An Experimental Study on the Dynamic Reconfiguration of Software Projects

Maurício Covolan Rosito[1], Marcelo Blois Ribeiro[2] and Ricardo Melo Bastos[1]
[1]*Pontifícia Universidade do Rio Grande do Sul (PUCRS), Porto Alegre, RS, Brazil*
[2]*GE Global Research, Porto Alegre, RS, Brazil*

Keywords: Dynamic Reconfiguration, Empirical Evaluation, Organizational Workflows, Software Project Management.

Abstract: Developing a software product is a complex activity that involves many uncertainties. Software projects usually experience many modifications during their execution phases. These adjustments can be understood as reconfigurations in the schedule, in the resources allocation and other design elements. The large amount of information that the project manager must deal, combined with the frequent changes in the scope and planning, makes this activity more challenging. In addition, the manager may need to consult other departments in the organization during the execution of a software project. The distinction between the specific activities in a project with activities that take part in the organization's common activity flow can be observed. In order to contribute to the solution of the noted difficulties, it was proposed a computational model called SPIM. In this sense, this article presents the results of an experimental study related to dynamic reconfiguration of software projects, with emphasis on the integration of project management with organizational flows. A software tool was built to demonstrate and evaluate the results.

1 INTRODUCTION

Software development requires planning and execution of activities, in which it is necessary to deal with both technical and managerial issues. 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 (Kerzner, 2000); (Pressman, 2009). Particularly in software projects, aspects such as uncertainties in the specification of requirements and their instability throughout the project development, in the use of applied technology and human nature itself potentiate these difficulties.

During the planning and execution of software projects, different types of tasks are assigned to resources with different characteristics in order to reach the goals related to time and costs of these projects. In response to new information or estimations, managers may need to make changes to the project plan, such as reallocating resources or canceling tasks (Joslin and Poole, 2005). These adjustments, required for the project according to over time changes, give rise to the term 'project reconfiguration'. More recently, research in this area has addressed this problem from a dynamic perspective, through which the projects adapt themselves during their implementation. Such changes often determine impact on costs and previously established deadlines of the project.

In the same scenario, the manager may need to interact with other departments of the organization during the execution of software projects in order to obtain relevant information to a specific project (for example, contact the finance department). Thus, the distinction between the specific activities in a project with activities that take part in the organization's common activity flow (here called organizational flow) can be observed. 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 projects.

In order to contribute to the solution of the noted difficulties, it was earlier proposed a computational model called Software Planning Integrated Model (SPIM). The SPIM allow supporting dynamic reconfiguration of software projects considering the planning and replanning of their activities. To evaluate the model and embodiment of the proposal,

Covolan Rosito M., Blois Ribeiro M. and Melo Bastos R..
An Experimental Study on the Dynamic Reconfiguration of Software Projects.
DOI: 10.5220/0004442902320239
In *Proceedings of the 15th International Conference on Enterprise Information Systems* (ICEIS-2013), pages 232-239
ISBN: 978-989-8565-60-0

a software tool was developed and used in an experimental study.

2 PROJECT MANAGEMENT

The development of a software product requires an effort that involves dealing with, among other things, activities and resources to produce the desired results (Schwalbe, 2010). Generally, a project is intended to achieve a specific result and involves the coordinated implementation of interrelated activities. More than that, projects are planned, executed and controlled by individuals, and are restricted by limited resources.

Software companies often make use of project management knowledge in order to build their solutions with quality and within scope, time and resource constraints. Project management practices are responsible for monitoring the achievement of project goals through the application of a group of techniques and tools. Thus, project managers need some kind of decision making support, usually based on a project management methodology, to deal with different responsibilities, tasks and project variables.

During a project's lifetime, actual data, such as the time or resources that were spent to perform a particular task, are collected and entered by the project manager. The manager usually creates a project plan to specify and limit the scope of the project describing the work breakdown structure (WBS) and the project schedule. When creating a project schedule, the manager begins with a set of tasks in the WBS (Pressman, 2009). Then, he specifies all project-related information, such as the individual tasks, the execution's sequence of these tasks and the resources to perform these tasks.

However, the manager may not have all relevant information up front, forcing him to interact with other departments in the organization (such as the human resources department). Hence, the flow of activities in an individual project is usually related to other common activity flows of the organization. Both types of flows are executed in parallel, have their own resources and may influence the timing of activities and costs of software design (see Fig. 1).

It can found potential dependency relations between the activities in both workflows. As an example, the activity of developing a web site (which fits in the project's workflow) may depend on the hiring of staff by the responsible department (this activity fits in the shared workflows of the company). Consequently, it was observed the need for a decision support solution to anticipate these requirements related to the other departments of the company during the execution of software projects.

3 SPIM: AN INTEGRATED MODEL

The SPIM was first developed considering the integration of project management concepts provided in Project Management Book of Knowledge Guide (PMBOK) (Project Management Institute, 2008) with the concepts of software development provided in Rational Unified Process (RUP) (Kruchten, 2000) and in Object-oriented Process, Environment and Notation (OPEN) (Graham et al., 1997). Details about these integrations models can be seen in Callegari et al (2008) and Rosito et al (2012). The detailed study of the PMBOK, RUP and OPEN metamodels helped to identify how their classes are organized and which are the valid relations between the elements of each model. It allowed the development of a methodology for integrating models of project management with models for software development processes.

A first experimental study using SPIM was conducted by six undergraduate and four postgraduate students of computer science (see Rosito and Bastos, 2012). This experiment reveals that the use of the SPIM model approach help managers to create and conduct a more precise project plan than the traditional method. However, some limitations in this first experiment were observed (such as the use of graduate students) and advances in research are described below.

Process Engineering Meta-Model Specification (SPEM) (Object Management Group, 2011), is considered the reference metamodel for defining software processes developed by the Object Management Group. Thus, this research has advanced to creating the metamodel PMBOK+SPEM.

In the PMBOK+SPEM metamodel (see Fig. 2), the Organization class represents a company that is organized by programs. The organizations, usually, divide projects in several phases aiming a better managerial control. Also, a necessary resource for the project, such as people or equipment, is represented by the Resource class. These resources are divided into active resources (Stakeholder class) and non-active (PhysicalResource class). Then, the ActivityPhysicalResourceWork class associates physical resources to activities. It establishes the physical resources work load in that activity.

The PMBOK+SPEM metamodel was designed

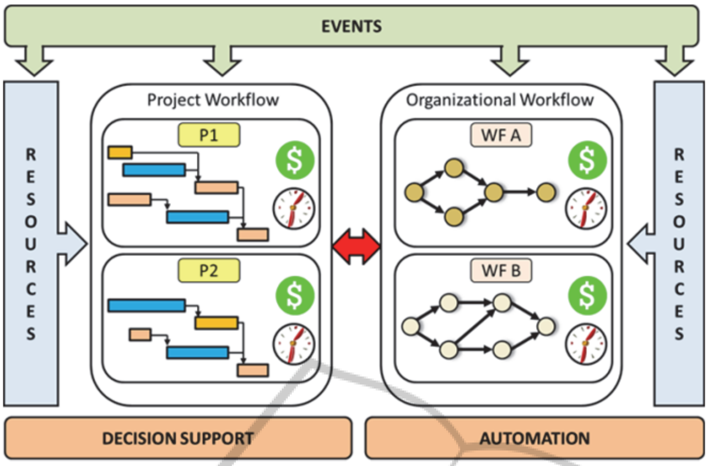

Figure 1: Interdependency between an organizational workflow and a software project workflow.

considering the need of project managers to access information from other departments of the organization during the software project planning. Then, this metamodel defines three different types of activities: (a) productive activities: activities directly related to the construction of the software product; (b) managerial activities: activities that are only required to coordinate the construction of the software product; and (c) management supporting activities: any other activities that do not belong to an individual project's activity workflow.

Stakeholders can play several roles during the execution of project activities. Thus, for each association between a role and activity (ActivityStakeholderWork class) there must be an association of this activity with a stakeholder able to play that role. Then, managerial activities are performed by managerial roles and productive activities are performed by productive roles.

According to the SPEM metamodel, an activity supports the nesting and logical grouping of elements contained in the WBS (BreakdownElement class). The self-defined relationship to the Activity element allows reuse of content set to an activity in another activity. Thus, it becomes possible to inherit a structure defined for an activity in terms of its elements nested in a second activity. The relationship between the Activity class and the ProcessParameter class establishes input and/or output parameters to the activities in terms of work products. The ProcessPerformer class establishes the relationship between the activities and the roles in the project. The ProcessResponsabilityAssignment class establishes the responsibility relationship between the roles and work products.

The PMBOK+SPEM metamodel helped to provide the conceptual framework necessary to develop a unique model, called SPIM, to assist in project planning considering the concepts arising from the software development processes. Also, it was developed a tool called Software Planning Integrated Tool (SPIT). SPIT aims to offer some kind of support to help managers in the decision making process of software projects through the concepts proposed by the SPIM. In this experiment, participants had access to the following modules: Validator, BackOffice and Workflow Integrator.

The SPIM Validator acts as an add-in for Microsoft Project and performs the SPIM validation rules on software projects. The BackOffice is responsible for managing the information required by the SPIM, such as roles definition, types of activities and associated work products. This information is exported to Microsoft Project through custom field to be used by the SPIM Validator. The Workflow Integrator module is responsible for synchronizing the information contained in organizational workflows with those present in a specific project. Currently, the organizational workflows were developed through the Visual Studio Workflow Designer tool.

4 EXPERIMENTAL STUDY

To perform the evaluation of models and products where the human factor is considered, the literature provides some approaches based on an experimental strategy. Pfleeger and Atlee (2009) suggest the following approaches to evaluate processes, products and resources: feature analysis; case studies, surveys, and experiments. Experiments represent a more controlled type of study, usually

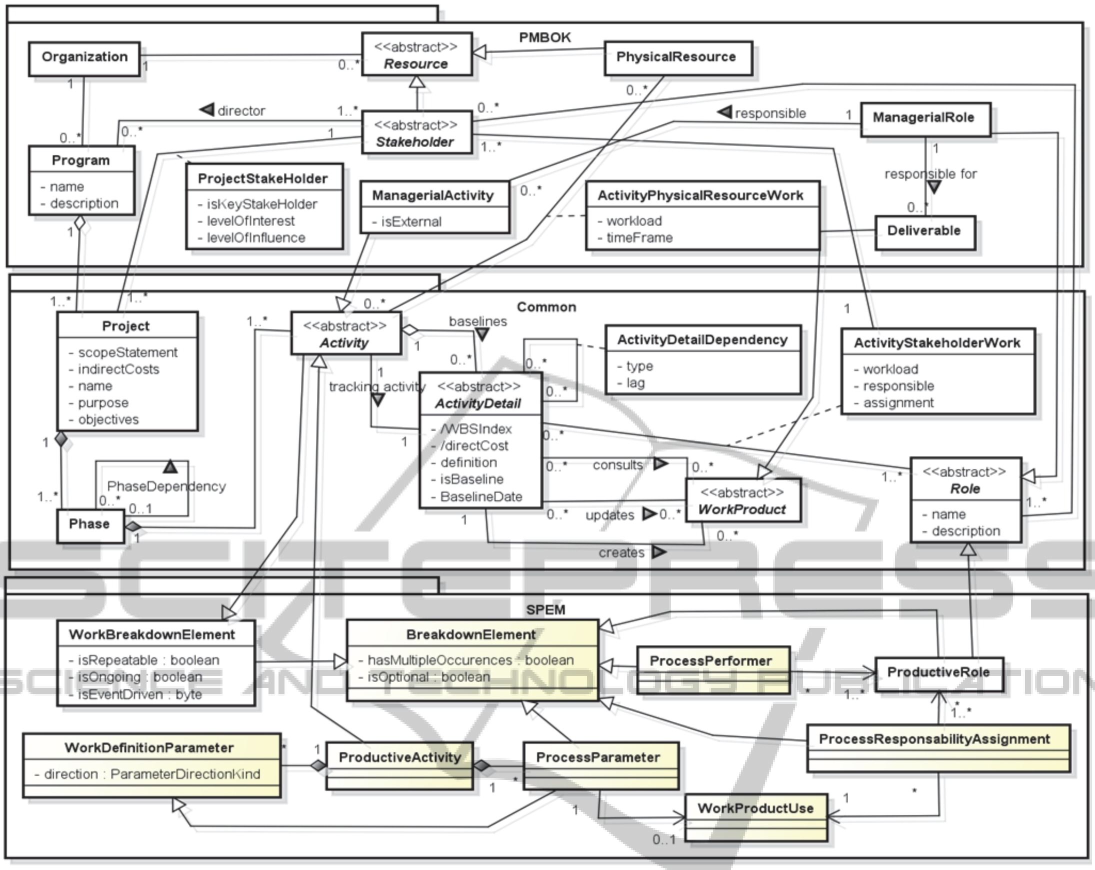

Figure 2: Part of the PMBOK+SPEM metamodel (more details not shown due to space reasons).

conducted in laboratories. In this research, the use of a formal experiment method was chosen. However, as the experiment has a quantitative approach (Wohlin et al. 2000), an integrated survey was also used in order to evaluate qualitative data. The proposals of Juristo and Moreno (2003), Field (2005) and Wohlin et al. (2000) were used as guides to conduct this experiment.

4.1 Objective Definition

The Goal-Question-Metric technique (GQM) (Solingen and Berghout, 1999) was used to define the study, establishing the overall goal, the objectives and the measurement. It was decided that the purpose of this research is to compare, in the Unified Process, the accuracy and the effort of integrated planning model SPIM compared to the traditional model of software project planning.

4.2 Design

In this stage the researchers should formalize the hypotheses, determine the independent and dependent variables, selection of participants, preparation of the experiment and the conceptual consideration of the validity of the experiment. Then, these researchers selected an 'in-vitro' and 'offline' approach in which participants performed the experiment in a controlled environment. To conduct this experiment, the context involving students of two distinct universities was chosen. This approach can reduce risks and costs not covered by the scope of the research at this time. Thus, the experiment was conducted by thirty six students of post graduation courses in Project Management. After that, based on the previous informal definition of the two issues in this research, it was possible to formalize the two hypotheses and a definition of its measures for evaluation.

The first hypothesis is related to the effort of managers in planning activities and resources for software projects. Then, the first null hypothesis (H_0) is as follows: the effort involved in planning the activities of the software project using the SPIM integrated model is equal to the effort to do the planning of activities according to the traditional model. The effort should be measured by time spent in minutes with the planning of activities for software development projects in each approach.

The second hypothesis of this research is related to the accuracy of managers to plan the activities and resources in software projects. So, the second null hypothesis (H_0) is as follows: the accuracy in the schedule planning of projects regarding to the assignment of deadlines and resources considering the integration with the organizational flows through the SPIM integrated model is equal to the accuracy accomplishing the planning in the traditional model. The accuracy should be evaluated by the ratio of the participants' score and the total score possible.

Considering that experimental units are the objects upon which the experiment is run, five different scenarios of software development project were created, aiming to approach different software project risks. The first scenario is related to the assignment's compatibility of the involved stakeholder role with the type of activity (managerial or productive). The second scenario is related with the interaction among the organizational flows to the acquisition of new hardware during the project. The third scenario is related to the risk of identifying that the most qualified staff is unavailable at critical moments. The fourth scenario is related to the situation where new employees are hired and must perform some sort of training before joining the project. In the fifth scenario, the manager identifies that software components purchased from a third party contain defects which limit their functionality, so that he should contact the product's supplier.

The outcomes of this experiment (response variables) are concerned to the effort and accuracy in planning activities of software projects. Also, any project characteristics (called factors) intentionally varied were identified during experimentation. Each factor has several possible alternatives. In this experiment, there is one factor to be analyzed (project planning methods) and two alternatives: the traditional method of project planning and the method using the integrated planning model SPIM. Considering the characteristics of this research, the one-factor designs were chosen. This sort of design involves comparing the variable response to each alternative in a given number of experimental units. Some characteristics, however, would be desirable to be invariant, but they vary in an experiment (blocking variables). Project management (much more than other software development task) is an activity which performances highly depends on the person who does it, hence the risk that results are highly influenced by ability and experience is high. In this experiment, the level of experience in project planning is a blocking variable.

If both alternatives are used in the same project, two similar teams are required. The definition of which participants would perform each approach of software project planning (in the traditional way or with the help of SPIM model) occurred randomly. In this case, the experimenter took thirty six cards (half red and half black) from the pack; the red cards would correspond to the use of the traditional project planning method and the black ones to the use of the SPIM method. The experimenter shuffled the cards and allowed each subject to take a card for each experimental unit (software development project). The balancing principle was also used so that each propose of software project planning was performed by the same number of participants (eighteen participants to each proposal).

4.3 Execution of the Experiment

The realization of the experiment occurred in December 2011, when the set of participants performed the experiment in a controlled environment (university's computer lab). Initially, all participants received an email inviting them to join this experimental study. In this invitation it was explained that this event included a presentation of SPIM and the realization of a practical activity where participants would have the opportunity to perform exercises based on typical situations of project management. The experiment involved only students that had some interest in the area of project management. To take part in this event the invitees had to access the link to the event and create an access account. Thus, a web site was developed in order to store the questionnaires of this experiment aiming to maintain the integrity of the data obtained during its execution. An access control system ensured that each participant had access only to questions that have been designated for them.

The problem studied corresponds to five scenarios that simulated situations in software development projects. At first, all participants received a brief training in the SPIM model and had the opportunity to test the main features of the SPIT on a sample project. Later, they had the opportunity to make the first questions about the proposed work. Then, they were presented to the same description of each scenario and were asked to perform the corresponding project planning - some using the traditional method and others with the SPIT tool. In order to avoid possible distortions in the obtained results both in the trial of SPIT and the questionnaire's resolution phase it didn't occur having any interaction with the interviewer.

4.4 Analysis

There are different analysis techniques depending on the characteristics of the collected data and on the applied design. The methods of analysis can roughly be divided into two major blocks: parametric and non-parametric methods. According to Miles and Huberman (1994), parametric tests are statistically more powerful than non-parametric methods. However, if these parametric tests are not conclusive, then the analysis will have to resort to the application of non-parametric tests. Considering these two types of analysis techniques, the drawing of conclusions was attempted by rejecting the null hypotheses with the parametric test or/and accepting them with the non-parametric test.

For the testing of hypotheses, in a context of one factor and two treatments, the literature suggests the significance test called 'T test' for two independent samples (if performed a parametric test) or 'Mann-Whitney test' (if it is a non-parametric test). This definition was taken after verifying if the distribution was normal or not (by the 'Shapiro-Wilk test') and checking the variance of the data obtained by running the experiment ('Levene Test').

5 ANALYSIS OF RESULTS

According to the scope of this research, it was necessary to evaluate these two hypotheses: effort and accuracy. For the hypothesis analysis of this research, we used the T test (suitable for comparing the averages of a quantitative variable between two independent groups) or Mann-Whitney test (if the test is non-parametric). Then, the verification of each null hypothesis for each developed scenario was performed. The null hypothesis (H_0) is related to the randomness of the observed results, that is, if it is true, statistically the results of the experiment evidence to be occasional (no conclusion can be drawn). The alternative hypothesis (H_1) is one that will be accepted if the null hypothesis is rejected. Still, it must be noted that the level of the test significance (p-value) was fixed in 5%. The analyses presented in this experiment were made using the Statistical Package for Social Sciences (SPSS).

5.1 First Hypothesis: Effort

Through an initial analysis of the distribution, the behaviour of the samples could be evaluated. Initially, we studied the behaviour of each sample (traditional and SPIM) in order to find outliers. An outlier is an observation that lies an abnormal distance from other values in a random sample from a population (Grubbs, 1969). According to the boxplot graph it was observed that the effort variable does not have outliers. After that, it could be verified that the data has a normal distribution through Shapiro-Wilk test. However, the T test also assumes that the variability of each group is approximately equal. With this goal, two hypotheses were defined: H_0: The variances are equal; H_1: The variances are not equal. The Levene's test (see Table 1) shows if its assumption of the T test has been met.

Table 1: Levene's test for the effort variable.

Variable	Assumption	Significance
Effort	Equal Variances assumed	0.271
	Equal Variances not assumed	0.271

According to the results, the significance (p-value) of Levene's test is 0.271. If this value is lower than or equal to the significance level (α) for the test (in this case 0.05), then the null hypothesis in which the variability of the two groups is equal may be rejected, implying that the variances are unequal. If the p-value is greater than the α level, then, equal variances are assumed. In this case, 0.271 is greater than α, so the fact that the variances are equal was assumed. Once it was identified that the distribution was normal and variances were equal, the T test was applied (see results in Table 2).

This is a two-sided test, in which the p-value = 0.140 is directly compared with α = 0.05 (significance level). Since p-value=0.140 > 0.05, H_0 is not rejected. Thus, there is no statistical evidence to reject the hypothesis that the effort average to accomplish the planning of the activities using the traditional model is equal to the spent effort with the SPIM model. It was observed that the effort average, in minutes, to carry out the planning of the activities using the SPIM model was around 48 minutes while using the traditional model it was around 43 minutes. For the presented analyzes, the conclusion is that, statistically, there is no significant difference in relation to the effort to make the planning of projects using the traditional method and the SPIM.

Table 2: T test for the effort variable.

Variable	Criterion	T	Sig.
Effort	Equal Variances assumed	-1.511	0.140
	Equal Variances not assumed	-1.511	0.140

5.2 Second Hypothesis: Accuracy

Similarly to the analysis of the first hypothesis, it was observed that the accuracy variable does not have outliers. However, using the Shapiro-Wilk test, it was not possible to identify if that the data has a normal distribution. Therefore, a parametric test (like T test) could not be used. The Mann-Whitney, however, test could be used. It is a non-parametric analog to the independent samples T test and can be used when we do not assume that the dependent variable is a normally distributed interval variable. Thus, the Mann-Whitney test for two independent samples was used to verify that the observed differences between the averages in two independent groups are statistically significant. To achieve this objective, the following hypotheses were defined: H_0: There is no difference between the mean of the two samples; H_1: There is a difference between the mean of the two samples. The results of the Mann-Whitney can be seen in Table 3.

Table 3: Mann-Whitney test for the accuracy variable.

Variable	Mann-Whitney U	Wilcoxon W	Z	Asymp Sig
Accuracy	66.500	237.500	-3.046	0.002

Since the degree of significance (0.002) is smaller than the level of significance given (0.05), the hypothesis H_0 was rejected. Based on the results presented for the accuracy variable it is understood that there is a difference between the mean effort to do the planning with the traditional and SPIM methods. However, based on the results of the Mann-Whitney test only the null hypothesis can be rejected, but it was not possible to evaluate the alternative hypotheses. Comparing the mean values of the SPIM approach (84%) with the traditional approach (52%) we conclude that the accuracy in making the planning model using the SPIM is larger than in the traditional model.

6 QUALITATIVE ANALYSIS

A qualitative exploratory evaluation has also been conducted. At the end of the experiment's execution each participant answered a questionnaire, produced in accordance to Rea and Parker (2005). The survey had 17 questions where the first 8 were focused on the managers individual knowledge mapping and the remaining were used to estimate the SPIM model's contributions in the planning process from the project managers' point of view.

An analysis of the obtained results from the questions related to the profile of respondent individuals shows that 52.63% of these had a project management experience between two and five years and 21.05% had experience between five and ten years. In addition to that experience of the respondents, 34.21% of the sample reported their experience in project management as little while the remaining 65.79% declared it as moderate or advanced. In addition, 72.42% of the subjects classified their knowledge of software development processes as moderate or advanced. This indicates a sufficient range of experience regarding project management by the subjects.

Table 4: Perceived benefits in performing the integrated planning of managerial and productive activities.

Question	%
Reduction in time during the project's elaboration process	52.63
Identification of the dependencies between the management supporting activities and production activities	100
Identification and measuring of the indirect costs of the project, due to the management support activities	60
Being able to access enterprise workflow information	75
The capacity of avoiding distortions during planning when support activities are involved	100
Helps to anticipate the needs stemming from support areas of the organization during the project planning	87.50
Makes an explicit distinction between the activities of a software project and the activities belonging to other departments within the organization;	100

The analysis of the SPIM begins with the respondents' evaluation of the direct benefits in performing the integrated planning of managerial and productive activities in a project (see Table 4).

According to the second and the last rows in Table 4, all participants found that the integrated planning allows the identification of the hidden dependencies between the management supporting activities and the production activities, while avoiding frequent distortions in the planning of the projects due to the uninformed use of resources from the management supporting activities. The visibility of management supporting activities with the activities in the software project (whether productive of managerial) was also identified as a strong benefit of SPIM by all of the interviewees.

When questioned whether they agreed or not on the distinct nature of the three types of activities, most of the respondents (75%) answered that the

SPIM model helps managers to access enterprise workflow information. Also, 87.50% of the interviewed subjects agreed that the SPIM model contributes in identifying the dependencies of the activities between the project workflow and the organization workflow, which allows the prediction of the needs that come upon the organizational support areas during the planning of the project. As a final consideration, the majority of the participants found that the SPIM contributes in the identification and measuring of the indirect costs of the project, due to the management support activities.

7 CONCLUSIVE REMARKS

This paper presented the SPIM, a model to integrate software project management with organizational workflows. An experimental strategy was chosen to evaluate the proposed model. This experiment aimed to compare the accuracy and the effort of integrated planning model SPIM compared to the traditional model of software project planning, considering the characteristics and particularities involved in the Unified Process.

The experiment reveals that the use of the SPIM helps managers to create and conduct a more precise project plan than the traditional method. In certain circumstances, the project manager only perceives the need to have asked another department for some information earlier just at the very moment the team must execute a project's activity that depends on that other department. The obscurity in identifying this kind of relationship during the planning and execution of a software project can negatively affect the project schedule. This evidence was clear during this study while analysing accuracy variable.

An evidence related to the effort variable could also be extracted: the time for planning the activities using the SPIM is similar to the traditional model. The idea behind the SPIM comes from the need to reduce the complexity in visualizing the interdependencies of both organizational workflows and individual project's workflow of activities. Most of the effort of using the SPIM is related to filling the extra information proposed by this model. Nevertheless, the results of the effort variable did not become favorable to the traditional method. The results of this experiment reaffirm the benefits that the SPIM provides in solving problems related to the inadequate definition of tasks due to the obscurity in visualizing the interdependency between the organization's and project specific workflows.

REFERENCES

Callegari, D., Rosito, M., Blois, M., Bastos R. (2008). An Integrated Model for Managerial and Productive Activities in Software Development, In: *ICEIS - 10th International Conference on Enterprise Information Systems*, Spain.

Field, A. (2005). Discovering Statistics Using SPSS. 2nd edition. *Sage Publications*.

Graham I., Henderson-Sellers B. and Younessi, H. (1997). The OPEN Process Specification", Addison-Wesley.

Grubbs, F. E. (1969). Procedures for detecting outlying observations in samples. *Technometrics*, 11, 1-21.

Joslin, D., Poole, W. (2005). Agent-Based Simulation For Software Project Planning. In *Proceedings of the 37th Conference on Winter Simulation*. Orlando, Florida, pp 1059-1066.

Juristo, N., Moreno, A. (2003). *Basics of Software Engineering Experimentation*. Kluwer academic Publishers. 2nd edition.

Miles M.B., Huberman A.M. (1994). Qualitative Data Analysis, 2nd edition. London: Sage Publications.

Kerzner, H. (2000). *Applied project management: best practices on implementation*. NY: Wiley e Sons, 534p.

Kruchten, P. (2000). *The Rational Unified Process: An Introduction*, Addison-Wesley, 2nd edition, 2000.

Object Management Group (2011). *Software & Systems Process Engineering Metamodel Specification 2.0*, viewed September 25, 2011, <http:// http://www.omg.org/spec/SPEM/2.0//>.

Pfleeger, S. L., Atlee, *J. Software Engineering: Theory and Practice*. (2009). 4th edition, Prentice Hall.

Pressman, R. (2009). *Software engineering: a practitioner's approach*. 7th edition, McGraw-Hill.

Project Management Institute (2008). *PMBOK - A Guide to the Project Management Body of Knowledge*. Newtown Square, PA: PMI, 4th edition.

Rea, L. M., Parker, R. (2005). *Designing and Conducting Survey Research: A Comprehensive Guide*, Jossey-Bass. 33rd edition

Rosito, M., Callegari, D., Blois, M., Bastos R. (2012). Project Management and Software Development Processes: Integrating PMBOK and OPEN. In: *WASET - World Academy of Science, Engineering and Technology*, Malaysia.

Rosito, M., Bastos, R (2012). A model to integrate software project management with organizational workflows. In: *ISDA - 12th International Conference on Intelligent Systems Design and Applications*.

Schwalbe K. (2010). Information Technology Project Management, *Thomson Learning*, 6th edition

Solingen, R., Berghout, E. (1999). The Goal/Question/Metric Method. NY: McGraw-Hill.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., Wesslén, A. (2000). *Experimentation in software engineering: an introduction*. Boston: Kluwer Academic Publishers, 204 p.