatic process improvement framework for the requirements engineering process was developed, specifically focused on the situations of smaller firms. The background and motivation is presented, including why existing methods such as CMM are not appropriate in our context. Initial experience and initial evaluation of the developed framework are also presented. It can be concluded that the developed framework has gained acceptance by our evaluation partners and that we were able to perform a process improvement with a relatively small amount of effort.

Keywords: Adaptation and configuration of RE methods, Selection of RE methods, techniques and tools

1 Introduction

Convincing a software developing organization to improve their requirements practices is usually difficult (Hall et al., 2002; Juristo et al., 2002; Rainer et al., 2003). In smaller organizations, this difficulty takes on a further dimension. Not only do you have to face the fact that people are skeptical of requirements, these companies often have a less formal development process, less distinct roles and a smaller overhead to allow process improvement (Kamsties et al., 1998). The introduction of Agile methods, such as Extreme Programming (Beck, 2000), have also meant that there is less focus on the early steps of the development.

This paper presents an initiative to provide a pragmatic requirements engineering process framework for small- and medium-sized enterprises (SME). Even though it might at first seem that SME are a homogenous group of companies, this is far from the truth. Smaller firms are often forced to comply with the processes of bigger companies they deliver their solutions to. Also, the domains in which SME operate range from web portal development to safety-critical medical equipment. But what is common to all SME is the need for small and to-the-point solutions, as they do not have the same possibilities to try out and invest money in process improvement.

This paper is organized as follows: The background and requirements for the developed framework is presented in Section 2. In Section 3, other relevant requirements process improvement approaches are presented and discussed. Our framework is presented in Section 4, and in Section 5 our early experience in applying it is presented. Future work and a general discussion are found in Section 6, and the paper is concluded in Section 7.

2 Background

Large process models like Rational Unified Process (RUP) (Kruchten, 1999) or V-Model (Sommerville, 2001) are successfully implemented in many companies. The understanding and tailoring of such models is, however, difficult and demands a lot of experience. SME often lack this knowledge (Nikula et al., 2000; Hall et al., 2002; El Emam et al., 1995). Furthermore, they do not want to employ a consultant with the required knowledge because the cost and structural changes of implementing a large standard process model are not considered justified in their context and there is a lack of faith in the consultants (Paulk et al., 1995). Furthermore, these models focus on the overall process, and are not detailed enough to help a company with specific requirements engineering problems. Solutions cannot be found, with a reasonable amount of effort, with the limited software engineering knowledge of an SME.

There is a gap between the theoretical application and tailoring of standard frameworks and the ability and acceptance to apply them in real-life projects at an SME (Nikula et al., 2000). This is a problem, because the expectations on product quality are as high in SME as in other companies. Most SME have very close relationships with their customers and are often forced to react to customer requests immediately. SME have to develop these individual customer requirements, as opposed to developing a standard software product. The risks are often increased as the economic survival of the enterprise can depend on the success of a single project. One lost or cancelled project can be the ruin for the SME.

There is a trend among SME towards Agile methods, which are easy to understand and have proven to be practical in their context. Especially the concept of practices, which are easy to understand and intuitive parts that fit well together, makes Extreme Programming (Beck, 2000) attractive to SME. Even though Agile methods can be very helpful, without a systematic improvement process, inappropriate changes can take place. Typically, no assessment is done of the current situation and the applicability of techniques and methods is not evaluated. Hence, the typical situation is that an SME has both an ad hoc development process and a process improvement process (Kamsties et al., 1998; Nikula et al. 2000).

There is a need for a lightweight framework that is easy to understand and limits the scope of change to the particular problems. It is necessary to have a small set of practices that can be introduced in a single step. The scope of change needs to be clear so that it can easily be motivated and explained to the SME. The improvements should lead to a simple process adaptation, understood and accepted by all affected stakeholders (Kaindl et al., 2002).

3 Previous Work

There exist several requirements engineering (process) frameworks, focusing on different aspects of process improvement. The “good practice guide” by Sommerville and Sawyer (1997) gives pragmatic guidelines on how to improve the requirements process. The Open Process Framwork (OPF) provides a comprehensive RE process modelling tool (Firesmith, 2005). The R-CMM initiative details an assessment framework, similar to CMM, for RE (Beecham et al., 2005).

The pragmatic framework by Sommerville and Sawyer (1997) is focused on providing concrete and simple assessment and improvement suggestions. It consists of a list of good practices for RE activities. The practices are used for assessment as well

as to give improvement suggestions. The practices are categorized into basic, intermediate and advanced. It has been applied and evaluated in several case studies and proven to be useful (Sommerville et al., 2005). However, even though the framework has been shown to be useful, some shortcomings were also identified. The framework was originally developed for the safety-critical domain. It was noted that adaptation to different domains is necessary but is currently lacking (Sommerville et al., 2005). Hence, a practice might be basic in one domain and intermediate in another and so on. Furthermore, the cost-benefit model, with its eight levels, was perceived as too complex. A simpler model could suffice.

The OPF provides a rather comprehensive framework for RE process modelling (Firesmith, 2005). It provides a framework with modelling components (e.g., endeavours, languages, processes, stages, etc), list of OPF tools and process notations suggestion. Completeness and flexibility are two important goals of the framework. To achieve the former, a rather large repository is developed and documented with many components. To achieve the latter, a process meta-model underlies the framework. Associated with the framework is an improvement process containing 14 steps (Firesmith, 2004). This rather thorough process does not prescribe the use of OPF, but is built around it.

The R-CMM is focused on assessment of the requirements capabilities of software developing companies (Beecham et al., 2005). Through a series of studies, ranging from assessing typical weaknesses of the RE process, through a development of the components of the different levels and empirical evaluations of the selection and positioning of individual practices (Beecham et al., 2005), the development of a well-founded RE assessment framework is well underway. As the name suggests, R-CMM is a refinement of CMM with respect to requirements. The goal is to provide a process improvement tool by allowing assessments specifically for RE.

4 The Framework

The framework is developed to support SME in their improvements of the requirements process. As mentioned in Section 2, such a framework has to be easily available to the firms without much investment, neither in time nor in money. It is also important that it is easily understandable to all stakeholders, not just to the person responsible for the development process.

4.1 Background

The list of good practices by Sommerville and Sawyer (1997) was a starting point for the framework presented in this paper. It provides a pragmatic framework with many of the features we were looking for. However, as we were developing our framework, we quickly ended up discussing the list of practices, the definition of a practice and how to extend the ideas to support more usage contexts. The problem we struggled with was how to separate a general practice from a concrete solution. As we want to be able to give very concrete improvement suggestions to the SME, this is central to our framework. Furthermore, the good practice guide was developed focusing on the safety-critical domain, which was not our scope.

Cost, suitability and availability are central to an SME. The OPF (Firesmith, 2005) fulfils the first and third criteria, as it is free of charge and is available online. However, in our discussions with our application partners, the criticism of OPF was

that even though the results from a tailoring effort using OPF might be useful, the framework, which is quite comprehensive, is very difficult to comprehend and would require a large amount of effort to apply. Furthermore, as the OPF is large, the psychological factor of being faced with something that complex intimidates the practitioners.

Looking at formal assessment models, such as CMM (Paulk et al., 1995) or R-CMM (Beecham et al., 2005), a well-founded assessment of the current situation at a company is possible. However, they often fall short of providing pragmatic guidelines on how to proceed after the assessment and anything close to a formal assessment is intimidating for a SME, as the cost and internal acceptance are often low.

As the available and possible frameworks did not provide us with a satisfactory solution for a pragmatic RE process improvement framework for SME, we needed to improve on existing solutions. As a basis for the improvement, we selected the good practice guide, as the intentions there are similar to ours. Our main points of interest to improve are the adaptability to support more domains and to improve support for small, incremental improvements of the RE process.

4.2 Theory

The framework is divided into five phases, which are more or less standard for the requirements process:

- Requirements elicitation
- Requirements analysis
- Requirements specification
- Requirements verification and validation
- Requirements management

In each of these phases, a number of different tasks have to be performed. The phases are refined into practices and techniques. A practice defines an abstract task that should be performed in the requirements engineering process (e.g., elicit functional requirements). A technique describes how a practice can be implemented (e.g., apply Use Cases to elicit functional requirements). An overview of the different practices, arranged according to the phases, can be found in Figure 1. The relationship is depicted in Figure 2.

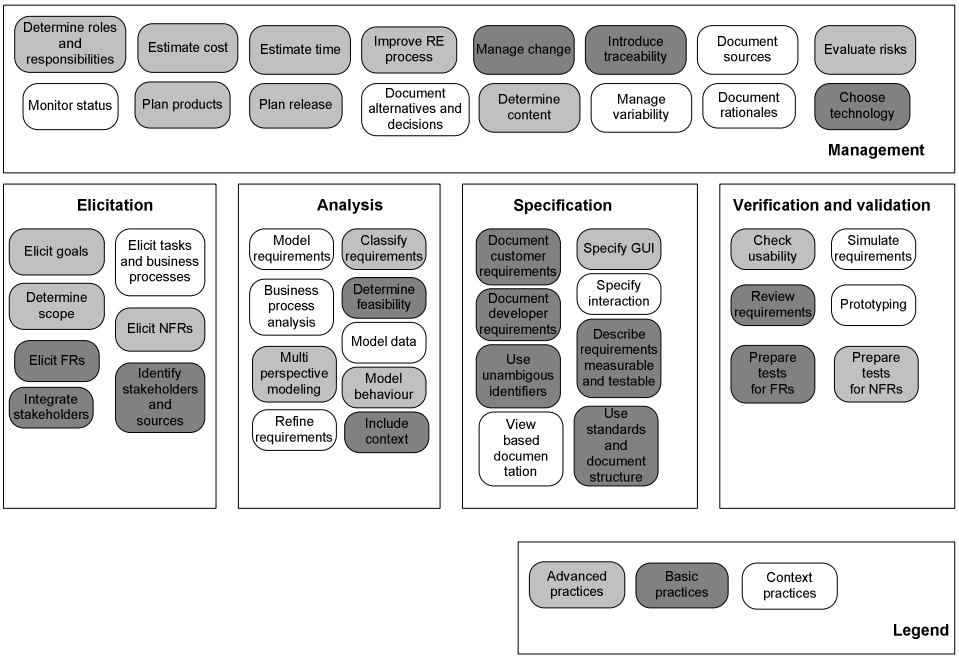

Figure 1 Overview of practices

4.2.1 Practices

A practice is usually an abstract activity (in terms of process modeling), e.g., "elicit functional requirements" which makes sense to perform in most requirements contexts. A practice can also be a principle. A principle is something that should be considered during one or more activities, e.g., "involve all stakeholders". The usage of the framework is, in principle, independent of whether a practice is an activity or principle. Hence, in order to simplify the presentation, the concepts are subsumed under the common name practices. Especially for SME, it is important to keep the presentation and usage as straight-forward as possible.

Based on experience and literature studies, we identified an initial set of practices and related techniques that are or might be of use to SME. The list is, of course, something that will always be debated. Our goal with this list is to provide a starting point from where we can refine and distill a list that receives broad support in the community.

A practice is associated with one of the five abstract requirements engineering phases. The practices are also categorized into three categories:

- Basic: represents a practice that is relevant in any requirements context.
- Advanced: represents a practice that is relevant in any requirements context, but requires other, usually basic, practices to be established.
- Context: represents a practice that is only relevant in certain project context.

As opposed to Sommerville and Sawyer (1997), we make a distinction between the concept of techniques and practices. A practice is an abstraction of the techniques

that propose a concrete solution. The motivation was to be able to associate concrete solution proposals to practices and to refine the concept of practice, in order to attain a cleaner set of practices. This was done to improve the adaptability, i.e., the applicability of the framework to more usage contexts. Also, we find, a practice is easier understood by the SME stakeholders.

Furthermore, the practices in (Sommerville et al., 1997) are categorized as Basic, Intermediate and Advanced. Basic and intermediate practices are comparable to basic and advanced practices in our framework. However, a distinction can be observed between advanced practices as defined in (Sommerville et al., 1997) and context practices in our framework. The advanced practices cover practices for the domain of safety-critical systems, while our context practices do not focus on any specific domain.

The description of a practice should be very short and as simple as possible (e.g., a PowerPoint slide or one page.). The motivation is that the effort spent reading about the different practices should be as low as possible, in order to enable the user of the framework to quickly identify relevant practices. SME usually want information to be accessible as fast as possible as they usually do not have the time to read several pages of text just in order to assess if something is relevant or not.

The template in Table 1 is used to describe a practice. The practice "Elicit non-functional requirements" is presented. It briefly describes why the activity should be applied and what can be achieved by applying it.

Table 1 Excerpt from practice template

Practice: Elicit non-functional requirements (NFR)
Goal: Elicit non-functional requirements to better support the business process and the goals of the company and the user.
Description: To capture the relevant requirements, it is not enough….
Phase: Requirements elicitation
Possible techniques to implement practice: Soft-goal notation
Required Practices: Elicit functional requirements (FR)
Supporting practices: Elicit functional requirements (FR)
Supported practices: None
Category: Advanced

4.2.2 Techniques

As opposed to practices, which represent abstract activities, techniques represent concrete solutions, describing how a certain practice can be implemented. The relation-

ship between practices and techniques is a many-to-many relationship, as a technique can support many practices. Furthermore, a practice might also be covered by more than one technique. Figure 2 shows an excerpt of these relationships for the practices "Elicit non-functional requirements (NFRs)", "Document rationale", "Elicit functional requirements (FRs)", and the techniques "Soft goals" and "Stakeholder Workshop".

The technique "Soft goals" supports the practice "Elicit NFRs" as well as the practice "Document rationales", as the rationales for the NFR are documented when using the technique. The practice "Elicit NFRs" is not only supported by the "Soft goals", but also by "Stakeholder workshop", which in turn also supports "Elicit FRs".

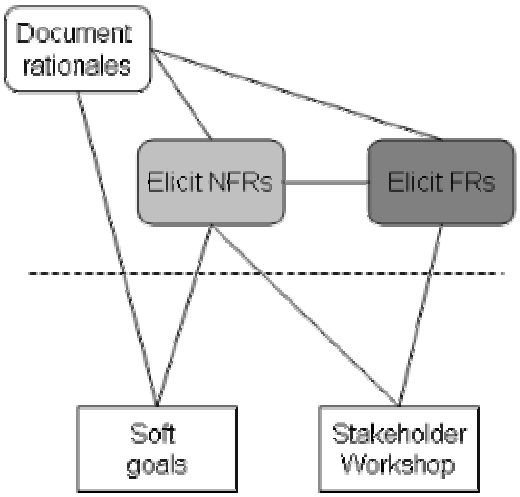

Figure 2 Relationship between practices and techniques

The practice "Elicit non-functional requirements" is presented in Table 1. As can be seen, it is not described how the practice can be performed. Therefore, techniques are introduced. They describe possible ways of implementing a certain practice, completely or partially. The techniques are described using the template in Table 2. Comparable to the description of practices, the description of a technique should also be compact. The template does not have to provide a complete guide on how to perform a certain activity, but should give a rough overview and contain all necessary information to look up the required information.

Table 2 Excerpt from technique template

Technique: Soft goals
Goal: Elicit, document and analyze non-functional requirements with regards to their relationships to functional requirements and architectural decisions.
Description: The following activities have to be executed to apply Soft goals: ...
Expert: Please contact ... for further information
Context: Not specific

Type: Method
(Partly) Accomplishes the practices:
Include stakeholder
Elicit non-functional requirements
Determine scope
...

There should be enough information in the technique description for the reader to understand if it can be the solution to their particular problem. The complete description and explanation of the technique is found in references provided.

In order to make the information contained in the templates accessible, practices and techniques are collected in a repository, also published on a web portal[1].

4.3 Framework usage

The framework can be used for different purposes. We identified three usage scenarios:

- Assess company setting - SME may use the framework as a support to assess their own requirements process. Based on the set of practices it is possible to determine how requirements engineering is performed within the firm. That is, which practices are already in place.
- Solve specific problem - In certain cases, companies want to solve a precise problem regarding the requirements process, e.g., security problems are inadequately handled. To some extent, it is possible to use the practices as a means to identify possible solutions. The title and the goal of a practice give a hint on whether a practice might be of use to assist in solving a certain problem.
- Process improvement - The framework can also be used for general process improvement of the requirements process.

As the process improvement involves the other uses of the framework, this usage is elaborated on in more detail.

1. Analyze current situation - Prior to making a decision regarding possible improvement steps, the current situation of the company should be identified in a first step. If an assessment (e.g., CMM (Paulk et al., 1995)) has already been performed, that information can be used. If this has not been the case, it is possible to broadly identify the current setting as presented in the first usage scenario.
2. Identify required practices - Based on the classification of practices (see Figure 1) the set of practices is prioritised. As a first step, based on the current setting, missing basic practices are identified. The company setting should be considered while identifying basic practices that should be introduced. It could be possible that in a certain setting, even a basic practice will not be suitable. The reason is

[1] www.re-wissen.de (in German)

that every company is different and even though attention was paid to making the classification generally applicable, there might be situations where a different prioritization is needed.

After all relevant basic practices have been identified, all relevant advanced and context practices are to be identified. With regards to context practices, a more detailed assessment of the company setting might be necessary, in order to identify practices appropriate to the company setting. The practice "manage variability", for example, is only relevant for a company that develops a set of similar products, i.e., has a product line type of situation.

After identification has been completed, the identified practices need to be prioritized. According to our experience, not more than two to four practices should be introduced at the same time. As with any process improvement, small steps and an iterative improvement are better. Exactly how many practices can be considered at the same time depends on the number of techniques needed to implement the practices.

3. Identify required techniques - As the practices identified in the previous step do not specify how a certain practice can be introduced, the set of all associated techniques is identified based on the identified practices.
4. Select and adapt a minimal set of techniques - The identified set of techniques is identified in two steps to identify and eliminate unsuitable techniques. In a first step, all techniques are assessed with respect to the overall company setting. In a second step, all techniques are examined regarding the expected input and preconditions. The remaining set of techniques is then minimized in a concluding step. As one technique may implement or contribute to more than one practice, the goal of this step is to identify a minimal number of techniques that implements all required practices.

5 Experience

The development of the framework is still in its early stages. We are in a continuous dialogue with industry to ensure that our solution is moving in the right direction.

Generally, the framework was well received by the different stakeholders of our industry partners. It was also possible for us to identify and tailor process improvement suggestions with a relatively small amount of effort. Using the practices as a mean to assess the current situation worked well. It took less than an hour to explain the concept of practices and to give a first overview of the different practices. Especially the graphical representation of the practices (see Figure 1) and the short description were appreciated. Hence, it was possible for all stakeholders to understand and discuss the current situation. In previous process improvement initiatives, the current situation was assessed by external consultants and by process owners. As all stakeholders could follow and participate in the process, the assessment was more credible to them and they accepted the deficiencies in their current way of working.

When it came to identifying and selecting techniques, the framework was perceived as intuitive and very helpful. The explicit modelling of the relationships between practices and techniques, see Figure 2, made it easy to identify techniques with high leverage suited for the context. In this particular case, we were able to identify

four techniques that would improve the 7 practices chosen for improvement (see Section 4.2.1). The principle of trying to identify the minimal set of techniques that cover all the desired practices proved suitable and if nothing else, has the psychological effect of being efficient, enhancing the acceptance at the firm. As mentioned in Section 2, a key requirement for the framework was to have a good balance between cost and benefit. The fact that the SME feels good about the improvement is a key success factor.

Another key psychological factor was the perception that there was no traditional process improvement as such. Rather, techniques solving a concrete problem were suggested, making the stakeholders see the benefit of introducing it. The process improvement, hence, was implicit, increasing the feel good factor, which is so important in all process improvement situations.

The aim is that the framework should empower the SME to participate and to perform many of the decisions themselves. The decision on which techniques to recommend, however, was made by requirements specialists and process owners. The reason is that there is a lack of empirical data supporting this decision. The suitability of specific techniques is judged by experts based on their experience, as this data was missing. This was perceived as a major drawback.

Even though the initial experience was positive, more evaluation is necessary. Future case studies are planned, both to evaluate the framework and to gather empirical data in individual techniques.

6 Discussion

The framework presented here tries to provide a comprehensive approach for improving the requirements process. It incorporates improvement suggestions for the requirements process and ease decision making by providing experience data on the improvement suggestions. Our motivation for coming up with a new framework comes from working with small and medium sized enterprises, where existing approaches and methods are not sufficient for their project and product needs.

An important internal property as we are developing our framework is the balance of sound scientific basis and a pragmatic and lean solution. We hope that, by introducing the concept of abstract practices and concrete techniques, we achieve a good compromise.

The list of practices is, however, important. If this list is not appropriate, then the abstraction is meaningless. This list is also something that different experts always argue over. Our plan is to attack these problems in two ways: By applying the framework in the firms of our partners to get feedback from them, and by letting requirements experts prioritize and optimize the list.

The list of techniques is much less troublesome. This list is a collection of existing methods and approaches. Of course, if the aim is to claim completeness, then the challenge grows. But initially we do not aim at having a complete list of techniques. We see this as a basis that will grow as we use the framework and perform process improvement in industry.

A central problem in all process improvement is which technique to recommend to a company. Our framework provides an abstraction mechanism through the practices, which helps. This is not the whole answer though. An open issue that we aim to tackle is how to incorporate empirical studies of techniques into our framework. Our

hope is by having concrete data on the performance of specific techniques, the selection of what to implement in a specific improvement context will be supported.

To further improve the adaptability of our framework, we aim to investigate how to describe different usage contexts. As noted in (Sommerville et al., 2005), a basic practice in one context might be advanced in another. Our framework does generalize when the practices are categorized into basic, advanced and context types. Hence, the open issue is how to describe the context of use and how the categorization of practices can be improved. We believe that it is unrealistic to completely describe any usage context. The question is: Which factors are the determining ones? Furthermore, is it possible to say that a practice is basic in all contexts, or are, in fact, all practices context dependent? Our hypothesis is that there are practices that are important in all contexts. But this is something we need to investigate further.

The continued work on the framework will have a strong empirical character. In order to further validate and improve the framework, further case studies in industry are necessary. It is also essential that key empirical data on techniques are gathered and documented, as at the end of the day, any improvement will depend on the identification of appropriate change proposals to the development process.

7 Conclusion

In an attempt to support small and medium-sized enterprises (SME) in their requirements process improvement, we have developed a framework. We want to provide an easy to understand and practical (in the sense that concrete improvement suggestions are delivered) framework, which still has a basis in sound requirements engineering.

Based on the experience both from consulting and from being part of the development, many of the existing frameworks and assessment methods are judged to be too big and cumbersome to use, while they often lack the necessary details for the requirements process.

The core of the framework is practices and techniques. The former is an abstract activity or principle that makes sense to include in the requirements process. A technique implements, at least partly, one or more practices.

Our initial evaluation together with several SME and research institutes shows promise for the framework. The companies could easily understand the framework and we were, with a small amount of effort, able to identify and tailor process improvement suggestions for the SME. The involved companies also reported that due to the intuitive nature of the framework, the developers could themselves understand the need for improvement and follow the arguments for the improvement suggestions, giving that improvement suggestions greater acceptance.

Future work involves further evaluation of the framework. Several case studies are planned, on the one hand to evaluate the framework as such, on the other hand to gather empirical data regarding requirements techniques.

Acknowledgements

This work is funded by the German Ministry for Education and Research (Bundesministerium für Bildung und Forschung, BMBF), under the grant for the ReqMan project (grant number: 01 IS C02 D). We would like to thank the other members of the ReqMan project as well as the colleagues in the Requirements and Usability department at Fraunhofer IESE and insiders technologies.

References

AAEN I (2003) Software Process Improvement: Blueprints versus Recipes. *IEEE Software* 20(5), 86-93.

BEECHAM S, HALL T, BRITTON C, COTTEE M and RAINER A (2005) Using an expert panel to validate a requirements process improvement model. *Journal of System and Software* 76(3), 251-275.

BECK K (2000) *Extreme Programming Explained.* Addison-Wesley

FIRESMITH D (2005), Open Process Framework (OPF), www.donald-firesmith.com, last visited 2005-07-20.

FIRESMITH D (2004) Creating a Project-Specific Requirements Engineering Process. *Journal of Object Technology*, 3(5).

EL EMAM KE and MADHAVJU NH (1995) A field study of requirements engineering practices in information systems development. In *Proceedings of the International Symposium on Requirements Engineering*.

HALL T, BEECHAM S and RAINER A (2002) Requirements problems in twelve software companies: An empirical analysis. *IEE Proceedings Software*, 149(5), 153-160.

JURISTO N, MORENO AM and SILVA A (2002) Is the European Industry Moving towards Solving Requirements Problems. *IEEE Software*, 19(6), 70-77.

KAINDL H, BRINKKEMPER S, BUBENKO J, FARBEY B, GREENSPAN S, et al. (2002) Requirements Engineering and Technology Transfer: Obstacles, Incentives and Improvement Agenda. *Requirements Engineering*, 7(3), 113-123.

KAMSTIES E, HÖRMANN K and SCHLICH M (1998) Requirements Engineering in Small and Medium Enterprises. *Requirements Engineering*, 3(2), 84-90.

KRUCHTEN P (1999) *The Rational Unified Process, An Introduction.* Addison Wesley.

NIKULA U, SAJANIEMI J and KÄLVIÄINEN H (2000) Management View on Current Requirements Engineering Practices in Small and Medium Enterprises. In *Proceedings of The Australian Workshop on Requirements Engineering*.

PAULK MC, WEBER CV, CURTIS B and CHRISSIS MB (1995) *The Capability Maturity Model: Guidelines for Improving the Software Process*. Addison-Wesley.

RAINER A, HALL T and BADDOO N (2003) Persuading developers to 'buy into' software process improvement: Local opinion and empirical evidence. In *Proceedings of the International Symposium on Empirical Software Engineering*.

SOMMERVILLE I (2001) *Software Engineering.* 6th edition, Addison-Wesley.

SOMMERVILLE I and RANSOM J (2005) An Empirical Study of Industrial Requirements Engineering Process Assessment and Improvement. *ACM Transactions on Software Engineering and Methodology*, 13(1), 85-117.

SOMMERVILLE I and SAWYER P (1997) *Requirements Engineering: A Good Practice Guide*. Wiley.