

# A Systematic Review on the Integration of Project Management with Organizational Flows

Maurício Covolan Rosito and Ricardo Melo Bastos

**Abstract**—Software projects are very dynamic and require recurring adjustments of their project plans. These settings can be understood as reconfigurations in the schedule, in the resources allocation and other design elements. Yet, during the planning and execution of a software project, the integration of specific activities in the projects with the activities that take part in the organization's common activity flow should be considered. This article presents the results from a systematic review of aspects related to software projects' dynamic reconfiguration emphasizing the integration of project management with the organizational flows. A series of studies was analyzed from the year 2000 to the present. The results of this work show that there is a diversity of techniques and strategies for dynamic reconfiguration of software projects'. However, few approaches consider the integration of software project activities with the activities that take part in the organization's common workflow.

**Keywords**—Dynamic Reconfiguration, Organizational workflows, Project Management, Systematic Review

## I. INTRODUCTION

THE software products development requires the defined planning and execution of activities according to the project scope, in which is necessary to deal with both technical and management issues [1]. In general, companies are organized in order to manage multiple projects simultaneously [2]. However, unlike the traditional model of project management (which describes the projects individually), models of multi-project management must deal with the interdependencies among the various projects with a set of constraints related to deadlines and available resources [3].

The growing complexity and volume of projects that a project manager must deal simultaneously contributes to the increasing challenges related to the development of projects [4], [5]. As the number of projects increases, the project manager must take into account more and more elements and less time in their decisions. Still, one must consider that during the planning and execution of software projects, different types of tasks are assigned to resources with different characteristics (resulting in a complex set of dependencies between these activities) in order to reach the goals related to time and costs of these projects [6]. Thus, in response to new information or estimations, one may need to make changes to the project plan, such as reallocating resources or canceling tasks [7].

M. C. Rosito is with the Pontifical Catholic University of Rio Grande do Sul, 6681 Ipiranga Avenue, 90619-900 Porto Alegre, RS, Brazil (phone: 51-3320-3558; e-mail: mauricio.rosito@acad.pucrs.br).

R. M. Bastos is with the Pontifical Catholic University of Rio Grande do Sul, 6681 Ipiranga Avenue, 90619-900 Porto Alegre, RS, Brazil (phone: 51-3320-3558; e-mail: bastos@pucrs.br).

In the same scenery, the project manager may need to interact with other departments of the organization during the planning and execution of the project in order to obtain relevant information to the project (contact the human resources department about the need for staffing, for instance). The distinction between the specific activities in a project with activities that take part in the organization's common activity flow (here called enterprise flow) can be observed. This way, a project manager must deal with this decoupling amongst the activity flow of a software project and other activity flows of the organization which provide some support to the project. Both types of workflows are executed in parallel, have their own resources and may influence the timing of activities and project costs. Therefore, the project manager needs some kind of support to help in the process of decision making taking into account the integration of these different streams of activities during the simultaneous execution of software projects.

Based on these premises, a study was carried out to determine the state of the art of dynamic reconfiguration of software projects, with emphasis on the integration of project management and the organizational flows, in order to identify the main gaps and challenges of research in this field. To reach this goal a methodology of systematic review was adopted. Systematic review is a research practice often used in the medical field, which was adapted to software engineering by [10] and [19]. This process helps to establish scientific rigor which is necessary to define the state of the art and to produce more reliable results.

This paper is organized as follows: Section II starts with an overview of dynamic reconfiguration of software projects to establish a background in this field. Section III presents the Systematic Review research methodology applied in this work. Section IV describes the plan of systematic review and in section V and VI the results and discussions of the systematic review are presented. Finally, in section VII the findings and suggestions for future works are described.

## II. DYNAMIC RECONFIGURATION OF SOFTWARE PROJECTS

According to [21], a project is a temporary endeavor with the goal of producing a single product or service. Usually a project is directed to achieve a specific result and involves the coordinated implementation of inter-related activities. More than that, projects are planned, executed and controlled by people, and are constrained by limited resources. In turn, project management is responsible for monitoring the achievement of project objectives through the application of a

set of techniques and tools [22]. Then, project managers need some kind of support, usually based on a project management methodology, to deal with different design variables, responsibilities and tasks. For this purpose, there are several proposals in the literature or practice already carried out in companies. However, software project management is not an easy task to be performed. Software projects are very dynamic and require recurring adjustments of their project plans. These settings can be understood as reconfigurations in the schedule, in the resources allocation and other design elements. A software project plan specifies and limits the scope of the project, describes the possible risks of the project, defines the capabilities of available hardware and software, describes the work breakdown structure and project schedule [23]. According to [24], the project schedule specifies dependency relationships, estimated time required to reach each milestone of the project and the allocation of people involved in each activity. In this scenario, a project configuration involves not only the planning of activities, but also the planning of available resources, their characteristics and how these resources are allocated to activities. Project-level information (such as the priority over other projects) and the availability of resources are added to it. However, we should consider the changes on the project plan, specifically those that occur during project execution. Thus, dynamic reconfiguration of projects is the successive planning for the same project during its execution. The planning of a project also involves the selection and definition of necessary activities in the project schedule. Since the sequencing and dependency of activities in a project determine the order in which they shall be held and also activities that can be performed in parallel.

Also, during the planning and execution of a software project, the project manager may need to interact with other departments within the organization to obtain relevant information to the project [8]. The project manager, for example, may need to contact the human resources department about the need of hiring staff for a particular software development project.

Therefore, it can be seen in Fig. 1 that the flow of specific activities of a software project can interact with other common activity flows of the organization (enterprise workflow). Both types of workflows are executed in parallel, have their own resources and may influence the timing of activities and costs of software design.

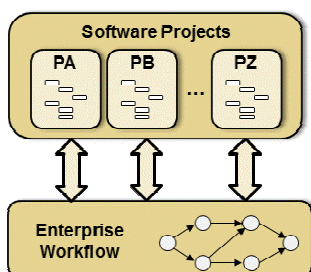


Fig. 1 Relationship between projects and organizational flows

Due to this separation between the activity flow of a software project and other flows of the organization's activities, there may be a relationship of dependency between activities belonging to these two types of workflows. For example, the activity of developing a web site (part of the software project workflow) may depend on the acquisition of a web server by the financial department (part of the company's organizational flow). The disregard of the dependencies among the activities of different workflows can result in software project planning distortions.

It can be seen, therefore, that some types of management activities are inherent to the process and do not appear at the time of project planning. It is precisely these activities (or their dependencies) that most often cause a delay in the schedule and are not considered in the definition of project risks.

Therefore, it is crucial to conciliate these two visions: the project manager must deal with both management issues and production during the planning and execution of projects, considering not only the activities which produce direct results on the software project, but the activities which belong, in a shared way, to the other support activity flows to the company's projects [8].

### III. SYSTEMATIC REVIEW IN SOFTWARE ENGINEERING

Systematic review is a research methodology that uses systematic methods to *identify*, *select*, and *critically evaluate* scientific studies in a specific field of research. It is a planned review to answer a specific question that can or not include statistic methods. Statistic methods used in the analysis and synthesis of the selected studies are called meta-analysis [20]. In this work, the meta-analysis stage was not adopted since it is a qualitative diagnosis of studies.

The systematic review is a way to evaluate and interpret the relevant work to a specific research question [9], [10]. Briefly, in a systematic review one carries out a search for relevant information to the research, defining "search strings" to be performed on mechanisms, such as those provided by IEEE and ACM. This information is generally reported through articles in conference proceedings, journals, and technical reports or even in books devoted to each area. The advantages of a systematic review, can be summarized as follows:

- assemble the existing evidence on a particular subject in the literature;
- identify any gaps that may point to future researches;
- provide a consistent overview to propose new research activities adequately.

The systematic review, give a scientific rigor to a literature review process and, as a consequence, minimize the slants that can happen during a conventional literature review. The guidelines to lead the process of systematic review established by [10] and [19] were adapted to reflect the specific problems of research in Software Engineering. These guidelines are composed by three stages: planning of the review; conducting the review; and reporting the review.

IV. SYSTEMATIC REVIEW PLAN

First of all, a systematic review should start with the definition of the research question. In this work we have adopted the term Question Focus (*QF*) as a way to represent the research question. To determine the *QF* some complementary issues were used. The next subsections summarize the sequence of steps that were established by [10] and [19]. In the section V the third stage is represented through de results analysis.

A. Research Scope

Considering the general scope of the research, the goal is to find solutions for the following questions (*Q<sub>i</sub>*):

- *Q<sub>1</sub>*: Which are the existent approaches and solutions related to dynamic reconfiguration of software projects' planning networks?
- *Q<sub>2</sub>*: How Software Project Processes are prepared to manage multiple projects simultaneously?
- *Q<sub>3</sub>*: Which are the project management methodologies that support the integration of the software project workflow with other organizational flows?

These questions were used to delimitate the scope that will effectively be answered by the process of systematic review established in this work, through the analysis and synthesis of the selected studies. Based on the pre-identified questions, the *QF* was defined as follows:

- *QF*: "Which existing approaches in the literature allow the dynamic reconfiguration of software projects' planning networks involving multiple simultaneous projects and integration with other organizational flows?"

The *QF* is essential to determine the structure of the review. If the *QF* is not well defined, it could substantially compromise the result of the research. All the steps of the systematic review were guided by the *QF*, which was also used as a way of judgment of the systematic review relevance.

B. Systematic Review Details and Protocols

After the definition of the *QF*, the next steps are, respectively, the selection of research sources, the definition of inclusion and exclusion criteria and the selection of the studies found. When this planning is completed, you should evaluate the protocol plan. If approved, the extraction of information will begin (see Fig. 2). Results should also be evaluated prior to performing the final analysis. In order to keep all information and document the decisions made by the researchers, aim is to make the storage of results throughout this process.

This systematic review uses the following search engines: IEEE Xplorer digital library, ACM digital library, Springer Link and Science Direct. The criteria to make the decision about the selected search engine were: (1) the database allow search engines based on key words and Boolean expressions; and (2) the availability of articles through Internet.

The selection of papers occurred in 3 months, between March and May 2011. The population defined for the study includes articles published in journals and conferences on the field of study since 2000 until now and that were written in English language. This range of years was chosen in order to perform a filter on the recent works in the area. The justification for the choice of English language is due to its universality, remaining itself the standard language of conferences and international journals. Then, several papers were analyzed in the areas of computer science, software engineering, project management, information systems, support systems for decision and management of business processes. The sources were exclusively accessed on the web, so manual search was not considered in this work.

Initially, for the definition of the "search string", several combinations and variations of the following key words were tried: "project management", "scheduling", "business process", "workflow", "managerial activity", "enterprise activity", "organizational activity" and "productive activity". After observing the behavior of search engines, these key words were combined through Boolean operators and filters, as presented in Table I.

TABLE I  
SEARCH STRINGS

ID	Search String
S <sub>1</sub>	(project management OR scheduling) AND (business process OR workflow) AND (managerial activity OR enterprise activity OR organizational activity OR productive activity) AND (year >= 2000 AND year <= 2011)
S <sub>2</sub>	(project management) AND (business process) AND (managerial activity OR enterprise activity OR organizational activity OR productive activity) AND (year >= 2000 AND year <= 2011)
S <sub>3</sub>	(scheduling) AND (workflow) AND (managerial activity OR enterprise activity OR organizational activity OR productive activity) AND (year >= 2000 AND year <= 2011)
S <sub>4</sub>	(dynamic reconfiguration) AND (business process OR workflow) AND (year >= 2000 AND year <= 2011)

It must be observed that some used "search strings" returned a lot of documents. The 132 articles selected were resulting of the string S<sub>1</sub>. The string selects articles from journals and conference published since 2000 which deal with software projects' dynamic reconfiguration emphasizing the integration of project management with the organizational flows.

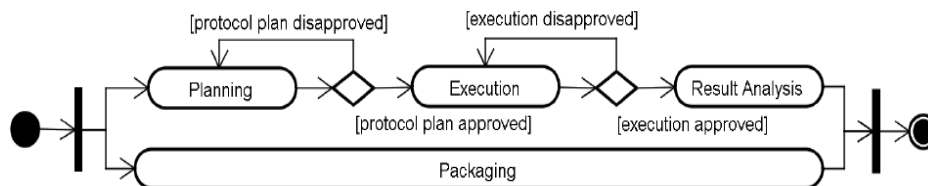


Fig. 2 Process for Systematic Review [10]

## V. RESULTS

The 132 articles were preliminary selected for a future analysis. Hence, this selection was based on the following sequence of steps:

- the search string was run in each of the search engines;
- the title and abstract of each article were read;
- when initially approved, the full texts were read for a final approval;
- if in doubt due to lack of clarity in the abstract, a quick reading of the text was conducted;
- the remaining articles were selected for full reading.

The implementation of this sequence of steps resulted in the selection of 24 works. Fourteen of them were later dropped due to the lack of minimal relevance expected. Thus, the final selection included 10 articles (7.57% of the overall results of the search engines). According to [9] and [10], this reduction is expected in a systematic review, mainly due to selection errors from the search engines.

The main criteria for selection of the surveyed papers was accepting only those which included a good level of description on the solution and enough information about the methods or strategies of any kind to resolve the issues related to the research topic of this article. More specifically, information on the following items was sought:

1. *Activity programming (scheduling)*: indicates if the article presents a solution to the scheduling of activities on the project schedule;
2. *Concept of management activities of support*: indicates if the article presents a distinction between project's specific activities from those that are common to every organization;
3. *Integration with organizational flows*: indicates whether the solution provides integration of the flow of project activities with the organizational flows of the company;
4. *Multi-projects support*: indicates if the solution supports more than one project simultaneously;
5. *Type of solution*: it shows the solution's "general category" (e.g. decision support, optimization and methodology.);
6. *Method*: the method used for the solution (e.g. Bayesian networks and dynamic programming.);
7. *Dynamic solution*: indicates whether the solution can provide immediate feedback during the course of the project;
8. *Use of simulation*: indicates whether the solution involves some kind of computer simulation;
9. *Evaluation by the scientific community*: Indicates whether the research was scientifically evaluated as a case study or experiment;
10. *Tool*: indicates if the solution includes a tool or prototype.

The classification of the studies based on these criteria is presented in Table II, and the discussion in section VI.

## VI. DISCUSSION

*By analyzing Table II, one can get some conclusions based on a quick quantitative analysis:*

- First, all selected works present solutions to the scheduling of activities on the project schedule;
- Four of the analyzed articles present a distinction between the technical and managerial activities;
- Only two articles present solutions that provide integration of the project's activity flow with the organizational flows of the company;
- Four papers deal with a multi-project scenario;
- It is interesting to note that all results show solutions that involve some kind of dynamic solution;
- Four studies presented solutions that involve some kind of computer simulation;
- An important observation is that more than half of the selected works have some kind of tool or prototype. However, only one study includes results from a case study or experiment.

The planning of activities is strongly affected by uncertainties and external events [7], [18]. The idea of using contingency plans, suggested later by this author, seems very promising and combats much of the subjectivity present in the daily decisions made by software project managers. Contingency plans are pre-encoded sequences of actions that must be performed once a problem occurs. So this is a dynamic solution. In other words, facing a problem in the current configuration, it is allowed to adopt different solution strategies for each situation presented. The work presented by [15] contains a set of integrated tools for modeling, analysis and management of systems. The tool PROSIM, for example, provides mechanisms for modeling, analysis and design of business processes. It provides a graphical environment to model business processes and then perform simulations of each process. Another tool, the ProjectLINK, an add-on for PROSIM, allows information from a process model to be transported to a project management tool, for instance, MS-Project. In [12], is shown ProPlanT, a multi-agent tool that allows planning of production activities and resource selection (based on mechanisms subscribe/advertise). Yet, within the systematic review, the only two studies that show the integration with organizational flows are [1] and [7]. The first presents a model integrating the PMBOK with RUP, conceptualizing managerial and productive activities and the interdependence relationship among these two types of activities. The second only presents an example of temporary employment of resources. Next to the idea of integrating the activities of a software project with flows of organizational activities, the article presented by [2] works with the concept of policy or policies. His work presents some strategies that the manager can take to manage your project, for example, allocating more resources to an activity, making use of multi-tasking to a developer, creating internal milestones, among others. The idea presented tolerate simultaneous projects, but

TABLE II  
RESULTS OF THE SURVEYED PAPERS

Source	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6	Item 7	Item 8	Item 9	Item 10
[1]	Yes	Yes	Yes	Yes	Decisions support	Integration Model	Yes	No	No	No
[11]	Yes	No	No	No	Decisions support	Mathematical Model	Yes	No	No	No
[2]	Yes	No	No	Yes	Decisions support	Sceneries	Yes	Yes	No	Yes
[7]	Yes	Yes	Yes	No	Decisions support	Search in space of strategies	Yes	Yes	No	No
[12]	Yes	No	No	No	Decisions support	Multi-agents	Yes	Yes	Yes	Yes
[13]	Yes	No	No	Yes	Framework	Workflow	Yes	No	No	Yes
[14]	Yes	No	No	No	Framework	Workflow	Yes	No	No	Yes
[15]	Yes	Yes	No	Yes	Decisions support	Toolkit	Yes	No	Yes	Yes
[16]	Yes	No	No	No	Framework	Workflow	Yes	No	No	No
[17]	Yes	Yes	No	No	Decisions support	Model	Yes	No	No	No

these may have different characteristics (for example, the first project may be executed based on an iterative process, while the second may be based on the waterfall model).

Through this analysis, it is observed that the surveyed works do not have a solution that allows the planning of software projects considering the interactions of the project manager with other departments within the organization. The disregard of the dependencies among the activities of different workflows can result in distortions in software project planning. Often the project manager only sees the need for soliciting previously requested information from another department within the company when performing a particular project activity that depends on other departments (e.g. purchasing equipment or hiring a new developer). Consequently, there is a need for a solution to anticipate the needs arising from the areas of support during the project planning software. This solution should consider the complexity of identifying the interdependence between the activities of the organization's workflows and projects' workflow. However, none of the jobs returned in this systematic review provides a full computational mechanism to resolve these issues.

In a previous study (see [8]) we have presented an integrated model called SPIM - Software Planning Integrated Model. The SPIM model was designed considering the need of project managers to access information from other departments of the organization during the software project planning. To support this functionality, this model defines three different types of activities:

1. *productive activities*: activities directly related to the construction of the software product;
2. *managerial activities*: activities that are only required to coordinate the construction of the software product;
3. *management supporting activities*: any other activities that do not belong to an individual project's activity workflow (and may be else shared by other projects).

The database modelling of a software application is an example of productive activity. Organize and conduct a follow-up project meeting is an example of managerial activity. This first two types of activities belong to the project's workflow. Hire a database administrator (activity usually performed by human resources department) is an example of management supporting activity. Following this definition, it is possible to distinguish which activities should be updated by other sectors of the organization (using a

mechanism such as a workflow) and which should be updated directly by the project manager. A tool called Software Planning Integrated Tool (SPIT) was developed to demonstrate the feasibility of the concepts proposed by the model SPIM. However, SPIT is still a prototype and needs to be evaluated on real projects in software companies.

## VII. CONCLUSIONS AND FUTURE WORKS

This paper presented some results from a systematic review of the dynamic reconfiguration of software projects, with emphasis on the integration of project management with organizational flows. Based on the obtained results, the ability of these works to resolve the 10 issues related to the research topic of this article was examined.

Due to the small number of papers returned in the research, one notes that it was possible to perform only a few comments on the dynamic reconfiguration of projects, considering the integration of project management with organizational flows. Even the more recent works do not present a solution that addresses all the problems at the same time. One reason may be the type of search string that was used in the systematic review's preparation: by the moment of search string definition, one automatically ends up restricting the returned articles and assuming the risk of leaving out some related works. Another possible reason is that this is a subject that has not been deeply addressed by other researchers.

Analyzing the current literature, we cannot identify studies that deeply address the subject and present specific solutions that provide some kind of integration of software project with organizational flows. The activities pertaining to organizational workflows use resources that are not allocated directly to the software project. However, these features may influence both in terms of activity deadlines and software project costs (e.g. if the doctor responsible for the entrance examination needs to get away for a few days, this delay may negatively impact the schedule of software projects).

The identification of the interdependence of the company workflows and software project during the project planning is not an easy task. Thus, the project managers need a support enabling them to avoid distortions in project planning (such as increased costs and delays in project timelines) in disregard that support activities pertaining to workflows of organizations use resources that are not directly allocated to the software project. But, according to the results of this research, we can

say that the problem presented in  $QF$  is far from being answered by current approaches.

Anyway, the collected information was sufficient to proceed with this research, since these facts indicate the need for further investigations and solutions for this area. As future work, we intend to develop a protocol to evaluate the model SPIM with software companies, using the prototype SPIT in real software projects.

#### REFERENCES

- [1] D. Callegari, and R. Bastos, "Project Management and Software Development Processes: Integrating RUP and PMBOK", ICSEM - International Conference on Systems Engineering and Modeling, 2007.
- [2] B. Lee, J. and Miller, "Multi-Project Management in Software Engineering Using Simulation Modelling", Software Quality Journal, vol. 12-1, pp. 59-82, 2004.
- [3] T. Abdel-Hamid, and S. E. Madnick, "Software Project Dynamics: an Integrated Approach", Englewood Cliffs, Prentice Hall, 1991.
- [4] H. Kerzner, "Project management best practice: achieving global excellence", New Jersey: John Wiley & Sons Inc, 2006.
- [5] R. Pressman, "Software engineering: a practitioner's approach", McGraw-Hill, 6<sup>th</sup> edition, 2004.
- [6] D. Callegari, and R. Bastos, "A Multi-Criteria Resource Selection Method for Software Projects using Fuzzy Logic", ICEIS - International Conference on Enterprise Information Systems, pp. 376-388, 2009.
- [7] D. Joslin, and W. Poole, "Agent-Based Simulation For Software Project Planning", In: Proceedings of the 37<sup>th</sup> Conference on Winter Simulation, Orlando, Florida, pp.1059-1066, 2005.
- [8] D. Callegari, M. Rosito, M. Blois, R. Bastos. "An Integrated Model for Managerial and Productive Activities in Software Development", In: ICEIS - 10th International Conference on Enterprise Information Systems, Spain, 8p, 2008.
- [9] B. Kitchenham, "Procedures for performing systematic reviews", Joint Technical Report Software Engineering Group, Department of Computer Science, Keele University, 33p, 2004.
- [10] J. Biolchini, P.G. Mian, A.C.C. Natali, and G.H. Travassos, "Systematic review in software engineering". Rio de Janeiro, 31p, 2005.
- [11] Y. Chen, M. Fang, "Research on Resource Scheduling for Development Process of Complicated Product". In: Proceedings of the 9<sup>th</sup> International Conference on Computer Supported Cooperative Work. Coventry University, UK, Vol. 1, pp.229-233, 2005.
- [12] A. Riha, M. Pechoucek, H. Krautwunnova, P. Charva, A. Koumpis, "Adoption of an Agent-Based Production Planning Technology in the Manufacturing Industry", In: 12<sup>th</sup> International Workshop on Database and Expert Systems Applications, pp. 640-646, 2001.
- [13] C.L. Zhao, X.G. Ming, X.H. Wang, and D. Li, "A Framework of Supplier Involved Collaborative Project Management", 1<sup>st</sup> International Conference on Information Science and Engineering, pp. 4130-4135, 2009.
- [14] T. Aye, and K. Tun, "A Collaborative Mobile Agent-based Workflow System". In: 6<sup>th</sup> Asia-Pacific Symposium on Information and Telecommunication Technologies, pp 59-65, 2005.
- [15] D. Delen, P. Benjamin, and M. Erraguntla, "An Integrated Toolkit for Enterprise Modeling and Analysis", In: Proceedings of the 1999 Winter Simulation Conference, pp. 289-297, 1999.
- [16] D. Diwakar, and S. Diwakar, "CINWEGS-An Integrated Web and Grid Services Framework for Collaborative Solutions", In: 4<sup>th</sup> International Conference on Next Generation Web Services Practices, pp. 21-27, 2008.
- [17] J. Grudin, "Managerial Use and Emerging Norms: Effects of Activity Patterns on Software Design and Deployment", In: Proceedings of the 37<sup>th</sup> Hawaii International Conference on System Sciences, pp. 1-10, J. 2004.
- [18] P. Jalote, A. Palit, P. Kurien, V.T. and Peethamber, "Timeboxing: a process model for iterative software development", Journal of Systems and Software, vol.70, 1-2, pp.117-127, 2004.
- [19] J. C. de Almeida Biolchini, P. G. Mian, A. C. C. Natali, T. U. Conte, and G. H. Travassos, "Scientific research ontology to support systematic review in software engineering", Adv. Eng. Inform., 21(2):133-151, 2007.
- [20] J. P. Higgins and S. Green, editors. "Cochrane Handbook for Systematic Reviews of Interventions Version 5.0.1 [updated September 2008]", The Cochrane Collaboration, 2008.
- [21] Project Management Institute, "PMBOK - A Guide to the Project Management Body of Knowledge" Newtown Square, PA: Project Management Institute, 4th edition, 2008.
- [22] K. Schwalbe, "Information Technology Project Management", Thomson Learning, 2nd edition, 2002.
- [23] J. Lee and N. Lee. "Least modification principle for case-based reasoning: a software project planning experience". In Expert Systems with Applications, volume 30, Issue 2, pp 190-202, 2006.
- [24] I. Sommerville, "Software engineering", Addison-Wesley, 8th edition, 2006.