PRO-REQ: a facilitator guide to implement CMMI-Dev requirements engineering and management areas

Alfraino Souza Diniz [1], Rosely Sanches[1], Rosana T. Vaccare Braga[1]

[1] Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo (ICMC-USP)
Postal Code 668 – 13560-970 – São Carlos – SP – Brazil

alfrainodiniz@gmail.com, {rsanches, rtvb}@icmc.usp.br

Abstract. *Requirements engineering processes are among the major sources of problems found during software development. A way of reducing these problems is to introduce maturity models like CMMI, but they are more likely to define what has to be done, instead of how it should be done. This paper presents a guide whose goal is to facilitate the implementation of requirements engineering processes in organizations that cannot afford their expensive costs. The guide bases on CMMI practices, but also considers several other sources of good practices on requirements engineering and management. The results of applying it in a small Brazilian software development organization are also presented.*

1. Introduction

Over the last twenty-five years, software requirements are being repeatedly recognized as a real problem in the software development process [Lamsweerde, 2000]. Requirements engineering and management processes have a deep impact on system costs and functionalities, but yet a great number of organizations have these processes badly defined or even non-defined [Sommerville & Ramson, 2005]

Aiming at finding solutions to the mentioned problems, several efforts are being done. Guides containing good requirements practices were created, such as the Requirements Engineering - Good Practice Guide, by Sommerville and Sawyer [Sommerville & Sawyer, 1997]. Other authors presented standards for requirement processes models [Hagge & Lappe, 2004], frameworks to help generating and deploying processes [Jiang et al, 2004], as well as the famous maturity processes based on the capability of the deployed processes, such as CMMI [SEI, 2006], and process reference models such as the international standard ISO/IEC 12207 [ISO 2001, 2004], among others. The latter two models mentioned are focused not only in the requirements process enhancement, but also in the software development as a whole. There are also in the Requirements Engineering literature several other techniques, methods and frameworks to support the process.

Even with such a great amount of knowledge about Requirements Engineering (RE) spread in the literature, many projects fail, possibly because managers know "what" to do in the improvement process, but they do not know "how" to do it [Niazi et al, 2004]. More than that, a CMMI evaluation, without considering the consulting costs for deploying the model, may cost between US$ 40.000 and US$100.000, which can be

very expensive for most developer organizations from small to medium-size ones [Cuevas & Serrano, 2004].

The main goal of this paper is to present a reference guide that was built during a master thesis at our institution [Diniz, 2007], which is based on a set of good practices related to engineering and management requirements processes, as well as the products generated by these practices. The guide, named PROREQ, is organized to facilitate the implementation of these processes, which are adherent to CMMI-DEV model. However, the processes are presented in a detailed format, so that they can be easily used by organizations that need to improve their requirements processes, even when they do not have available the required resources to hire specialized consulting services.

The remaining of this work is organized as follows: Section 2 presents the references used as basis to identify the good practices that compose the guide, as well as related work concerning requirements engineering improvement. Section 3 presents a PROREQ and how to use it. Section 4 presents a process that resulted from using the guide in a small Brazilian software organization. Section 5 presents the conclusions and future work.

2. References used to compose the guide and related work

PROREQ contains a set of good practices that need to be selected and executed to fulfill the goals of process areas related to requirements engineering, as well as the corresponding work products generated by the their execution. The identification of these practices was done based on the study of several works regarding requirements engineering and management, as illustrated in Figure 1: ISO/IEC 12207 (2001, 2004), ISO/IEC 15504 (2003a, b, c), PMBOK (2004), SWEBOK (2004), and two books about RE [Sommerville & Sawyer, 1997; Kotonya & Sommerville, 1998].

The SWEBOK project (2004) was one of the sources used to build PROREQ. In particular, PROREQ uses concepts contained in the Software Requirements knowledge area, which concerns activities related to the requirement management process. From the PMBOK guide – Project Management Body of Knowledge (2004), PROREQ uses the concepts of project management to support requirements engineering and management process execution. CMMI-DEV [SEI, 2006] was used by PROREQ to structure the process areas related to requirements – Requirements Development and Requirements Management. Based on the process areas structure, PROREQ uses CMMI-DEV sub-practices as classifiers for good practices collected from other models, as illustrated in Sec. 3.1. The ISO/IEC 12207 standard (2001, 2004a) was another source of processes and practices used by PROREQ. In particular, it uses ISO/IEC 12207 processes from the fundamental process category, which are those related to the RE area: Requirements Elicitation, System Requirements Analysis, System Architectural Project, and Software Requirements Analysis.

The book “Requirements Engineering - A Good Practice Guide” [Sommerville & Sawyer, 1997], here simply referred to as RE-GPG, contains a set of approximately 50 good recommended practices for the requirements process, which are used in PROREQ too. From the book “Requirements Engineering – Process and Techniques” [Kotonya & Sommerville, 1998], here simply referred to as RE-PROTEC, it was extracted only the first part, in which are presented concepts related to process models,

process actors, support processes, and process improvement, as well as concepts relative to the activities of requirements elicitation, analysis, validation, and management.

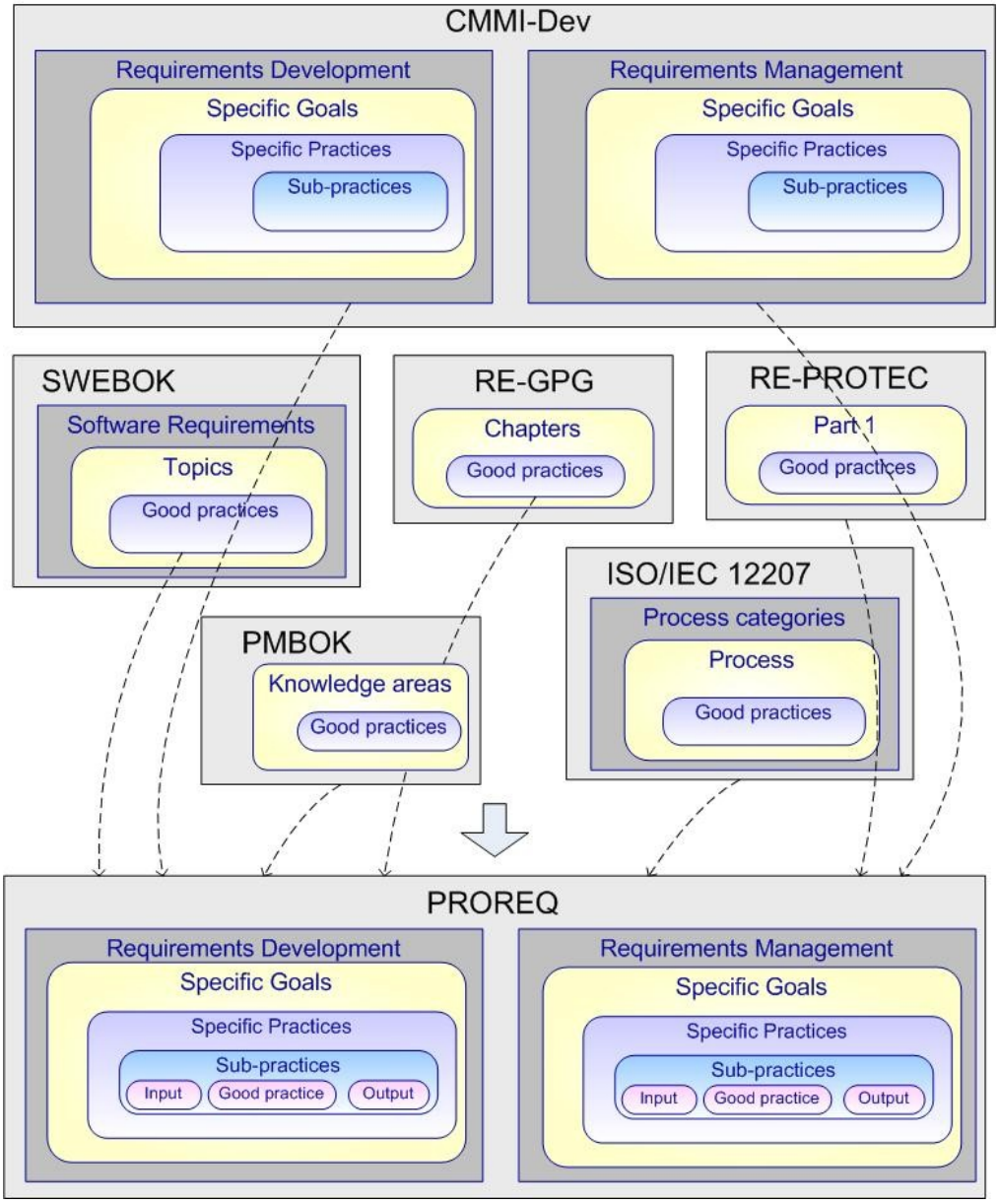

Figure 1. Structure and Sources of Fundamental Practices for the PROREQ guide

Regarding the evaluation model proposed by PROREQ, it was based mainly on ISO/IEC 15504 Parts 2 and 5 [ISO 2003, 2004b] and on MPS-BR (2006), which is a Brazilian model for software process improvement. ISO/IEC 15504 Part 2 was used as a guide to formulate the evaluation of the guide good practices, while Part 5 was used as a reference to the process evaluation model, supplying a standard structure to describe the processes, their practices, their work products, and also their expected features.

Other related works about requirements processes improvement are described next. Sommerville and Ransom (2005) describe a case study using the REGPG mentioned previously. Their study revealed that the set of good practices should be used

according to the organization context and, thus, there are no standards in the chronological order of practices application. A study conducted by Kauppinen et al. (2004) identified a set of factors that affect the success of programs for requirements processes improvement in software development organizations. These factors were used to guide the case study using REGPG. Beecham (2003) outlines the main problems found in improvement programs for twelve software development organizations. Later on, Beecham et al (2005) describe a proposal for easing the implementation of requirement management processes using CMM, in which a model named R-CMM is used to link RE processes to the maturity levels, dividing them in sub-processes that should be defined and evaluated.

3. PROREQ – A facilitator guide to implement requirements processes

In this section we present the PROREQ guide and how it can be used to introduce requirements engineering processes into an organization. PROREQ contains fundamental practices that should be used to build RE processes, as well as practices that should be used to get process improvement. According to Strauss and Corbin (1998 apud KAUPPINEN et al, 2004) the act of grouping concepts into categories is important because categories have the potential to explain and predict about the phenomena under study. Therefore, practices have been grouped into fundamental practices and organizational practices, following the schema presented by ISO/IEC 12007.

Fundamental practices are those related to the technical aspects of the requirements process, such as elicitation and analysis practices. They were collected from SWEBOK, PMBOK, RE-GPG, RE-PROTEC, and ISO/IEC 12207, and then organized according to the CMMI-DEV structure, as illustrated in Figure 1. As CMMI-DEV relies on the best practices obtained after many years of empirical studies (BEECHAM et al, 2005), the fundamental practices were organized according to its structure. The goal is to use this accumulated knowledge as a reference to the selection and priorization of the practices that a particular organization needs to achieve success in its improvement program. So, PROREQ proposes that each specific practice and corresponding sub-practices be considered when choosing the fundamental practices.

On the other hand, organizational practices intend to deal with issues that should be present in the organization in order to effectively use the fundamental practices. Also, they want to guarantee that the fundamental practices are useful and required in the organization along the time (e.g. there is support from project managers to improve processes). Another feature present in PROREQ is the controll of results expected by each process (this feature was based on the MPS.BR guide). The goal is to guide users when they are selecting the fundamental practices required for their projects success.

Finally, a relevant aspect to implement the process is the improvement implementation strategy. In PROREQ, a strategy to implement the improvements was ellaborated based on the strategy proposed by ISO/IEC 15504 and also based on organizational practices. A simplified evaluation model was presented, based on the process evaluation model proposed on ISO/IEC 15504 and on the MPS.BR evaluation model. Figure 2 illustrates all the PROREQ components and corresponding references used as basis to their construction.

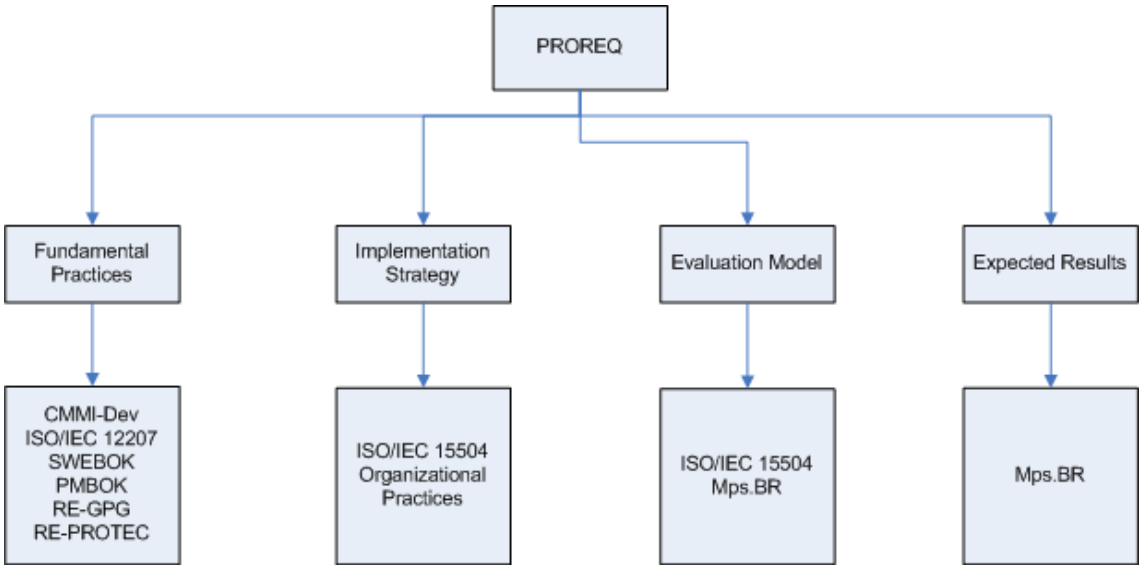

Figure 2 – PROREQ Components and reference bases

3.1. Organizing the Fundamental Practices

PROREQ fundamental practices were gathered from a set of guides and books and have the main objective of documenting the knowledge required to improve the organization processes relative to Requirements Development and Management. These practices are described in such a way that helps to decrease the abstraction level of practices proposed by CMMI-DEV. This is achieved by giving more detailed descriptions of how to fulfill the goals of its requirement process areas. As mentioned in Section 2, concepts coming from CMMI-DEV, such as specific goals, specific practices and corresponding subpractices were used to structure the guide. Additionally, another level was included, named "fundamental practices", as illustrated in Figure 3. This way, we try to keep CMMI-DEV as the main reference to define (select and prioritize) practices from the requirements processes, but at the same time decreasing the abstraction level, allowing users to know how they proceed to execute its specific practices and subpractices and, thus, fulfill the processes specific goals.

Table 1 shows an example of PROREQ fundamental practices, corresponding to a specific practice of CMMI-DEV. Table 2 exemplifies input and output work products for a few of these fundamental practices. Each work product feature represents concrete evidence that a particular practice is being executed and, consequently, the set of work products generated by a process make evident the existence of a process capability.

3.2. Implementation Strategy

In order to make good use of PROREQ, it is necessary to define a way of introducing the good practices into the organization. Several organizational practices have been collected from a study done in the literature [Beecham et al 2005, Kauppinen et al 2004, Niazi et al 2004, Cuevas et al 2004, Sommerville and Ransom 2005, Beecham et al 2003] and, also based on the ISO/IEC 15504 process improvement strategy, we have

adapted a simple improvement strategy to be used with the PROREQ guide. The proposed model is represented in Figure 4.

For each step of Figure 4 there is a set of organizational practices that should be followed. They were taken from empirical works related to requirement process improvement or software process improvement in general. Table 3 presents an example of these organizational practices for step 1 – Plan the Improvement Project.

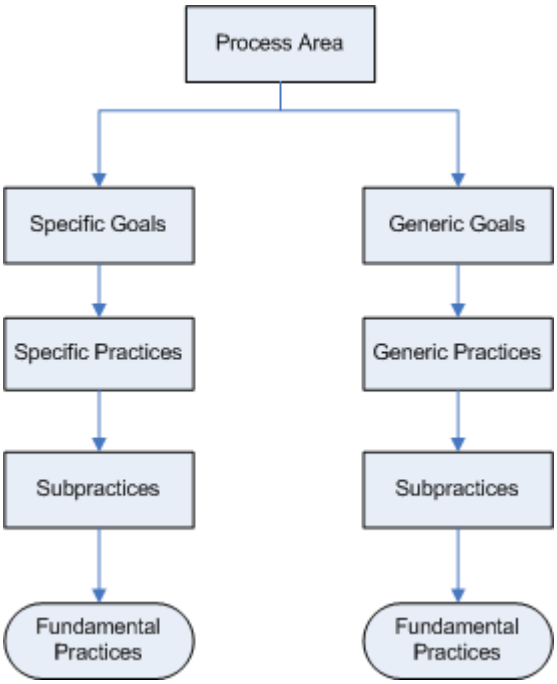

Figure 3 – Organizational Structure of PROREQ fundamental practices

Table 1. Example of PROREQ fundamental practices

Process Area	Requirements Development
Specific Goal	Develop Customer Requirements
Specific Practice	Elicitate *stakeholders* needs, expectations, restrictions, and interfaces for all product life cycle phases.
Results	
Subpractices	**Fundamental Practices**
1. Engage relevant *stakeholders* using methods to elicitate needs, expectations, restrictions, and interfaces.	Identify and consult project requirements sources
	Evaluate Financial Feasibility
	Identify and Register requirements sources
	Register rationale that lead to requirement
	Establish a continuous communication with the customer
	...
2. ...	...

Table 2. Relationship between Fundamental Practice, Input and Output features

Input Features	**Fundamental Practice**	**Output Features**
Mapping from the acquiring organizational structure and business goals or interests	Identify and Register requirements sources	List of requirements sources, documents, systems, etc.
Requirements Customer	Evaluate Financial Feasibility	Financial Feasibility Report

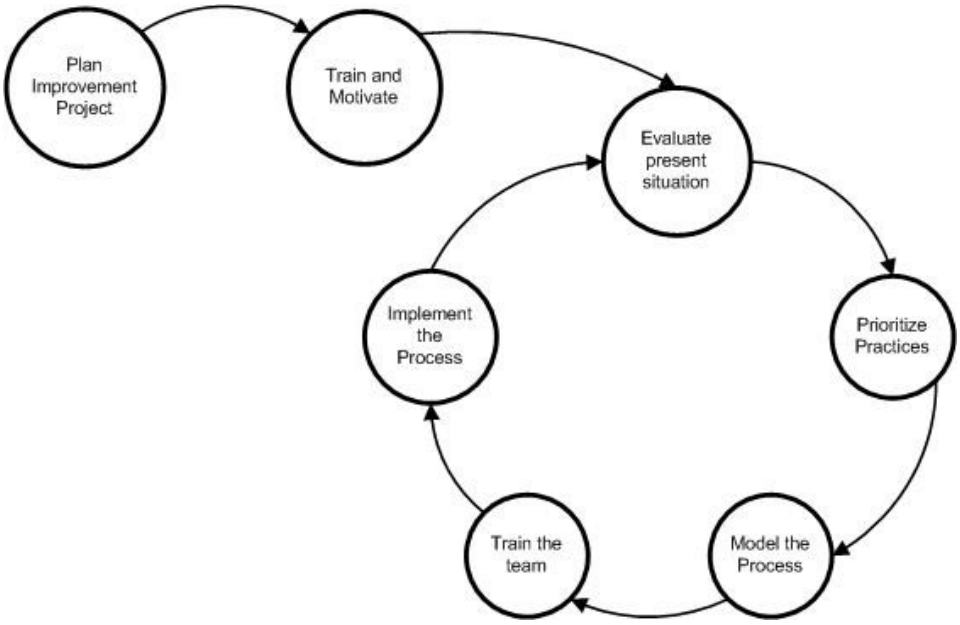

Figure 4- Strategy for implementing the improvement process suggested by PROREQ

Table 3 - Organizational Practices for the Planning Step

Step 1: Plan the Improvement Project
Consider skills, responsiveness and experience of the persons involved.
Make available both time and resources to the process improvement project.
Ellaborate a plan for the improvement project.
Consider human social and cultural factors during the planning of the process improvement program.
Use a systematic, incremental and person-oriented approach for implementing the change.
Observe technical, business, economical, and political factors in the RE process improvement project.
Develop an implementation plan.
Alocate time for the organization really change its practices.
Alocate several people to perform the change, in order to eliminate the risk of vanishing the process in case the person allocated to it leaves the organization.
Involve all user groups that use or produce products of the requirements process.

According to Figure 4, when an organization decides to use the PROREQ guide, initially it has to elaborate a planning for implementing the improvement project, where schedule, tasks, deadlines, human resources, responsibilities, measures, and a mechanism to outlook the project are defined. Motivation training is then performed with all people involved in the process, to elucidate the relevance of the requirements process, and give an overview of the process and of how it relates with the development process of the organization. The next step is to evaluate the present state of the process, using the model presented later in Section 3.3. When this step occurs in the first improvement cycle, it is useful to make users familiar with the guide, as they need to investigate the meaning of each good practice, their input and output features, so the

learning process is intense at this step, while it eases the subsequent project improvement steps. The next step concerns defining priorities for the practices according to the organization business goals, using a questionnaire like the one exemplified in Table 4.

Table 4 – Example of a question to prioritize good practices

Good Practice: Evaluate Financial Feasibility	
Input	Output
Customer Requirements	Project Financial Feasibility Report
Priority: What is the importance of this good practice and its associated work products to the organization context? ( ) Irrelevant ( ) Little Importance ( ) Important ( ) Very Important	

In order to prioritize the practices, one should consider four aspects of similar importance for the success of an organization improvement program:

• The organization project context, which involves numerous factors such as application area, techiniques known, how critical the project is, etc.

• The expected results of the process: the organization should use as an essential parameter to select its practices the expected results from the requirements process, listed on Table 5, because these results can be an evidence of whether or not the objectives are being fulfilled.

• The structure of the set of fundamental practices: the set is organized according to the CMMI-DEV process areas structure, because its process areas are related to requirements as a reference of practices that should be executed to obtain sucess in the requirement process. Thus, it is important to consider all CMMI-DEV practices and sub-practices when selecting those to be used.

• The organizational practices of the current step.

After defining the practices priority, the process is modeled by composing the work products and describing activities to produce them, based on the good practices that were selected. This activity is exemplified in Section 4. In the next step, the team is trained to learn how to use the process, and this training should enforce aspects already mentioned in the previous training, as well as consider the following issues: allocate workers with knowledge about requirements to give the training and support the process implementation; support team work; fulfill the team present needs; integrate all people that will use the requirements process to ease communication.

Finally, the use of the process inside the organization begins. Based on the work of Kauppinen et al (2004), in this step we suggest that the process is used in pilot projects before initiating the implementation in the whole organization. After the pilot projects are finished, they are evaluated again to measure their results. However, in this step only the implemented practices are evaluated. Results and problems found are used as feedback to begin another improvement cycle. This cycle is repeated as many times as necessary to achieve the organization goals.

Table 5 – Characterization of process attributes fulfillment degree

Implementation Degree	Characterization
Totally implemented (T)	• The work product is present and is considered adequate • There is a template of the work product that confirms its implementation • There is a log of the fundamental practices selected from the guide • No substantial weak point was noticed
Largely implemented (L)	• The work product is present and is considered adequate • There is a template of the work product that confirms its implementation • There is no log of the fundamental practices selected from the guide • One or more substantial weak points were noticed
Partially implemented (P)	• The work product is not present or is considered inadequate • Artefacts/assumptions suggest that some aspects of the expected result are implemented • There is no log of the fundamental practices selected from the guide • Weak points were documented
Not implemented(N)	• Any situation different from the above
Not evaluated (NE)	• The project is not on the development phase that allows to achieve the result or is not part of the scope to achieve the result
Out of Scope (O)	• The expected result is out of the evaluation scope, as documented in the evaluation plan.

3.3. Evaluating the requirements process

PROREQ aims at improving process areas regarding Requirements Development and Management. The maximmum level of improvement, in this guide, is represented by the capacity level 1 for all the existing processes. As described in MPS.BR, the process capacity is characterized by the ability of a process in achieving its business goals. This means that it fulfills the process attributes for the intended capacity level [MPS.BR, 2006]. Thus, following the pattern established by the ISO/IEC 15504 measure framework, the process must fulfill the process attribute called “PA 1.1 Process Execution Attribute” described as:

• PA 1.1 Process Execution Attribute: This attribute measures the extension in which the process proposal is fulfilled. A complete attribute fulfillment means that the defined results for the process are achieved, i.e., the development of all work products that the process must produce.

The process attributes have a set of indicators for associated process attributes, which give an indication of how much they fulfill the instantiated process attribute. These indicators can be activities, resources or results associated to the achievement of an attribute proposed by the process [ISO, 2004b].

The present work uses the classification of process attribute indicators from the MPS.BR evaluation method, named MA-MPS. This method suggests three indicators:

• Direct indicators: represent the activity goal, i.e., the main product resulting from the activity execution.

• Indirect indicators: are used to confirm that the organization is able to implement a result.

• Assertions: are obtained through interviews and/or presentations, and are used to confirm the implementation of a process, its results and attributes.

PROREQ considers work products as direct indicators, while templates of work products and logging of performed activities are considered indirect indicators. Assertions are used as execution indicators for fundamental practices that do not generate explicit work products such as, for example: “Use business interests to drive the elicitation”; to clarify doubts related to work products; or to log selected practices.

Therefore, the scale to characterize the fulfillment level of PROREQ process attributes was adapted from the logic of the MPS.BR evaluation model, to characterize the fulfillment level of the expected results, as described in Table 3.

In summary, PROREQ process attributes are composed of work products, corresponding templates, logging of practices selection, and assertions relative to other process attribute indicators. The following procedure should be observed:

• Work Products produced: to verify the appropriate production of a work product, it should be checked if all its expected characteristics are present, through the verification of fundamental practices that produce them.

• Work Products Templates: work products characteristics should reflect the result of all fundamental practices contained in the instantiated process. This means that each fundamental practice of the instantiated process should have at least one associated characteristic.

• Logging of used practices: to check the utilization of the fundamental practices, it should be also checked if there are loggings of which of them were selected from PROREQ, and if they are described in order to ease the work of process users.

• Assertions: assertions are used to raise evidence of practices utilization, mainly for practices that do not produce any characteristic directly, but that help the process execution as, for example, the fundamental practice “Use business interests to drive elicitation”.

In PROREQ, the capacity assignment is performed with focus on the project that instantiated the process. This means that each process of each project has a grade assigned to it. Therefore, using the ISO/IEC 15504 framework structure, the process will be classified with a Level 1 capacity if all its attributes belongs to the scales: L (largely implemented) or T (Totally implemented).

4. Case study

The results of applying the PROREQ guide are presented in this section. They were obtained in a small Brazilian software development organization with focus on Web development. In the initial evaluation, it was observed that the only fundamental practice regarding requirements done in the organization was “Identify business goals and interests”. Information resulting from this practice was described in a summarized

way through a work proposal for the customer. No documentation was created for requirements. Nevertheless, the results of the initial evaluation supplied data to define the first version of the organization process.

After analyzing the situation, the first version of the process was focused on adopting a template for the requirements document, as well as defining how each requirement would be described individually. In order to achieve that, the following fundamental practices were adopted:

- Define a standard structure for the requirements document;
- Define standard templates to describe requirements;
- Use a simple, consistent and concise language;
- Uniquely identify each requirement;
- Obtain business goals and interests;
- Obtain requirements through interviews;
- Use business interests to drive elicitation;
- Model the system architecture;
- Alocate requirements to architecture components.

The selection and priorization of these practices was performed using the aspects mentioned in PROREQ improvement strategy, in particular the step regarding practices priorization, described in Section 3.2. This first version of the process was modeled and used in two pilot projects, each of which with two months duration. Then, a new evaluation was done, checking only the realization of the implemented practices. Serious problems were detected due to the lack of requirements validation inspections, and also the lack of policies for managing requirements and their sources. Even with these problems, by comparing the average percentage of re-work hours from previous organization projects with those from these two projects, there was a reduction of about 30% re-work.

After collecting the results of implementation of the initial good practices, there was an increase in the motivation of the project sponsorships, so a new cycle was initiated. Data about the previous evaluation was used to guide the choice of practices to solve the mentioned problems. However, in this new cycle the situation of the organization was different, because other processes were being implemented and, thus, the requirements processes areas were embedded in those processes.

The result of this new cycle was the implementation of practices, grouped in process activities, as exemplified in Table 6. Activities were divided in two processes that are particular to the organization, named “Sell” and “Requirements Engineering”. Five requirements activities were created, each of which dealing with a set of practices and, consequently, generating a set of work products. Figures 5 to 9 list each of the resulting activities, containing their good practices and work products.

In this new cycle, two pilot projects are being done. One of them has already finished and dealt only with the requirements engineering of a bigger project. In this

case, there are no comparative data about re-work, however the process work products were created and their features are presented in an appropriate form, as detected by an evaluation of the work products.

Table 6 – Example of activity for the Sell Process

Activity	**V05 – Define Customer Requirements**
Goal	Identify customer needs, expectations and restrictions with relation to the product.
Roles	Requirements Engineer Sales Team
Inputs	Proposal Request, Template for the Traceability Document, Template for the Technical Proposal, Template for the Interview Document.
Outputs	Technical Proposal, Traceability document.
Resources	MS Word, MS Excel.

Tasks	• Analyze business needs contained in the Proposal Request to guide the requirements identification.
	• Identify requirements sources, create Traceability Document and fill in the "Sources List" folder. Examples of sources are: people, documents, legacy systems, etc.
	• If necessary, consult legacy systems and related documentation to identify customer requirements.
	• Identify customer requirements through interviews.
	  • Prepare the interviews beforehand based on business needs.
	  • Produce Interview Document
	• Create Technical Proposal and document customer requirements in the "Preliminary Requirements and General Functionalities" section, using natural language, so that the customer can understand them.
	• Generate the system conceptual model in the "Conceptual Model" section of the Technical Proposal. A suggestion for the conceptual model is the site map.
	• Fill in the "Interactions List" folder of the Traceability Document with the customer requirements sources and the rationale that lead to the requirement.

Process: Sell
Activity: Define Customer Requirements

Good Practices:
- Use business needs to guide the elicitation
- Identify requirements sources
- Perform interviews to identify requirements
- Register rationale that lead to the requirement

Work Products:
- Technical Proposal
- Traceability Document
- Interview Document

Figure 5. "Define Customer Requirements" Activity

Process: Sell
Activity: Define Product Requirements

Good practices:
- Consult requirements sources
- Define operational process
- Define operational environment
- Define system bounds
- Derive product requirements from customer requirements
- Document traceability between customer and product requirements

Work Products:
- Traceability Document
- Requirements Document

Figure 6. "Define Product Requirements" Activity

Process: Requirements Engineering
Activity: Detail Requirements and Functionalities

Good Practices:

- Model the system architecture
- Allocate requirements to the architecture components

Work Products:

- Requirements Document

Figure 7. "Details Requirements and Functionality" Activity

Process: Requirements Engineering
Activity: Analyze requirements critically

Good Practices:

- Prioritize requirements
- Detect conflicts
- Solve conflicts
- Communicate conflict resolution decisions
- Check if the requirements fulfill business needs and goals

Work Products:

- Requirements Document
- Traceability Document
- Conflict List

Figure 8. "Analyze Requirements critically" Activity

Process: Requirements Engineering
Activity: Verify and validate requirements

Good Practices:

- Create multidisciplinary teams to revise requirements
- Define checklists to validate requirements
- Use prototypes to animate requirements

Work Products:

- Requirements Document
- Verification Checklist

Figure 9. "Verify and Validate Requirements" Activity

5. Concluding Remarks and Future Work

This work aims at reducing CMMI-DEV abstraction level, specifically for the Requirements Development and Management process areas, making easier its introduction in small organizations. This is done by identifying requirements good practices and explaining in more details how they can be performed, without forgetting that each organization has its own reality and that no technique or practice fits well in all cases, so adaptations can be done.

The structure proposed in PROREQ eases the accomplishment of a RE process implementation, because it gives more details about how to make sure each practice is being correctly followed, instead of only enumerating the practices. As mentioned before, PROREQ good practices were extracted from several sources, but its structure was based on CMMI-Dev requirements and management engineering process areas. Therefore, it contains the same 2 specific goals, 10 practices and 51 sub-practices of CMMI-Dev, but provides approximately 200 good practices, together with expected input and output work products. The complete set of practices, as well as the corresponding work products can be found elsewhere [Diniz, 2007].

Our work is in accordance to the continuous improvement philosophy, but was adapted to financial, resource, and time restrictions commonly found in small organizations. By using parts of the PROREQ guide it was possible to define a requirements process in a small Brazilian software organization. Some of the projects conducted in that organization after the definition of this process presented good results, with less re-work done regarding requirements problems, while in other projects results cannot be compared yet, but they already presented the process work products and their main features, which characterizes the evidence of the process execution.

As future work, we intend to apply the PROREQ guide in other software development organizations, aiming at validating its utility and at the same time improving its utilization process. Another possibility is to cover a bigger number of CMMI process areas that are related to the areas studied in this work, such as Configuration Management and Technical Solution. Another area that deserves more research is to collect data that allows estimating the implementation time for organizations that intend to use the guide. Different types of organizations could be subject of case studies aiming at identifying how they use the guide to create their processes. At the end, guidelines could be proposed of how to better apply the guide according to the organization type, area of the project to be developed, team size, project size, etc.

We plan to make the PROREQ guide available in a wiki soon, in order to make its usage easier and public. Also, a tool can be implemented to ease PROREQ utilization.

References

Beecham, S., Hall, T., Rainer, A. (2003) Software Process Improvement Problems in Twelve Software Companies: An Empirical Analysis, Empirical Software Engineering, 8, 7–42.

Beecham, S., Hall, T., Rainer, A. (2005) Defining a Requirements Process Improvement Model, In Software Quality Journal, 247–279, Inc. Manufactured in The Netherlands.

Cuevas, G., Serrano, A., Serrano, A. (2004), Assessment of the requirements management process using a two-stage questionnaire, Proceedings of the Fourth International Conference on Quality Software (QSIC'04).

Diniz, A. PROREQ - Um Guia facilitador para a implantação dos Processos de Gestão de Requisitos, Master Thesis, ICMC – University of São Paulo (in Portuguese), 2007.

Hagge, L., Lappe, K. (2004) Patterns for the RE Process, In Requirements Engineering Conference, Proceedings of the 12th IEEE International Requirements Engineering Conference, p. 90-99.

ISO (2001) ISO/IEC 12207 Amendment - Information Technology - Amendment 1 to ISO/IEC 12207, The International Organization for Standardization and the International Electrotechnical Commission, Geneve: ISO, 2001.

ISO (2004a) ISO/IEC 12207 Amendment - Information Technology - Amendment 2 to ISO/IEC 12207, The International Organization for Standardization and the International Electrotechnical Commission, Geneve: ISO, 2004.

ISO (2003) The International Organization for Standardization and the International Electrotechnical Commission. ISO/IEC FDIS 15504-2, Information Technology - Process Assessment – Part 2 - Performing an Assessment, Geneve: ISO.

ISO (2004b) The International Organization for Standardization and the International Electrotechnical Commission. ISO/IEC FCD 15504-5 - Information Technology - Process Assessment - Part 5: An exemplar Process Assessment Model, Montreal: ISO/IEC JTC1 SC7.

Jiang, L., Eberlein, A., Far, B. H. (2004) A Methodology for Requirements Engineering Process Development, In Proceedings of the 11th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems (ECBS'04), v. 00, p. 263.

Kauppinen, M., Vartiainen, M., Kontio, J., Kujala, S., Sulonen R. (2004) Implementing requirements engineering processes throughout organizations: success factors and challenges, Information and Software Technology 46 (2004) 937–953.

Kotonya, G., Sommerville, I. (1998), Requirements Engineering: Process and Techniques, John Wiley and Sons.

Lamsweerde, A. (2000) Requirements Engineering in the Year 00: A Research Perspective, 22nd International Conference on Software Engineering (ICSE '00), Limerick, Ireland.

MPS.BR – Guia Geral, versão 1.1, Associação para a promoção da excelência do software brasileiro – SOFTEX, 2006**.** Disponível em: <http://www.softex.br/mpsbr/_guias/default.asp>. Acesso em: 20 de março de 2009.

Niazi, M., Wilson, D., Zowghi, D., Wong, B. (2004), A Model for the Implementation of Software Process Improvement: An Empirical Study, PROFES 2004, LNCS 3009, pp. 1-16.

PMBOK – Project Management Body of Knowledge, PMI – Project Management Institute. A guide to the project management body of knowledge. Syba: PMI Publishing Division, 2004.

Software Engineering Institute/SEI (2006), CMMI® for Development, Version 1.2, CMU/SEI-2006-TR-008 ESC-TR-2006-008, Improving processes for better products, August.

Sommerville, I., Sawyer, P (1997), Requirements Engineering – A good practice guide, John Wiley and Sons.

Sommerville, I., Ransom, J. (2005) An Empirical Study of Industrial Requirements Engineering Process Assessment and Improvement, In ACM Transactions on Software Engineering and Methodology, Vol. 14, No. 1, Pages 85–117, January.

SWEBOK – Guide to the Software Engineering Body of Knowledge, A project of the IEEE Computer Society Professional Practices Committee, 2004. Available at: <http://www.swebok.org>. Accessed in: 25/03/2009.