This version of the article has been accepted for publication, after peer review (when applicable) and is subject to Springer Nature's AM terms of use, but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record is available online at: https://doi.org/10.1007/978-3-642-31199-4_13
Postprint of: Miler J., Wesołowska H., Improvement of Task Management with Process Models in Small and Medium Software Companies. In: Winkler, D., O'Connor, R.V., Messnarz, R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2012. Communications in Computer and Information Science, vol 301. Springer, Berlin, Heidelberg.

Improvement of Task Management with Process Models in Small and Medium Software Companies

Jakub Miler[1], Hanna Wesołowska[2]

[1]Gdansk University of Technology, Department of Software Engineering, Gdansk, Poland
jakubm@eti.pg.gda.pl
[2]Blue Services Sp. z o. o., Quality Assurance Department, Sopot, Poland
hwesolowska@gmail.com

Abstract. Small and medium software companies exhibit many special features that give reason for a dedicated approach to process improvement. They often cannot afford implementing maturity models or quality standards both in terms of time and money. Instead, they expect simpler solutions that can allow to run projects in more systematic and repeatable way, increase quality and knowledge management. In this paper, we present a method focused on improvement of task management using the process models. The method proposes the integration of modeling and task management tools, where models become templates of enacted projects. We applied the method in two case studies with SMEs, where sample process models were built and enacted in adapted task management tools, followed by a survey. The survey resulted in 82.5% of positive answers. The case studies show considerable potential of our method in solving some improvement problems of SMEs.

Keywords: software process improvement, software process model, SPEM, model enactment, project management, task management, SMEs.

1 Introduction

Many researchers [1-6] and organizations [7-8] acknowledge that software process improvement (SPI) in small companies requires a dedicated approach different from simple tailoring of standards and maturity models such as CMMI [9], ISO 12207 [10], and ISO 15504 [11]. Small and medium enterprises (SMEs) with less than 250 employees [12] and, in particular, very small entities (VSEs) with less than 25 employees [13] explore their advantages such as flexibility, innovativeness, market reaction and managerial agility [14] to achieve their specific key business goals identified in [15]. Application of reference models was perceived by SMEs as harnessing their potential as it involves high costs, long term investment, considerable staff and overall change of organizational culture [16]. Success of SPI in SMEs comes from taking advantage of their specifics which should be carefully taken into consideration [17-19].

According to maturity models such as CMMI, first step of process improvement requires manageability, repeatability and reuse of good practices [9]. To achieve this, large enterprises employ effective but complex and expensive software development and process management tools such as IBM Rational tool suite [20]. In turn, small and medium enterprises (SMEs) commonly use simple marketed or proprietary project and task management tools such as [21] which do not offer any process abstraction and reuse. SMEs follow agile-inspired development culture [22] and define tasks manually from scratch in every new project. Even if SMEs describe their processes, these descriptions often remain ignored or bypassed and end up stored in a handbook such as the Quality System [23] or a piece of software called Electronic Process Guide [24] that nobody (or most of people) bothers to study or use in practice.

We identify a common problem of SMEs as a gap between the definitions of processes and the daily management of tasks related to these processes. As a result, the considerable effort on process definition is wasted, which increases negative attitude to future improvement initiatives. The people and the entire organization learn slower, when the reactive approach to daily work cannot be successfully superseded with a proactive approach based on managed processes. The projects fail to meet the success criteria such as schedule, budget and the quality of products.

The research question is how to facilitate introducing more systematic and repeatable way of project realization and practical usage of good practices in SMEs. Our research is aimed at designing

a method of using process definitions in the management of actual projects' tasks in software SMEs. The solution would be to introduce such technique of process definition and their integration with task management that is affordable to SMEs and possible to smoothly introduce into current culture, practice and toolset. This solution should use modern and popular approach to process definition which is supported by easily achievable software tools. Additionally, the definitions should be possible to apply as templates of individual and team tasks specified in currently used task management tools. Such approach facilitates evolutionary change in organizational culture and daily routine, making the process improvement more likely to succeed.

2 Related work

The systematic reviews of SPI for SMEs presented in [3] and [19] reveal the need for automated tools support to facilitate SPI, but the actual SPI models and techniques discussed do not cover tool-assisted enactment of tasks from the process models. Mishra and Mishra [5] discuss several methods of SPI dedicated to SMEs: Self-diagnosis, SPM Model, ASPE-MSC, PRISMS, and MESOPYME. These methods allow to identify the improvements and specify them as process guides or action packages, but none of them mentions how to implement these recommendations in daily task management. Savolainen et al. [25] present a simplified way of process modeling dedicated to SMEs. They apply wall-charts which are documented in an 'electronic version'. This is used more as an analytical and training tool within SPI rather than a process model for project management. Additionally, Savolainen et al. do not propose any specific software tools.

Friedrich and Bergner [26] define a formal method of deriving actual project's actions from plans in plan-driven process enactment. Although they place process models atop the project plans, the translation of process models to project's actions and the implementation in a software tool remains future work. Yaeli and Klinger [27] focus on enacting only one aspect of a process definition, namely the responsibility assignments of roles to work definitions and work products. Gonzalez-Perez and Henderson-Sellers [28] propose an approach to methodology definition and enactment focusing on work products instead of processes and tasks. As a support, they do not use commonly used tools but provide a proprietary MethodComposer toolset, both for process definition and enactment. The method is intended to improve the software development processes, however no application is discussed, in particular in the context of SMEs. Valiente et al. [29] propose model and tool integration schemes based on ontology covering both software engineering and IT service management processes. However, they do not discuss the application of their approach by SMEs and mention only MS Project as a project management tool overlooking the internet-based agile-like task management tools often used by SMEs.

3 The Method

The proposed method involves description of software development processes with models and using these models in process instantiation and daily management of projects and their tasks. The method is assisted with software tools both for process modeling and task management. It consists of the following elements:

- process modeling,
- process modeling tool and task management tool integration,
- process model enactment,
- process model improvement.

We use process models as templates and guidelines for tasks in actual projects. Among several process modeling paradigms discussed in [30] activity-oriented metamodels satisfy best this goal, as they build upon concepts close to task management. From the activity oriented-metamodels we chose Software Process Engineering Metamodel (SPEM) [31] which is commonly used in description of software processes and is supported by a number of modeling tools.

SPEM defines basic elements of process structure such as disciplines, activities, artifacts and roles. Disciplines group activities in a common area of knowledge e.g. business analysis or testing. Activities describe elements of work to do in the process. Artifacts are work products processed by activities on input, output or both. Roles define skills and competencies of performers of the activities

as well as their responsibilities for activities and artifacts. Additionally, the aforementioned model elements are supplemented with guidelines, among which artifact templates and practice descriptions are most important.

Process models are built with dedicated software process modeling (SPM) tools supporting SPEM such as Rational Method Composer [32] or Eclipse Process Framework Composer [33]. To fit in our approach, an SPM tool must offer saving or exporting models into formats that allow to access the model's data (e.g. XML) and creating easily navigated web pages with description of process elements (disciplines, activities, artifacts, roles). We recommend the Eclipse Process Framework Composer (EPF Composer) which is freely available for commercial use and satisfies the above requirements. It addresses the needs of SMEs in terms of affordability, ease of use, flexibility and compatibility with current toolset. EPF Composer does not require complex deployment and substantial training.

Our method assumes that process models are enacted in a task management (TM) tool currently used in a company. Our focus is to respect current practice and people's knowledge and smoothly overlay process definition on daily tasks. As a TM tool we understand a software tool that allows, among others, to add projects, add tasks and subtasks, assign tasks and subtasks to projects, add roles, assign tasks to roles, assign persons to roles, and add descriptions to tasks. JIRA is an example of such tool [21].

To enact the process models from an SPM tool in a TM tool, these tools must be integrated. We integrate tools on the data level. To understand the data model of an SPM tool, it is helpful to create a sample model of a process, save it to the text file format (e.g. XML) and inspect its structure. Then, the data model of the TM tool should be analyzed compared to the SPM tool model structure. As a result, a mapping between objects that store similar information in both data models is built (e.g. tasks in the task model are able to store similar information as activities in the process model). The mapping should also include missing objects (e.g. persons in task model have no equivalent in the process model). Equivalent objects can be transferred from the SPM tool to the TM tool automatically with a translator or import filter; missing ones will have to be added to the TM tool manually. We assume that appropriate translator is already available or can be easily developed as a stand-alone application or an extension to the TM tool. Table 1 presents the mapping of SPEM elements and their representation in EPF Composer models on the concepts used in TM tools.

Table 1. Mapping of elements from SPEM, EPF Composer model and TM tool data model

SPEM	EPF Composer	TM tool
Role	<ContentElement xsi:type="uma:Role">	Role/group
-	-	Employee/user
Task	<ContentElement xsi:type="uma:Task">	Subtask
Discipline	<ContentCategory xsi:type="uma:Discipline">	Task
Artifact	<Attachment> in <ContentElement xsi:type="uma:Template"> in <ContentElement xsi:type="uma:Artifact">	Template
Guideline	subelements of <ContentElement>	Task description

The instances of activities, disciplines and roles created in the TM tool must be supplemented with their descriptions, templates and guidelines. To achieve this, the process model should be published as a website. Task descriptions can then link to this site. This way an employee assigned to a task can easily access guidelines for this task, download templates of documents or read his/her role description. The integration of SPM and TM tools is shown in Fig. 1.

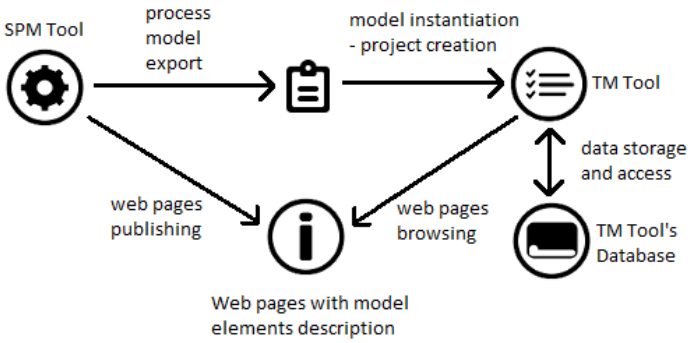

Fig. 1. SPM and TM tool integration

The application of the process model to task management in an actual project requires model enactment. Model is enacted by employees that perform tasks assigned to their roles [26]. To support this, the process model must be instantiated in the TM tool. The TM tool administrator takes the model file and runs the translator which creates an new project where generic model elements become parts of this project (e.g. tasks are created based on modeled activities). These tasks are linked to a process guide exported from the SPM tool, which is made available to employees. Once people are assigned to their roles, they learn their tasks and can start using the TM tool in daily task management. If necessary, they can modify existing tasks or define new ones that can possibly be incorporated back into the model.

The process models must be constantly improved to ensure new projects learn from the past ones in a knowledge management cycle. This requires collecting comments on the process from employees and management, analyzing processes and identifying the needs for change, designing changes, and making changes to models. These activities should be conducted periodically, based on observations of realization of projects carried out in accordance with the models and taking account of changing organization's business needs.

4 Case Studies

The research method to validate our approach involved case studies in representative SME software companies, where we used survey and participant observation to collect the data. The survey was chosen to obtain comparable data from both companies, while the participant observation allowed us to gain better insight and internal knowledge. The goal of the case studies was to assess the suitability and accessibility of the approach to SME software companies and, in particular, to verify the following research hypotheses:

1. It is possible to combine process definitions and task definitions;
2. Such combination can be done with commonly used software;
3. Such combination meets the needs of SMEs as a tool for process improvement.

The case studies involved two companies that fit the profile of potential users – the SME software companies. The first one appears as Company A, the second one – Company B. One of the authors had an opportunity to conduct a participant observation in both, making familiar with their way of working and the problems. This helped to propose the sample process model content and better present how the solution works and what could be the key benefits. It should be noted that both companies had some common needs addressed by solution. Both companies were interested in standardizing the implementation of projects, increasing the quality of processes and products, and knowledge management.

The Company A is a typical representative of SME – with low budget, tight deadlines and short term strategy constraints. Its strategy can be characterized by high sensitivity of the market situation and the need to acquire many new customers for relatively short-term contracts. It carries out orders for 11 years for external clients, often for the public sector, and has about 30 employees. It has close to functional structure and market organizational culture. Because of the relatively small size, it was easy to make changes in the company and communicate directly. The participant observation in company A took 17 months.

The Company B is a rapidly growing organization, which exists for 2 years, but derives from another organization with 12-years experience. It's clients are often from banking and insurance sector but also telecommunication and others. It has close to project structure and hierarchy culture. As for now, it has about 50 employees. It runs longer contracts with smaller financial constraints. Far-reaching strategy is being determined and the company is establishing position in the market and its customer segment. This is watershed moment, the last and best opportunity to streamline and standardize processes. The participant observation in company B covered 7 months.

In both companies, the tool support for automatic creation of tasks in a TM tool based on the process models made in an SPM tool was designed and implemented to verify the hypotheses 1 and 2. The sample content of the process model was developed and converted to web pages. The tools and models were presented to employees and feedback was collected in a survey to verify the hypothesis 3. The survey included the following questions starting with "Can this solution…":

- Q1: facilitate project realization in a more systematic and repeatable way?
- Q2: have positive impact on process and product quality?
- Q3: provide employees with access to company's knowledge, information and templates needed in the project?
- Q4: have positive impact on your comfort with the work because of instructions, templates, clearly assigned responsibility, and the list of tasks in the project known in advance?
- Q5: be applied in practice in the company's projects?

The possible responses were: definitely yes, rather yes, rather no, definitely no. Answers were selected so that respondents should decide about conviction they have (yes or no) and its strength (definitely or rather). We assume that to verify the hypothesis 3 positively we should obtain at least 60% of the answers "rather yes" and "definitely yes" for all questions.

4.1 Company A Case Study

EPF Composer was selected as an SPM tool for company A. The process model was limited to technical documentation department and covered all stages of project realization. The descriptions of model elements and the artifacts were based on those currently used. Two sample process models for two project types (a tender and an ordering) were build. The former consisted of 4 disciplines, 6 tasks, 2 roles, and 3 artifacts, while the latter comprised 3 disciplines, 3 tasks, 2 roles, and 3 artifacts. EPF Composer screenshot with some elements of the process model is shown in Fig. 2.

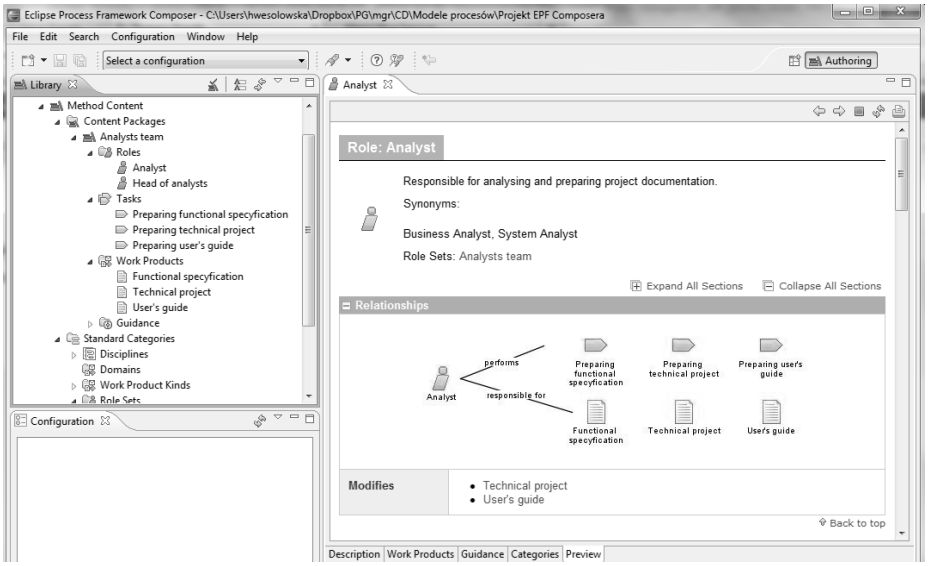

Fig. 2. EPF Composer with part of the process model in company A

In company A, each employee used daily a self-made company's TM tool, coded here as the Projects-Tool. This tool was adequately modified to support the translation of the EPF Composer's process models into its data model. Links to artifact templates and pages with process model elements

generated from EPF Composer were included in the description of tasks. The implementation of the extensions to the Projects-Tool is shown in Table 2 (compare to Table 1).

Table 2. Implementation of model to task translation in TM tool of company A

TM tool element	Projects-Tool database table
Role	TRoles (added)
User	TUsers (modified)
Subtask	TTaskDetails (modified)
Task	TTasks (modified)
Template	Link to artifact template in Description field of TTaskDetails
Task description	Link to intranet webpage in Description field of TTaskDetails

Extension to the TM tool took about 80 working hours. This included learning both tools, data model analysis, design and programming the integration code. Process modeling took about 16 working hours. Most of this time was used to prepare detailed description of elements as well as artifact templates.

The anonymous survey involved 4 employees: an analyst, a technical documentation specialist, a tester and a project manager. This group was representative to the company's structure and the processes covered by our SPI initiative which was introduced bottom-up from the employees' level. In total, the survey achieved 100% positive answers (50% "Definitely yes" and 50% "Rather yes"). For Q1, Q3 and Q4, 3 respondents answered "Definitely yes" and 1 "Rather yes". For Q2, 1 person answered "Definitely yes" and 3 "Rather yes". All respondents believed that our solution could be ("Rather yes" answers) applied in practice in the company (Q5).

Participant observation revealed strong positive attitude of employees to this solution which we attribute to lack of similar SPI initiatives. Our approach was appreciated as a possible way to introduce some order into daily work and achieve better quality of project realization. So far, the solution was not used in any projects because of lack of strong support from the board. This can be due to market centered organizational culture, where improvement of internal processes seems to have lower priority.

4.2 Company B Case Study

The company B is interested in improving the process and hired a consultant, whose task was to develop a standard of projects realization. The consultant developed the process model and its content with the participation of one of the authors and taking account of guidance of project managers and the board. Similarly to company A, EPF Composer was selected as an SPM tool. The process model covered all phases of project realization (offer, analysis, project, realization, deployment) and included guidance for roles, tasks, and artifacts (the latter only for the analysis phase). The model consisted of 3 disciplines, 111 tasks, 12 roles, and 20 artifacts.

In company B, each employee uses every day a popular TM tool – JIRA [21]. The translation of EPF Composer's process models to the JIRA data model was achieved with a plug-in to JIRA. The roles, users, tasks, subtasks and task descriptions of the TM tool data model (see Table 1) are directly implemented by JIRA. The templates were mapped onto JIRA attachments. The process content stored as a website exported from EPF Composer was linked to from the task descriptions. Integration and adaptation of the tools took about 80 working hours, while process modeling took about 24 working hours. Model elements were described briefly and used already developed artifact templates.

The anonymous survey involved 4 employees - 1 member of the board and 3 project managers. This was the first target group of our SPI initiative, which was introduced top-down from the board and managerial level of the company. In total, the survey achieved 65% positive answers (5% "Definitely yes" and 60% "Rather yes") and 35% negative answers (10% "Definitely no" and 25% "Rather no"). In detail, questions Q1, Q3, Q4 and Q5 received 75% positive answers, while question Q2 received 25% positive answers. The latter is attributed to small testing sample and the answers of one respondent being significantly below the average (only negative answers). Due to survey anonymity we cannot assign this to any particular person or role.

In company B there was a positive attitude to our solution, but not as strong as in company A. Participating observer supposes that the company is open to new solutions because it is young and still developing. Nonetheless there were already some failed initiatives to standardize the processes, which could result in slightly less enthusiastic reception of our approach. The quality of project management in B contributed to more successful projects than in A. Company B also exhibited greater confidence in its own competencies. Our approach is decided to be used in everyday practice. In this case, it received strong support from the board. Company B has good position on the market and can afford investing in SPI.

4.3 Assessment of Results

In both case studies presented above it was possible to integrate process definitions and task definitions. Processes were defined in terms of models and content, which were then used to define tasks of particular projects following the models and using the content. This positively verifies the hypothesis 1 of our research.

We used EPF Composer as a process modeling tool in both case studies. It was selected owing to its popularity, free license, support for SPEM metamodel and website based process guides. In task management, we integrated process models with the proprietary TM tool in company A and a common marketed JIRA tool in company B. Both tools were adapted with little effort. In our opinion, the hypothesis 2 of our research can also be positively verified.

The summarized results of the survey from both case studies are shown in Fig. 3.

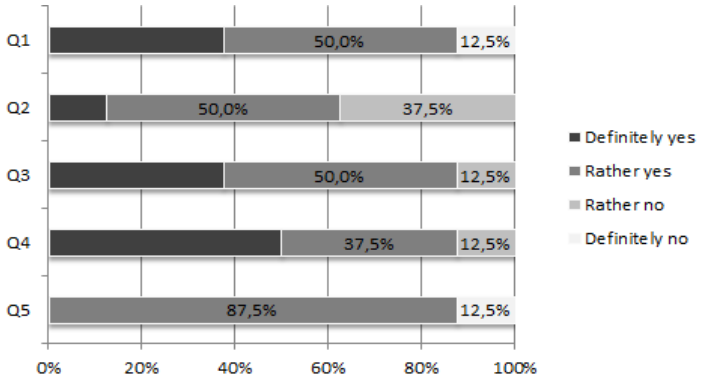

Fig. 3. Summarized survey results in companies A and B

In total, four questions received 87.5% ‘yes’ answers and one question received 62.5% ‘yes’ answers. It allows to verify positively the hypothesis 3 at this stage of the research. In detail, our method has the potential to facilitate project realization in more systematic and repeatable way (Q1). Also support for knowledge management (Q3) was rated highly. Possible increase on the comfort of work was perceived even better (Q4). What is the most important, the survey shown that our solution could be applied in practice to support the companies’ projects (Q5). Lower positive impact of our solution was perceived on process and product quality (Q2), particularly in company B, but the repeatable processes (Q1) are good starting point (first and the most important step) for improvement and achievement of better quality.

5 Conclusions

In this paper, we presented a method of improvement of project management which employs process models to enact project tasks with more guidance and repeatability. The method was applied in two case studies. Because results of process improvement require long time, resources and iterations, current research was limited to sample process modeling, tool integration and a survey on approach assessment complemented with participant observation. The case studies allowed to verify positively all the research hypotheses. In particular, we combined process modeling and task management in two different environment using commonly available tools. The results show that our solution can help small and medium companies in implementing the projects in a systematic and repeatable way, managing the company knowledge and increasing the employees work comfort. The applicability to

practice was also appreciated. Based on the above, we can claim that our solution is a valuable contribution to address the research question. A more robust conclusion might be drawn from a larger sample or some more case studies in comparable organizations.

The main benefit from our solution is the systematization of processes – they have to be modeled, this means – presented in a certain form. This affects the consolidation of knowledge and enables access to it. The whole organization can start to work in the same, repeatable way. Models can be improved from version to version which requires only updating the process model in a PM Tool. Our approach contributes to facilitation of using process models in practice of task management. Process models are enacted in a task management tool already used in an organization. This can reduce the resistance of people as process models turn out to be feasible and useful. Employees are assigned tasks in the TM tool just like before and get specific support during their execution immediately. A model will ensure the consistency of all tasks and enable the project (and the whole organization) to move in the right direction. Our solution can be progressively implemented in selected parts of organization. It is also characterized by low cost and ease of implementation.

Despite many opportunities offered by defined processes and support for their enactment, the success of SPI with our approach still depends on many factors: scope of implementation in a company (insufficient training, missed opportunities), selection of roles, tasks, artifacts (inadequacy to company needs), model content (little valuable roles, tasks, artifacts descriptions), model improvement. The case studies carried out so far provided opinions on the improvement potential of our approach. Future work is planned to enact actual projects based on defined process models and assess the benefits to the process improvement. The observation of actual SPI results to task management with our approach should take into account long time to assess changes (more difficult if the company has previously worked differently for each project), dynamic SMEs changes, strategy, strong dependence on market, organizational structure, employee's roles, and competence needs.

Detailed specification of our approach and the complete description of company A case study is presented in [34].

References

1. Basri, S., O'Connor, R. V.: Understanding the Perception of Very Small Software Companies towards the Adoption of Process Standards, In: Riel, A., O'Connor, R., Tichkiewitch, S., Messnarz, R. (eds.) EuroSPI 2010, CCIS vol. 99, pp. 153–164. Springer, Heidelberg (2010)
2. Pino, F.J., Pardo, C., Garcia, F., Piattini, M.: Assessment methodology for software process improvement in small organizations. Information and Software Technology 52 (10), 1044-1061 (2010)
3. Pino, F. J., Garcia, F., Piattini, M.: Software Process Improvement in Small and Medium Software Enterprises: A Systematic Review. Software Quality Journal 16(2), 237–261 (2008)
4. Pino, F. J., Garcia, F., Piattini, M.: An Integrated Framework to Guide Software Process Improvement in Small Organizations, In: O'Connor, R.V., Baddoo, N., Cuadrago Gallego, J., Rejas Muslera, R., Smolander, K., Messnarz, R. (eds.) EuroSPI 2009, CCIS vol. 42, pp. 213–224. Springer, Heidelberg (2009)
5. Mishra, D., Mishra, A.: Software Process Improvement Methodologies for Small and Medium Enterprises, In: Jedlitschka, A. Salo, O. (eds.) PROFES 2008, LNCS vol. 5089, pp. 273–288. Springer, Heidelberg (2008)
6. Von Wangenheim, C.G., Weber, S., Hauck, J.C.R., Trentin, G.: Experiences on Establishing Software Processes in Small Companies. Information and Software Technology 48(9), 890–900 (2006)
7. Software Engineering Institute, International Process Research Consortium, CMU/SEI-2006-SR-001, (2006)
8. ISO/IEC JTC 1/SC7 Working Group 24, http://profs.etsmtl.ca/claporte/English/VSE/
9. CMMI Product Team: CMMI for Development, Version 1.3, Software Engineering Institute, CMU/SEI-2010-TR-033, (2010)
10. ISO, ISO/IEC 12207:2008 Systems and software engineering - Software life cycle processes, (2008)
11. ISO, ISO/IEC 15504-4:2004: Information technology - Process assessment - Part 4: Guidance on use for process improvement and process capability determination, (2004)
12. European Commission: The New SME Definition, User guide and model declaration, (2005)
13. ISO/IEC 29110-4-1:2011 Software engineering -- Lifecycle profiles for Very Small Entities (VSEs) -- Part 4-1: Profile specifications: Generic profile group. (2011)
14. Richardson, I., Von Wangenheim, C.G.: Why are Small Software Organizations Different?, IEEE Software 24(1), 18–22 (2007)
15. Clarke, P., O'Connor, R. V.: The Meaning of Success for Software SMEs: An Holistic Scorecard Based Approach, In: O'Connor, R.V., Pries-Heje, J., and Messnarz, R. (eds.) EuroSPI 2011, CCIS, vol. 172, pp. 72–83. Springer, Heidelberg (2011)

16. Johnson, D.L., Brodman, J.G.: Tailoring the CMM for small businesses, small organizations, and small projects, In: El Emam, K., Madhavji, N. H. (eds.), Elements of Software Process Assessment and Improvement, pp. 239–259. IEEE CS Press, New York (1999)
17. Kautz, K.: Software Process Improvement in Very Small Enterprises: Does it Pay Off? Software Process – Improvement and Practice 4, 209–226 (1998)
18. Dyba, T.: An empirical investigation of the key factors for success in software process improvement. IEEE Transactions on Software Engineering, 31 (5), 410-424 (2005)
19. Sulayman, M., Mendes. E.: An extended systematic review of software process Improvement in small and medium web companies, IET Seminar Digests 2011 (1), 134-143 (2011), DOI:10.1049/ic.2011.0017
20. IBM, Rational software, www.ibm.com/software/rational/
21. Atlassian, JIRA - Track bugs, tasks, and projects for software development, http://www.atlassian.com/software/jira/overview
22. Manifesto for Agile Software Development, http://agilemanifesto.org
23. ISO, ISO 9001:2008 Quality management systems — Requirements, (2008)
24. Dingsoyr, T., Moe, N.B.: The Impact of Employee Participation on the Use of an Electronic Process Guide: A Longitudinal Case Study, IEEE Transactions on Software Engineering 34 (2), 212-225 (2008)
25. Savolainen, P., Sihvonen, H.-M., Ahonen, J. J.: SPI with Lightweight Software Process Modeling in a Small Software Company, In: Abrahamsson, P., Nathan Baddoo, N., Margaria, T., Messnarz, R. (eds.) EuroSPI 2007, LNCS vol. 4764, pp. 71–81. Springer, Heidelberg (2007)
26. Friedrich, J., Bergner, K.: Formally Founded, Plan-based Enactment of Software Development Processes, In: Raffo D. et al. (eds.) Proceedings of the 2011 International Conference on Software and Systems Process ICSSP'11, Waikiki, Honolulu, HI, USA, ACM, New York (2011)
27. Yaeli. A., Klinger. T.: Enacting Responsibility Assignment in Software Development Environments, In: Dubinsky, Y. (ed.) Proceedings of the 1st international workshop on Software development governance SDG'08, Leipzig, Germany, ACM, New York (2008)
28. Gonzalez-Perez, C., Henderson-Sellers, B.: A work product pool approach to methodology specification and enactment, The Journal of Systems and Software 81 (8), 1288–1305 (2008)
29. Valiente, M.-C., Garcia-Barriocanal, E., Sicilia, M.-A.: Applying Ontology-Based Models for Supporting Integrated Software Development and IT Service Management Processes. IEEE Transactions On Systems, Man, And Cybernetics—Part C: Applications And Reviews 42(1), 61-74 (2012)
30. Hug. C., Front. A., Rieu. D., Henderson-Sellers. B.: A method to build information systems engineering process metamodels, The Journal of Systems and Software 82 (10), 1730–1742 (2009)
31. Object Management Group: Software & Systems Process Engineering Meta-Model Specification, Version 2.0, http://www.omg.org/spec/SPEM/2.0/PDF (2008)
32. IBM, Rational Method Composer, http://www-01.ibm.com/software/awdtools/rmc/
33. The Eclipse Foundation: Eclipse Process Framework Project (EPF), http://www.eclipse.org/epf/
34. Wesołowska, H.: Tool and procedural support for software development processes of SMEs (in Polish), MSc thesis, supervised by J. Miler, Department of Software Engineering, Gdansk University of Technology, Gdansk (2011)