

## Process Reference Guides – Support for Improving Software Processes in Alignment with Reference Models and Standards

Jean Carlo R. Hauck<sup>1,2</sup>, Christiane Gresse von Wangenheim<sup>1,2</sup>,  
Richard H. de Souza<sup>1</sup>, Marcello Thiry<sup>2</sup>

<sup>1</sup> UFSC – Universidade Federal de Santa Catarina. Campus Universitário, Trindade.  
88040.900 Florianópolis - Santa Catarina, Brazil

<sup>2</sup> UNIVALI – Universidade do Vale do Itajaí. Rodovia SC 407, Km 4.  
88102.280 São José - Santa Catarina, Brazil.

jeanhauck@egc.ufsc.br, {gresse, richardhenrique, marcello.thiry}@gmail.com

**Abstract.** Software process improvement in small organizations in alignment with reference models or standards remains complicated. In this paper, we enhance an approach for software process improvement and introduce the concept of process reference guides as a way to explicitly map reference models/standards and potential solutions in order to systematize and facilitate the process definition in improvement initiatives. Experiences provide a first indication that such reference guides can be useful in this context and may help to reduce the effort for process definition.

**Keywords:** Software Process Improvement, Process modeling, CMMI, ISO/IEC 15504, Alignment

### 1 Introduction

Many small organizations have problems in improving effectively and efficiently their software processes [1]. Although, today, exist a variety of widely accepted reference models for various processes, such as, CMMI [2], ISO/IEC 15504 [3], ITIL [4], still only a small number of small organization manages to successfully systematize their software process in alignment with those models. For example, a survey run by the ISO/IEC JTC1/SC7 Working Group 24 on Life Cycle Processes for Very Small Enterprises (VSEs) [5], found out that less than 18% of very small organizations (with less than 25 employees) are certified and among the 82% of VSEs not certified, only 25% claim to use standards.

One of the predominant reasons for this is that reference models or standards are often perceived as difficult and bureaucratic, not offering adequate guidance for small business environments. These models are typically developed focusing on larger companies and do not simply “scale down” to small organizations, especially to those with a low capability level [1]. Consequently, compliance with such standards or reference models is difficult, if not impossible for them to achieve.

Another aspect is that these reference models or standards define requirements in relation to software processes. They do not intend to nor provide detailed support on how these requirements can be satisfied. In this context, process models, such as RUP [6] or ICONIX [7] for example, provide generic process descriptions. Yet, such process models also cannot simply be deployed in an organization, as in order to be effective and operational, processes need be defined based on the actual processes in place in the organization taking into consideration its specific characteristics, needs and limitations [8].

Thus, in order to assure an effective adoption of a defined process, the process should be defined in a balanced way by eliciting the actual process in place and only improving or completing the existing process, where necessary [9]. In this context, there exist a variety of approaches, which deal on different levels of formalism with the descriptive modeling of software processes (e.g., [10] [11] [12] [13] [14]).

However, considering high-level reference models on one side and software process modeling approaches on the other side, in practice, we observed a need for intertwining descriptive and prescriptive process modeling in order to come up with a defined process aligned with reference models. Therefore, we need to integrate descriptive and prescriptive process modeling activities as well as provide guidance, which in a more refined way presents possible solutions for process shortcomings in alignment with reference models or standards.

In this paper, we describe such an extension to the process improvement approach ASPE/MS (Approach for Software Process Establishment in Micro and Small Companies), which has been developed to ASPEI/MS (Approach for Software Process Establishment and Improvement in Micro and Small Companies) specifically for software process improvement (SPI) in small companies. As part of this, we introduce the concept of process reference guides, which can be considered a flexible collection of alternative processes, techniques and tools mapped to practices required by reference models and standards. Such process reference guides can facilitate the improvement of existing processes by indicating various alternative solutions to be tailored to the specific needs of the organization. An example process reference guide for project monitoring & control is presented. We also summarize first experiences and feedback from the application of the approach in practice.

## **2 ASPEI/MS**

Few approaches have been described specifically for the establishment and improvement of software processes in small organizations [13], [14]. In this context, we are defining ASPEI/MS by integrating and adapting existing approaches (including [10], [11], [12], [13], [14]) to the characteristics of small software companies [1], [15]. We do not intend to develop a new method, but rather aim at the integration and tailoring of existing approaches to the context of small software companies.

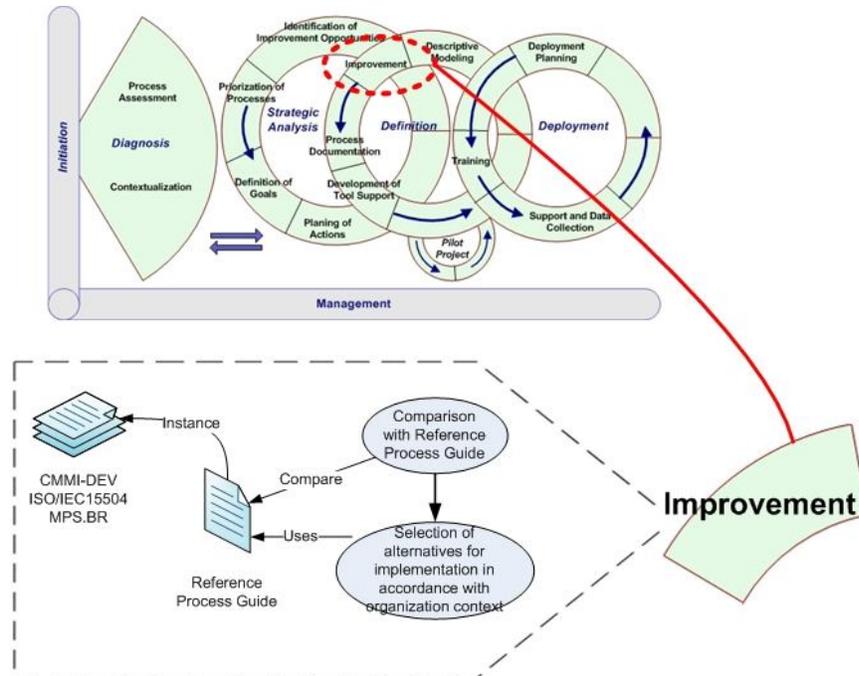


Fig 1. ASPEI/MSC – enhancing the improvement of the process definition

The principal phases of the approach are Instantiation, Diagnosis, Strategic Analysis, Definition and Deployment to be executed in an iterative and incremental way in order to establish and improve step-by-step one or more process(es) within an organization (Figure 1). During the **Instantiation** phase, the software process improvement initiative is prepared and the necessary infrastructure and pre-conditions are created (e.g., allocating personnel for SPI). During the **Diagnosis** phase, a process assessment of the actual software process is performed, identifying its strengths and weaknesses and establishing a target process profile to be achieved. Such an assessment can be done on different levels of scope and detail, ranging from an overview assessment to a focused assessment of a set of processes in alignment with one or more reference models or standards, using, for example, the MARES method [16]. Based on the assessments results, during **Strategic Analysis**, processes to be established and improved are prioritized in accordance to the organization’s business and improvement goals. As a result, improvement cycles and the process(es) to be improved in each cycle are defined. During the **Process Definition** phase, each process chosen is modeled, improved and documented in form of an organizational process guide. This is done by eliciting the actual process in place creating a descriptive process model, which in alignment with relevant reference models or standards is being improved, where necessary. The standard process being defined is applied and evaluated in pilot-projects and then, during **Deployment**, institutionalized throughout the organization.

In addition, the approach also covers the management of the process establishment, including planning, monitoring & control and post-mortem.

We developed a first version of this approach (called ASPE/MS) in 2005, which has since then been applied successfully in several SPI initiatives in small organizations [13], [15], [16], [17]. However, while adopting this approach, we also perceived a specific difficulty in finding solutions for weaknesses identified especially in order to satisfy requirements of reference models or standards. And, although, today, exists a variety of alternative solutions to satisfy requirements of reference models or standards, there basically does not exist a more systematic support or know-how on which alternative processes, techniques or tools could be adopted in a specific environment and how they may need to be tailored to fit the specific context. Thus, the identification of potential solutions to improve an organizational process still remains a complex task, which requires high level of expertise and is typically performed without reuse and no explicit, specific know-how. This represents especially a problem to small organizations, which typically cannot count with experienced software process engineers, which have a profound understanding of the reference models or standards and the available processes, techniques or tools. To fill this gap, we enhanced the improvement step in the ASPE/MS approach by introducing the concept of process reference guides, which map existing alternatives with required practices in reference models and standards and, thus, facilitate the identification of possible solutions in order to improve the current process in place (Figure 1).

### **3. Introducing Process Reference Guides**

By process reference guides we understand a model, which maps requirements of reference models or standards (e.g., outcomes or base practices) with a broad variety of processes, techniques and/or tools to satisfy these requirements. Process reference guides are expected to provide detailed information on these processes, techniques and tools as well as information on their applicability in certain contexts and on how to tailor them to suit specific needs and characteristics. They are not a single, prescriptive process description, but rather a collection of diverse alternative solutions, from which potential improvement solutions can be selected and tailored to a specific organization's standard process.

As one of the purposes of the process reference guides is to provide guidance for process improvement in consistency with reference models and standards, the guides are based on relevant reference guides and present detailed information on how the requirements of the models can be satisfied. A structure for such a process reference guide is shown in Table 1.

Such process reference guides can be developed initially based on a literature research, compiling and comparing well-accepted processes, techniques and tools as well as consolidate experiences on their application in practice. This includes also a detailed analysis of relevant reference models and standards.

**Table 1.** Process reference guide structure

<b>Introduction</b>	Visualization of the scope of the respective process and its relation to other processes (Figure 2).
<b>Basic concepts</b>	Description of basic concepts and terminology in relation to the respective process.
<b>Assessment</b>	Set of assessment indicators derived from the considered reference models and standards (e.g. CMMI-DEV and ISO/IEC 15504) with a mapping of potential processes, techniques and tools, which may help to improve the process (Figure 3).
<b>Typical activities</b>	Set of typical activities executed as part of the respective process, describing for each activity its purpose, objective, steps, working products and templates, roles as well as relationships between activities.
<b>Practices</b>	Presentation of practices required by reference models or standards, such as, CMMI-DEV or ISO/IEC 15504.
<b>Techniques</b>	Description of relevant techniques, which can be used in order to establish required practices and achieve the required outcomes.
<b>Tools</b>	Comparison and reviews of relevant tools in relation with the support they provide for the specific process in alignment with the considered reference models and standards.

However, the principal focus of the development of such reference guides has to be on their continuous evolution. Such an effort, in practice is only viable through collaborative knowledge management. In this context, various types of tools can support the management of such process reference guides. For example, process modelling tools, such as, SPEARMINT [18] or Wagner [19]; general modeling tools, such as, Enterprise Architect [39], BPMN Designer (free) [40] or tools specifically designed for process framework development, such as the Eclipse Process Framework Composer (EPF Composer) [41] as well as general tools for collaborative knowledge management, including semantic web and ontologies [46], WIKIs [20] and, especially, semantic WIKIs [47].

Currently, we are initiating the development of process reference guides for various processes focusing on the context of small organizations. An example is a process reference guide on Project Monitoring and Control [17] being developed in alignment with CMMI-DEV [2], ISO/IEC 15504 [3] and the Brazilian Software Process Improvement Model MPS.BR [21] based on literature in the area of project monitoring & control, including [2], [23], [24], [25], standards [26], [27] and guides [28], [29], [30], as well as experience reports in the context of small organizations [1], [31], [32], [33] and our experiences on establishing this process in practice.

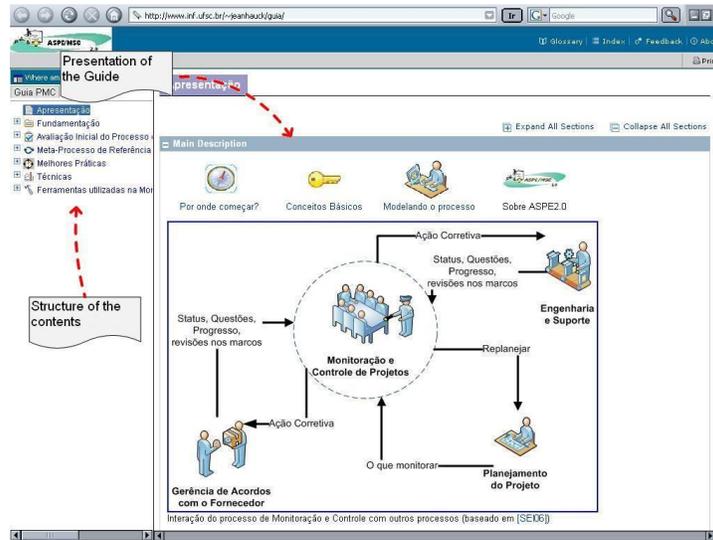


Fig 2. Introduction of the project monitoring & control guide [42]

For example, Figure 2 shows the introduction of the process monitoring & control guide. Starting from this page, the guide can be used in different ways depending on the level of expertise of the process engineer, either by directly accessing an aspect of interest or by guiding the engineer through an assessment questionnaire, which explicitly links potential solutions to identified shortcomings (Figure 3).

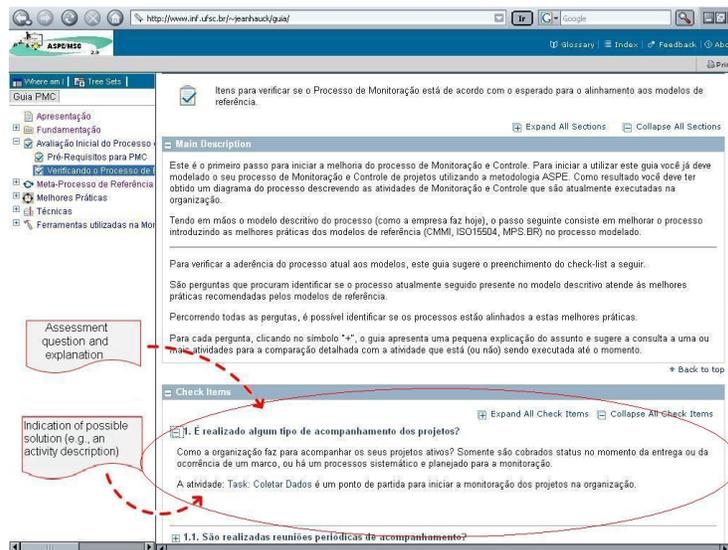


Fig 3. Support provided for process assessment [42]

In this way, process reference guides provide a structured overview on existing processes, techniques and tools mapping these alternatives to reference models and standards. Offering such a process reference guide as support, is expected to reduce the related effort and time as well as to improve the quality and adherence of defined processes.

Software processes can be defined on various levels of detail, ranging, typically, from concrete instantiated processes in a specific project, organizational standard processes to high-level and generic reference models or standards. Using the Software Process Engineering Metamodel (SPEM) [34], a standard for software process modeling, we can express these levels and place the concept of process reference guides. Therefore, we enhance the SPEM architecture of four to six levels of abstraction, refining level M1 in order to separate organizational process models from generic process models, similar to the proposal by [35]. In Figure 4, these different levels of processes are shown with respect to the four-layered organization of SPEM. Layer 3 describes the process modeling metamodel. Layer M2 defines the ASPE/MS notation based on the SPEM notation as model for the definition of process models.

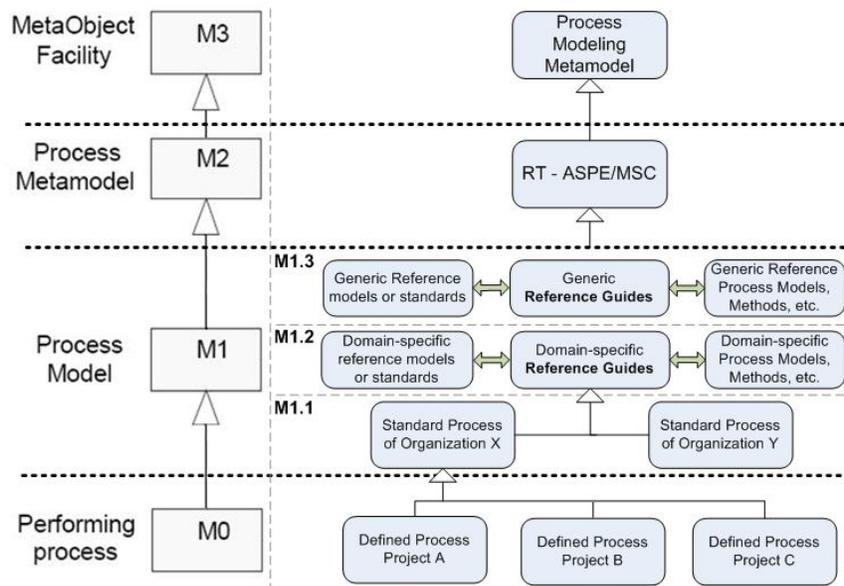


Fig 4. Definition of process levels in alignment with SPEM

Following the original definition of SPEM, Layer M1 contains process models. Yet, this representation does not explicitly express different levels of process modeling and, thus, we suggest dividing layer M1 in three sub-layers:

- Layer M1.1 represents organizational standard processes, which are sets of definitions of the processes that must be incorporated into the defined processes that are implemented in projects across the organization.
- Layer M1.2 represents references models or standards for a specific domain or sector, such as, e.g., S4S (SPICE for Space [36]), AutomotiveSPICE [43] as well as domain-specific process models or reference process guides, such as, for example, a reference process guide for project monitoring & control in small organizations.
- Layer M1.3 represents generic reference models or standards, such as CMMI, ISO/IEC 15504 as well as generic process models, such as RUP, and generic reference process guides.

Layer M0 represents the defined process, which is an instantiated process in a specific project that is managed (planned, monitored and adjusted), tailored from the organization's set of standard processes according to the organization's tailoring guidelines. It provides a basis for planning, performing, and improving a project's tasks and activities.

#### **4 First Experiences**

So far, we have started to apply the enhanced approach ASPEI/MSD using the project monitoring & control guide in two process improvement initiatives. The first application was run in parallel to the development of the guide in the software R&D group CYCLOPS [44] at the Federal University of Santa Catarina/Brazil. A second application has been run in a small software company in Florianópolis/ Brazil.

The focus of the first improvement initiative was focused on project management, including monitoring & control as well as on requirements development and management. The process improvement was coordinated full-time by a junior process engineer and weekly supported by external senior consultants. We followed the ASPEI/MSD approach and during the Definition phase elicited the processes in place through process workshops with the process performers. A gap analysis was done in order to identify shortcoming of the processes actually in place in relation with the respective reference models. Based on an initial version of the process reference guide, process engineers started to improve the processes in cooperation with process performers. The organization's standard process has been documented in form of an Electronic Process Guide (EPG) as part of the organizational WIKI. We, then, started to implement the process in pilot projects, identifying aspects in which it remained inefficient or ineffective. Once an operational standard process had been achieved, we started to train and deploy the process organization-wide.

The total effort spent during the process improvement, so far, is about 265 person-hours, with about 80 person-hours spent on the definition and deployment of the project monitoring & control process, which represents about 2% of the total effort spent by the R&D group during this period. Today, the standard process is used in all projects of the CYCLOPS Group. More than 60 percent of the specific practices of the project monitoring & control process area of CMMI-DEV are characterized at

least as largely implemented in the majority of the software projects of the organization.

In this first application, we could not yet identify a significant advantage through the usage of process reference guides, principally due to the fact that the project monitoring & control guide was being developed in parallel.

Using the organizational WIKI for the documentation of the process description was considered helpful, especially as it made the review and collaborative evolution easy. However, we also observed some weaknesses, especially when compared to other process modeling tools, which provide more support for structuring EPGs and linking elements automatically as well as offering functionality for the graphical visualization of process elements. As a consequence, we decided to use the Eclipse Process Framework Composer for the development of the process reference guides. The principal advantages are its support for various levels of abstraction and its comprehensive support to interrelate elements of the guide. Another substantial advantage is that the EPF Composer permits in a very simple way to construct an organizational standard process from a reference guide. Similar to other EPG tools, it also allows to publish the process guide on the web. Yet, a significant shortcoming at the moment is the inability to continuously evolve the framework easily in a collaborative manner. But, such a support is foreseen to become available as part of the next releases with EPF WIKI.

The second application took place after a first version of the process reference guide on project monitoring & control had been developed. We also applied the ASPEI/MSD approach focusing on the establishment of the project planning, monitoring & control as well as the requirements management. The improvement initiative has been realized by a part-time process engineer and external junior and senior consultants. Similar to the application at the CYCLOPS Group, we performed a high-level process assessment in the beginning and started to define the process in place. Based on a gap analysis in relation with the considered reference model, we identified improvement opportunities and indicated possible solutions using the process reference guide. The defined standard process has been applied in a pilot project and the company is now starting to use the process organization-wide.

During this application of ASPEI/MSD, we started to perceive an indication for a potential effort reduction during the process definition. In comparison also to other initiatives in small organizations where we improved the project monitoring & control process and spent an average of 23 person-hours on the process definition, the definition of the project monitoring & control process using the process reference guide was reduced to 12 person-hours. Yet, this may have happen also due to other reasons, as, e.g., the fact that the junior consultant got more experienced after having established the process in several organizations. Subjectively, the consultants perceived that the process reference guide can facilitate the process improvement by mapping weaknesses of the organization's process to expected practices of the reference models and standards. In this respect, especially, the association of practices and solution alternatives was considered valuable. So far, a principal shortcoming is that insufficient information on the applicability and tailoring of solution alternatives in certain contexts is available. In addition, we also observed that even using the EPF Composer, it remained difficult to navigate through the guide.

## **5 Discussion**

Our experiences provide a first indication that such process reference guides can be useful in order to support the definition of a software process. In contrast to reference models or standards, such as CMMI or ISO/IEC 15504, they intend to provide more concrete information on how to implement the required practices by presenting various alternatives. In this way, they also go a step further than implementation guides, which are being published as part of some reference models, as, e.g. the implementation guide of the Brazilian Process Improvement Model MPS.BR [21] or [28].

A difference of the concept of process reference guides in comparison to process frameworks, such as, the Rational Unified Process (RUP) [6], ICONIX [7] or Cleanroom [37] is that the idea of reference guides is to offer a broader process vision, which may cover several different process frameworks and presenting them (or parts of these frameworks) as alternative solutions, including also information on when to apply which and how to tailor such frameworks to a specific organization. Another similar concept are Process Patterns, which are collections of general techniques, actions, and/or tasks (activities) for developing object-oriented software [38] [45]. They describe a proven, successful approach and/or series of action for developing software. And, although, originally limited to object-oriented software development, they also seem to be valuable using other methodologies. Ambler classifies three types of process patterns [38]: task process patterns, stage process patterns, phase process patterns. Each pattern is structured as Forces, Initial Context, Solution, and Resulting Context. But, as process pattern are understood to describe what should be done, they do not describe the exact details of how.

Thus, in comparison, we consider a principal strength of process reference guides their detailed description also on how to execute practices as well as their larger variety of potential alternative.

## **6 Conclusions**

In this paper, we introduce the concept of process reference guides as a way to explicitly map reference models and potential solutions in order to systematize and facilitate the process definition in improvement initiatives, principally in small organizations. Experiences provide a first indication that such reference guides can be useful in this context and may help to reduce the effort for process definition. We are planning to continue the application of the approach in future improvement programs and the evaluation of the process reference guides as well as to amplify the definition of reference guides for other process areas, such as, project planning, requirements development, etc.

## Acknowledgement

Our thanks to all involved in the improvement initiatives at CYCLOPS/UFSC and the small software company.

## References

1. Richardson, I., Gresse von Wangenheim, C. Why are Small Software Organizations Different?. *IEEE Software*, vol. 24 no.1, January/February (2007)
2. CMMI Team. CMMI-DEV: CMMI for Development, version 1.2. Software Engineering Institute/Carnegie Mellon University, Pittsburgh (2006)
3. International Organization for Standardization. ISO/IEC 15504: Information Technology – Process Assessment: Part 1- Part 5 (2003 -2006).
4. ITIL v3, 2007. <http://www.ital-officialsite.com>
5. Laporte C. Y., Alexandre S., Renault A. Developing International Standards for Very Small Enterprises. *IEEE Computer*, vol. 41, no. 3, March (2008)
6. IBM Rational Unified Process. <http://www-306.ibm.com/software/awdtools/rup> {2008-05-06}.
7. Rosenberg, D., Collins-Cope, M., Stephens, M. Agile Development with ICONIX Process. Apress. (2005)
8. Acuña, S.T. et. al. The Software Process: modeling, evaluation and improvement. Handbook of Software Engineering and Knowledge Engineering. Vol.1, World Scientific Publishing Company (2001)
9. Jacobson, I. Does Process come from the top or from the bottom? Flexi Newsletter, 1 (2007). <http://www.flexi-itea2.org/newsletter.php> {2008-04-12}.
10. Machado, L. F., Oliveira, K. M., Rocha, A. R. Using Standards and Maturity Models for the Software Process Definition. Quality Week. Belgium (2000)
11. Scott, L., Zettel, J., Hamann, D. Supporting Process Engineering in Practice: An Experience Based Scenario. IESE Technical Report no. 033.00/E, Fraunhofer Institute for Experimental Software Engineering, Germany (2000)
12. Becker, U. Towards Systematic Knowledge Elicitation for Descriptive Software Process Modeling. IESE Technical Report n° 036.01/E, Fraunhofer Institute for Experimental Software Engineering, Germany (2001)
13. Thiry, M., Gresse von Wangenheim, C., Zoucas, A. Uma Abordagem para a Modelagem Colaborativa de Processos de Software em Micro e Pequenas Empresas. SBQS – Brazilian Symposium on Software Quality, Vitória. (2006)
14. Dingsøy, T., Moe, N.B., Dyba, T., Conradi, R. A Workshop-Oriented Approach for Defining Electronic Process Guides. In: JURISTO, N.; ACUÑA, S.T. Software Process Modeling, Kluwer Academic Publishers (2005)
15. Gresse von Wangenheim, C., Weber, S., Hauck, J. C. R., Trentin, G. Experiences on establishing software processes in small companies. Elsevier, Vol. 48, September (2006)
16. Gresse von Wangenheim, C., Anacleto, A., Salviano, C. F. Helping Small Companies Assess Software Processes. *IEEE Software*, Vol. 23, No. 1, Jan/Feb. (2006)
17. Hauck, J. C. R., Wangenheim, C. G., Thiry, M. Suportando a Modelagem de Processo de Monitoração e Controle em Micro e Pequenas Empresas, alinhado ao CMMI, MPS.BR e ISO/IEC15504. SBQS – Brazilian Symposium on Software Quality, Ipojuca (2007)
18. Becker-Kornstaedt, U. et. al. Support for the Process Engineer: The Spearmint Approach to Software Process Definition and Process Guidance. Springer Lecture Notes on Advanced Information Systems Engineering, LNCS Vol. 1626/1999, Heidelberg. (1999)

19. Scott, L., Kurniawati F. WAGNER – Web-bAsed process Guide aNd Experience Repository. Centre for Advanced Software Engineering Research (CAESER), University of New South Wales, Sydney. (2001)
20. Louridas, P. Using wikis in software development. *IEEE Software*, vol. 23 no. 2, March/April (2006)
21. SOFTEX, MPS.BR – Brazilian Software Process Improvement Model – Implementation Guide, version 1.2, Brasília. (2007) (in Portuguese)
23. Project Management Institute. A Guide to the Project Management Body of Knowledge (PMBOK® Guide), 3. ed., PMI, Pennsylvania (2004)
24. Hughes, B., Cotterell, M. *Software Project Management*, McGraw-Hill, 3. ed., (2002)
25. Department of Defense & US Army. PSM - Practical Software and System Measurement, A foundation for Objective Project Management, Ver. 4.0c., (2003)
26. ANSI, ANSI/EIA 748. A Standard for Earned Value Management Systems. ANSI Standard. (1998)
27. International Organization for Standardization. ISO/IEC 10006: Quality Management – Guidelines to Quality in Project Management, 2. ed., (2003)
28. Jalote, P. *CMM in Practice: Processes for Executing Software Projects at Infosys*. Addison Wesley Longman. (2000)
29. Kulpa, M. K., Johnson, K. A. *Interpreting the CMMI®: a process improvement approach*, Auerbach Publications (2003)
30. Kerzner, H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. Wiley, 9 ed. (2003)
31. Orci, T; Laryd, A. *CMM for Small Organisations Level 2*. Suécia: Umeå University. (2000)
32. Meneses, J. B. *Inspector: Um Processo de Avaliação de Progresso para Projetos de Software*. Máster Thesis, Federal University of Pernambuco, Recife/Brazil (2001)
33. Otoyoy, S., Cerpa, N. *An Experience: A Small Software Company Attempting to Improve its Process*. *Software Technology and Engineering Practice* (1999).
34. OMG - Object Management Group. *Software Process Engineering Metamodel Specification, Version 1.1 OMG Specification*. (2005)
35. Järvi A., Makila, T. *Observations on Modeling Software Processes with SPEM Process Components*. Proc. of 9th Symposium on Programming Languages and Software Tools, Estonia, (2005)
36. Cass, A., et. al. *SPiCE for SPACE: A Method of Process Assessment for Space Software Projects*. Proc. of the International SPICE Conference (2000)
37. Mills, H.; M. Dyer and R. Linger. *Cleanroom Software Engineering*. *IEEE Software*, vol. 4 no. 5, Sep. (1987)
38. Ambler, Scott. *Process Patterns - Building Large-Scale Systems Using Object Technology*, Cambridge University Press (1998).
39. Sparx Systems. <http://www.sparxsystems.com.au> {2008-05-05}.
40. Intalio, BPMN Designer. <http://www.intalio.com> {2008-05-06}.
41. Eclipse Process Framework. <http://www.eclipse.org/epf> {2008-05-07}
42. Process Reference Guide. <http://www.inf.ufsc.br/~jeanhauck/guia/> {2008-05-06}
43. AutomotiveSPICE. <http://www.automotivespice.com> {2008-05-06}
44. CYCLOPS Research Group. <http://www.cyclops.ufsc.br/> {2008-05-06}
45. The Process Patterns Resource Page. <http://www.ambysoft.com/processPatternsPage.html> {2008-05-06}
46. Berners-Lee, Tim; James Hendler and Ora Lassila (May 17, 2001). *The Semantic Web*. *Scientific American Magazine*.
47. Sebastian Schaffert, Joachim Baumeister, François Bry, and Malte Kiesel *Semantic Wikis*. *IEEE Software*, July/August 2008.