Timo Poranen (ed.)

Software Project Management Summaries 2014

UNIVERSITY OF TAMPERE
SCHOOL OF INFORMATION SCIENCES
REPORTS IN INFORMATION SCIENCES 31

TAMPERE 2014

UNIVERSITY OF TAMPERE
SCHOOL OF INFORMATION SCIENCES
REPORTS IN INFORMATION SCIENCES 31
SEPTEMBER 2014

Timo Poranen (eds.)

Software Project Management Summaries 2014

SCHOOL OF INFORMATION SCIENCES
FIN-33014 UNIVERSITY OF TAMPERE

ISBN 978-951-44-9584-7

ISSN-L 1799-8158
ISSN 1799-8158

Preface

This report contains summaries of project management articles published in international scientific journals and conferences. The summaries were written as a compulsory task for the “TIETS19 Software Project Management, Theory and Practice – Theory” –course held spring 2014.

The summaries were written in English or in Finnish. The summaries are not in any specific order; only English language summaries are first. All summaries have three sections: Introduction, Results and Conclusions. In the end there is also a book review.

Timo Poranen

Tampere, September 2014

Reducing I.T. Project Management Failures: Early Empirical Results

G.J. Hidding and J.M. Nicholas, in Proceedings of the 47th Hawaii International Conference on System Science, pages 4305-4314, 2014

Background

The paper reports empirical research results in hopes of helping to reduce failures in I.T. projects. The paper starts with an overview of literature regarding I.T. failure rates in the past decades. There is a summary of two project management paradigms; Value-Driven Change Leadership (VDCL) and Project Management Body of Knowledge (PMBOK).

Report from 2010 by the Standish Group states that failure rates still remain high even though there has been lot of effort trying lower the rates; "*24% of I.T. projects were canceled [–] 44% c ompleted over budget, late and/or with fewer features and functions than originally specified [–] 32% were delivered on time, on budget and met requirements*" (p. 4305). The goal of the study is to identify which factors are connected to projects failure or success.

Results

The study was was conducted with a survey including 30 questions each corresponding to a certain theme or area from VDCL or PMBOK. One or more question represent a factor that might affect project success or failure. Respondents answered the statements on 7-point ordinal Likert scale. The number of statements was kept small as the increase of statements was feared to reduce the number of answers.

The participating organizations Chief Information Officers were asked to identify two I.T. intensive projects; one that was considered a success and one that was considered a failure. This was done to remove the possibility of sample bias and also to have the same number of successful and failed projects. The questionnaires were then limited to only project managers of these projects as collecting data had to be done in person and questioning all stakeholders would have taken significantly more time and effort. The 8 organizations that participated to the study were all based in Chicago and represented six different industries.

As the research is ongoing and the size of the dataset was small the exploration of data was done using non-parametric tests and thus "*may not identify results that more-sensitive analyses might uncover*". As from each organization a pair of projects was collected and in some cases the projects were managed by same managers the variability within-organization is assumed to be minimal and thus paired-sample statistical tests were applied, namely sign test and Wilcoxon signed rank test. For an orservation to "*fall inside or outside of a rejection area, which, under the null hypothesis, is determined by a binomial probability distribution with p=0.5*" (p. 4311.) The results are based on Likert-score differences of each factor gotten by subtracting the score of unsuccessful project from the score of the successful project of the same organization.

The study "*identified three factors that were associated with successful projects according to the both the sign test and the Wilcoxon test (p<0,01; one tailed):Communication/ expectations management, scope management, and establishing the architecture of the end-item by release 1*". With lower statistical significance (sign test p<0.10 and Wilcoxon p<0,05) three more factors were also associated with a successful project; "*[a]greement on projects purpose, the end-item's architecture is reflected in the project plan, and time/ schedule management*". Seven more factors were identified but had statistical significance only on sign test or Wilcoxon test. "*Three factors (p<0.05 on only the Wilcoxon test) are: Human change over repeated activities, focus on agree-to-agree and common ground, and executive sponsorship. [–] Two other factors (p<0.1 on only the Wilcoxon test) are: Cost/ budget management and giving team members stake in the project.[--] Finally, we found two factors with statistical significance on only the sign test: Keep the business case updated throughout the project (p<0.05) and develop person-to-person relations (p<0.1)*" (p. 4312.) From these factors six were from PMBOK and seven from VDCL.

Conclusions

The study suggests that some traditional PMBOK practices seem to be strongly associated with project success. Also replacing some practices with practices from all themes of VDCL might be considered. Lastly "*project managers should reflect the end-item architecture in the project plan and establish the architecture by the first release*" (p. 4312). It would seem that both traditional PMBOK and VDCL offer good practices, but as the study is based on relatively small number of answers, the answers being self-reported by project managers wider conclusions cannot be drawn from at this time.

Antti Eikonsalo

An empirical study on the implementation and evaluation of a goal-driven software development risk management model

S. Islam, H. mouratidis, and E.R. Weippi, Information and Software Technology, volume 56, pages 117–133, 2014

Background

Every software project has uncertainties and risks at every stage of its lifecycle. Many experts have already defined all kinds of risk factors, such as “top-ten” risks factors which is provided by Boehm and five related influential factors: “technological newness, application size, lack of expertise, application complexity and organizational environment” (p.118). Also these risk factors are ranked by Iacovou et al. as “very important, important and less important” level with standards from three dimensions: “communication, client’s internal management and vendor capabilities” (R.T. Nakatsu, C. Iacovou, A comparative study of important risk factors involved in offshore and domestic outsourcing of software development projects: a two-panel delphi study, Information and Management 46 (1), page 57–68, 2009). Furthermore, Nakatsu et al. “investigated and compared the risk factors between offshore and domestic outsourcing” which showed the risk are caused by failure of “managing end user expectation and considering all project cost”. All these risk will do negative impact to the project and as Ropponen’s research showed that “75% of the project managers did not follow any detailed risk management practice and did not have adequate knowledge about software risk management” (p.120). Therefore, risk management, which “describes an integrated engineering approach to reduce the chance of project failure” (p.119), is necessary in the whole project lifecycle.

Boehm risk-driven Spiral model was the first contribution of “putting risk management into a single framework” (p.120) which combined with “theory W” at the same time. Later on, Software Engineering Institute (SEI) introduced a framework which focus on “identification, analysis, communication and mitigation strategies” (p.120). Riskit methodology, brought by Kontio, “provides a complete conceptual framework for risk management using a goal/expectation approach from the stakeholders and risks which threaten the goals” (B. Freimut, S. Hartkopf, P. Kaiser, J. Kontio, W. Kobitzsch, An industrial case study of implementing software risk management, SIGSOFT Software Engineering Notes 26 (5), page 277–287, 2001). However, all these framework showed different level of limitations. For example, the risk-driven Spiral model “is difficult to attain in real on-going project situations” while the SEI framework lacks the standard of integration between risk management and organization processes and necessary techniques. Even Kontio’s Riskit is also “not clear from where goals can originate and the risk analysis is based on scenarios which are difficult to formulate” (p.120).

However, goal-driven software development risk management model, apply the context concept of understanding the risk management from both “internal and external” aspects and “follows the guideline within the management process” (p.120).

Results

The Goal-driven Software Development Risk Management Model (GSRM) is a “framework that supports assessment and management of risks from the early requirements engineering phase” (p.120). Framework of GSRM consists of four layers, which includes suitable tasks, methods and techniques for performing specific activities under all layer. The first layers is Goal layer, which describe what the necessary things to a successful project are and who is on duty for fulfilling the goals. Next, the Obstacle layer defines all main causes that “reduce the ability to achieve a single or multiple goals” (p.122), and the following assessment layer “quantifies” the risk events as a serials of risk factors. The last layer is Treatment layer focuses on the control actions to solve the risks and all the goals can be attained. There are also five GSRM activities related to the four layer framework. In order to evaluate GSRM, a case study has been introduced in this thesis which obey the “Data collection and analysis” model (p.126).

The case study is a project which consists of several separate modules to ministry operational activities at a company of Domain Technologies Ltd. After introducing the GSRM to this project, a kick-off workshop will be the first tutorial which provide an overview of the whole GSRM process. The first task will be recognizing the high risk factors in the project and some necessary context such as user training, product specification and quality standard will be defined in this stage. Second activities is identification and modeling of goal. In this stage, the Risk Management (RM) team will produce a brainstorming session for defining and refining goals and sub-goals. Next activities will do identify and model obstacles. In the project, two big obstacles has been found by this activity: one is the numerous changes by the users will lead to an unstable project and the other is complex training budget, and the detailed events will be outlined. The forth activity is assessment of risks by prioritizing them into several scales, such as “very important” and “less important”, and assess each of them with Bayesian Belief Network. The treat and monitor stage will give “countermeasures to control the high and medium prioritized risks” (p.117). For example, “selected users’ training” would reduce the training budget problem”. At last, a goal-risk model for the complete project in the estimated budget and another one for obtaining positive reputation will be illustrated clearly.

With the “studied project and obtained results based on combination of research methods” (p.130), a comprehensive understanding about the usefulness of the GSRM can be showed clearly. An early application of goal-driven will be beneficial on communication among all stakeholders in one project and these people can “demonstrate their expectations” easily in the process and all the project goals will be emphasized. Because of the integration of GSRM with requirements engineering, errors can be found at the start of one project and make different roles, like project manager and user participation can “effectively support activities within requirements engineering and risk management”. Different views of the goal and risks which provided by these different roles were “identified and combined” (p.131), and presented by “both textual and graphical” format artefacts, which make it systematic and reasonable.

Conclusions

The research results showed that “a goal-driven approach is suitable for risk management and risk management is well integrated into the early requirements engineering” (p.132), and can be a “practical and reasonable risk management method” in an industrial context. The goal oriented view made it easy to understand and communicate in groups, and “a short training session was adequate for this purpose” (p.130). At last, people have gained experience and insight lessons learned from the case study. These can be useful for integrating risk management practice into software projects.

Ao Li

Project risk: Theoretical concepts and stakeholders' perspectives

Budi Hartono, Sinta R. Sulistyo, Poetry P. Praftiwi and Danar Hasmoro, International Journal of Project Management, volume 32, issue 3, pages 400–411, 2014

Background

Project Risk Management (PRM) has an effective role which brings success to project based organizations. However, there is a contradiction between practitioners' perspective on risk management and normative theory of decision making in risky situations according to past studies. The paper gives four examples from past studies held in developed countries such as United States and Sweden. As a result of these past studies, theoretical risk concepts and perspective on risk management differ from each other. Although these studies provide initial research for differences between theory and practice of project risk management, there are some limitations need to be considered and investigated for this research. Based on these limitations, this paper focuses on three research opportunities as investigating specific branch of project management whereas past studies only considered general project management, doing research in developing countries (in this case Indonesia) because of Hofstede's (1984) cross-cultural studies rather than developed countries and focusing on not only executive managers but also contractors and clients. Therefore, the paper makes important contributions to academic and practical perspectives. From academic perspective, the paper presents how theoretical knowledge and practitioners' behavior of handling risks differ. From practical perspective, the paper introduces how practitioners perceive risks and behave in a risky situation. This study performs an empirical study by conducting cross sectional surveys to Indonesian contractors and clients as a research method.

Results

Results of this study can be analyzed in five main topics as profile of respondents, perspectives toward risk, risk and project performance, dealing with risks and determinants affecting risky decision making.

Profile of respondents

First of all, target respondents of this study are people who work in construction companies and in a project related risky decision making position. Respondents' job profiles are defined by a questionnaire conducted to contractors and clients in order to see if respondents meet the study's criteria. 38.4% (out of 250) of contractors and 69.7% (out of 142) clients have responded to questionnaire. As a result of the questionnaire, contractors work as a project practitioners in different construction companies. Besides contractors' experience and duration of working, the frequency of project related risky decision making is measured as very often 24%, often 32% and seldom 40%. Also, 48 respondents were from the first line management, 32 respondents were from the mid management and 16 respondents were from top management. For the clients, they came from various companies such as multi-national and state-owned companies, government institutions and ministries. 48 respondents were from first line management, 54 respondents from the mid management and 2 respondents from the top management.

Perspectives toward risk

According to rational theories, risks could be positive as opportunities and negative as threats to the project. Additionally, risk exposure level can be defined in two dimensions as probability of occurrence of the risk and its possible damage. By contrast with this theory, Indonesian stakeholders relate the risk with its possible negative sides. Also, they don't consider probability of occurrence of the risk unlike it is stated in normative decision theory. Instead, they pay excessive attention to consequences of a risk.

Furthermore, risk attitudes of respondents were also asked. Majority of respondents have moderate or situational type of risk attitude. Hence, risk attitude of majority of respondents are adaptable to the circumstances. Also, it is found that Indonesian respondents have different risk patterns than those in developed countries. In short, it can be said that there is a difference between rational theories and stakeholders' perspectives toward risky situations. Respondents consider risk attitudes beyond risky choice whereas rational theories suggest that risk attitude is reflected by decisions of uncertain choices.

Risk and project performance

In this part of the study, respondents' opinion on relationship between risk and performance was investigated. It was investigated in two different versions as respondents' view on risk-performance relationship for future projects and on-going projects. For the first version of the investigation, there were three different "risk-performance" relationships which were asked to respondents to choose one of them. Majority of contractors (66%) and clients (67%) chose positive answered with positive "risk-performance" relationship. The answers no correlation and lower profitability for riskier projects are were asserted less. This means that majority of stakeholders expect a higher profitability for riskier projects.

For the second version of the investigation, performance is renamed as on-going project performance and risk is renamed as risky decision making. Respondents are asked to choose one among three possible risky decision making in two different cases which are "ongoing project performance is above expected" and "ongoing project performance is below expected". For the first case, contractors (94%) and clients (82%) preferred to have more aggressive stance by making decisions. For the second case, negative risk-performance relationship can be deducted since according to respondents' answer, the lower project performance tends stakeholders to make riskier decisions.

Dealing with risk

When current studies and past studies are considered, respondents' answers regarding dealing with risks match rational theories. Respondents already know that information is very important factor for decision making. If there is not sufficient information, they can postpone decision making until they get extra information. So that, dealing with risks can be controllable with information gathering and planning. However, there are substantial number of respondents who still rely on intuitions or gut feelings or simple analysis.

Determinants affecting risky decision making

Past studies suggest to put factors which affect risky decision making in two categories; external and internal factors. External factors include ‘company culture’, ‘country culture’, ‘law/regulation’, ‘socio-economic condition’, ‘geography’, ‘macro-economic condition’ and ‘involvement of other institutions’. Internal factors include ‘age’, ‘education level’, ‘gender’, ‘designation’, ‘compensation or bonus’, ‘work experience’ and ‘personality type’. Respondents are asked to rank top three most important factors among each of the categories. For contractors, socio economic condition, company size and cultures are most important external factors whereas clients rank law/regulation, macro-economic condition and involvement of other institutions in their top three list. For internal factors, contractors sort their preferences as current designation, education level and age. Clients find current designation, work experience and education level most important for internal factors. These findings differ from those in developed countries. In developed countries, stakeholders give more importance to nation level factors rather than socio economic factors.

Conclusions

This study aims to investigate stakeholders’ perspectives on four main issues as perspectives toward risk, risk-performance relationship, and attitudes during dealing with risks and determinants which affect risky decision making. The study held in one of the developing countries, Indonesia. It is found out that Indonesian stakeholders, both clients and contractors, think similar in risk related concepts. However, there are clear distinctions between their perspectives and normative risk management theory as in developed countries. The inconsistency between theory and stakeholders’ perspectives can be explained with two practical implications. First one is insufficient application of theories on daily decision making and second one is that the gap prevents implementation of models and tools. The most important step that should be done would be narrowing the gap by taking systematic approaches. In this regard, project practitioners need to understand and try to apply theoretical concepts in decision making. While new models are being developed, the gap needs to be considered by scholars.

Dilem Aydinli

Exploring the value of project management: Linking Project Management Performance and Project Success

F. Asad Mir and A.H. Pinnington, International Journal of Project Management, volume 32, issue 2, pages 202-217, 2014

Background

This paper analyses the relationship between PM Performance and Project Success. In previous decades, PM processes and tools have advanced significantly however the Project Successes have not shown the same enhancement according to scientific researches. Of course there have been researches pointing out the relation between Project Success and PM Performance but there is not enough research done currently to fully interpret the significant relation these two concepts have between each other. The main reason behind this has been the difficulty in measuring the relationship between PM Performance and Project Success quantifiably and therefore this paper aims to analyze these two concept's relationship by quantifiable and concrete data and measurements.

Before moving to the data analysis and findings, authors give their frameworks and definitions on PM Performance and Project Success concepts. This is significant because the definition of a project's success varies considerably from culture to culture and project to project therefore measuring project success becomes considerably difficult. Hence amongst a lot of Project Success definitions and frameworks one is chosen. Shenhar's framework and Stefanovic's concept of teamwork effectiveness is combined in one whole where Shenhar argues that "Projects are strategic and project success should be assessed according to short-term and long-term project objectives." Then authors add "Their framework links project success with competitive advantage and includes: Efficiency (meeting schedule and budget goals); Impact on customers (customer benefits in performance of end products and meeting customer needs); Business success (project benefits in commercial value and market share); and Preparing for the future (creating new technological and operational infrastructure and market opportunities)" (Page 204). After giving definition to project success the authors move on to chose a definition for PM Performance again amongst a lot of frameworks and definitions. It is decided that PMPA model of Bryde would be more suitable for the analysis of these two concepts. Bryda's PMPA (Project Management Performance Assessment) is based on five enablers of high PM Performance; PM leadership, PM staff, PM policy and strategy, PM partnerships and resources and project life cycle management process. Additionally, there is another area in this framework; PM Key Performance Indicators (KPIs) which is based on actual measurements of achievement in project management activities.

Lastly and most importantly, 3 propositions and 7 hypotheses are made based on the claim there is a clear relationship between PM Performance and Project Success.

Proposition 1: There is a positive influence of PM Performance on Project Success.

Hypothesis 1: There is a positive statistical relationship between PM Performance and Project Success

Proposition 2: The variables of the PM Performance construct have a positive influence on Project Success construct.

Hypothesis 2...7: There is a statistically significant positive relationship between Project Success and PM Success factors all separately (PM Leadership, PM staff, PM policy and strategy, PM partnerships and resources and project life cycle management process, PM KPIs)

Proposition 3: The individual Project Performance variables have a positive influence on individual Project Success elements.

Results

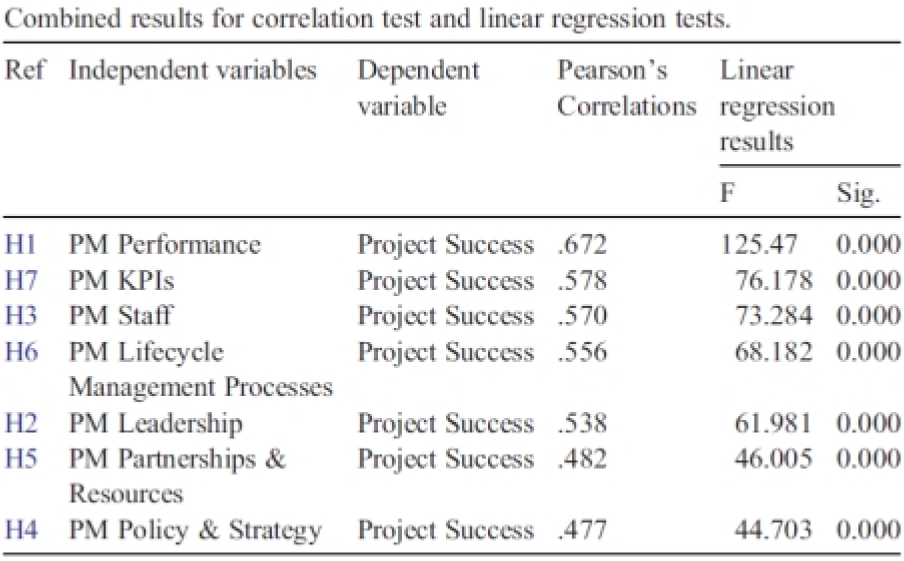
An online questionnaire was used as the research methodology to measure and assess the relationship between PM Performance and Project Success. The questionnaire was held in three sections where first section held questions about recipients and their organization, second section held questions towards PM practices in their organization which assessed PM Performance and the last section had questions regarding recently completed projects which helped assess project success in the organizations. A pilot questionnaire was made with five participants. This questionnaire held 48 questions in total and assessed the internal and external validity for this research. Lastly, the final version of the questionnaire was held amongst PM professionals working in UAE organizations. The questionnaire was asked to be forwarded online amongst the recipients but only 154 people responded amongst 1500 people however the results were enough and suitable for this research.

The data gathered were investigated according to the reliability factor using Cronbach's alpha method and some adjustments were made like removal of two questions from the data and the resulting data were reliable and perfectly usable for scientific research purposes. Pearson's correlation and linear regression methods were chosen for processing the data. Both of these methods assessed the relationship between the dependent variables and independent variables in the data.

After applying Pearson's correlation to the data it was found that Project Success was significantly correlated with PM Performance construct and each of its variables. Therefore every hypothesis was supported by the findings in this data analyses. PM Performance showed the strongest correlation between the independent and dependent constructs.

Then linear regression method was applied to the data. The findings were such that PM Performance explained %44.9 of the variance in Project Success, with a highly significant relationship. PM KPIs, PM Staff, and PM Lifecycle Management Processes explained at least %30 of the variances individually in Project Success. All the rest of the relationships explained less than %30 variance in the dependent variables.

It can be seen that both of the methods had similar results and are backing themselves up. The table below (Table 4 of the article) explains the results by cross validating correlation and linear regression results.



Combined results for correlation test and linear regression tests.

Ref	Independent variables	Dependent variable	Pearson's Correlations	Linear regression results	
				F	Sig.
H1	PM Performance	Project Success	.672	125.47	0.000
H7	PM KPIs	Project Success	.578	76.178	0.000
H3	PM Staff	Project Success	.570	73.284	0.000
H6	PM Lifecycle Management Processes	Project Success	.556	68.182	0.000
H2	PM Leadership	Project Success	.538	61.981	0.000
H5	PM Partnerships & Resources	Project Success	.482	46.005	0.000
H4	PM Policy & Strategy	Project Success	.477	44.703	0.000

Propositions 1,2 and all the hypothesis have been supported by the findings so far and in order to make judgment on proposition 3 a linear regression was made for project performance and each of its variables (independent variables) and were validated against each variable of Project Success (Dependent variables). Apart from this analysis, multiple regression method was used to further investigate the most variance in the dependent variable Project Success.

Lastly Principal Component factor analysis with varimax rotation was applied to PM Performance and Project Success to validate the structure of both concepts.

According to these results and findings, the influence of individual PM Performance variables on Project Success construct can be listed. PM KPIs were the most significant individual variable contributing to the success of any project followed by PM Staff being the second most important. PM leadership was also important after these two variables and has a significant effect on project success. PM lifecycle Management Processes have positive effect but it is not significant. Lastly PM Partnership & Resources and Policy & Strategy are at the end of the list when their relationship with Project Success is considered. If these two variables are taken out of the equation the variance in linear regression and the Pearson correlation values make much more sense towards relating PM Performance variables to Project Success.

Additionally the relationships of PM Performance construct and its variables with individual Project Success variables are analyzed accordingly with the findings. The results are important because this paper is the first research that analyzes the relationship between the PMPA framework elements and Project Success variables. According to linear regression results, impact on project team is the single most variance explained Project Success variable by the majority of Project Performance variables. PM KPIs have the most wide ranging impact across the different variables of Project Success since it has the highest correlation amongst other variables. Besides PM KPIs, PM Lifecycle Management Processes and PM Staff also contribute a reasonable amount of variance in the PM Success variables. Project efficiency and Business Success were ranked at the bottom and were the least impacted variables.

The main findings and recommendations of this study could be summarized as;

a) PM KPIs was the most important variable that affected the success of the projects therefore companies should invest in such area to develop PM methods that manage KPIs.
b) PM Staff was another very important variable that helps reaching success in projects and organizations must pay more attention on evaluating Staff performance and training Staff members.
c) Relations with customers and partners, and Policy and Strategy should be valued more in organizations and at all levels in the PM hiearachy. This will lead to success in projects and organization.
d) A successful PM Performance framework can be important in creating a positive impact on project teams so organizations should encourage themselves in investing PM Performance Frameworks.
e) Organization's future success is highly correlated with lifecycle management process and systems that are implemented in the organization. Therefore investment in processes and systems would bring future success to the organizations.

Conclusions

This research study showed through the use of Pearson's correlation method that PM Performance and Project success are positively correlated with each other. Furthermore linear regression analysis showed that PM Performance is the reason behind %44.9 variance in Project Success where %55.1 variance's source is unknown however it is not the main topic of this study. PM KPIs and PM Staff were the most important two variables of PM Performance on Project Success while PM Policy and Strategy was found to be the least important and influential variable. Another important finding was that PM Performance didn't have much effect on Project Efficiency.

Additionally, PMPA framework was claimed to be a suitable representation of PM Performance but high level factors such as Policy and Strategy, Partnerships and Resources and Leadership should be made visible and meaningful to employees.

As a result, this research points out the significant relationship between PM Performance and Project success. Organizations can make use of this relationship and increase their success in projects by the help of this research.

Batuhan Baykara

Key Drivers of Effectiveness in Managing a Group of Multiple Projects

P. Patanakul, IEEE Transactions on Engineering Management, volume 60, number 1, pages 4-17, 2013

Background

The project activity is increasingly applied in different fields and many organizations run a number of projects simultaneously which challenges project managers to deal with a group of multiple projects at a time. In that case project managers have to be able to act as multiple-project managers, and they might face the problem of effective management due to the lack of theoretical support and experience in that area. The article in question continues the previous studies of Patankul on management of a group of multiple projects (MGMP), summarizes practical experience data on MGMP effectiveness gathered from 169 multiple-project managers, and finally suggests the list of MGMP effectiveness key drivers. The topic of this study seems to reflect the current need for project managers to keep pace of the time and enlarge continuously their knowledge in project management. There are few publications on MGMP but they refer mainly to technical issues such as resource allocation or management control, whereas the human aspects such as overload, leadership or multiple team membership need to be investigated. The findings of the research will be utterly useful for those who want to be engaged in MGMP and improve the skills of effective and successful project management.

Results

The challenge to lead several projects simultaneously includes both planning management and switching between projects context. The more unique projects in a group the more difficult for project managers is to adjust their activities for a specific project due to difference in management and leadership style needed and objectives of the projects. On the other hand, leading such projects is likely to contribute to the knowledge transfer and overall increase of project management efficiency in case of good understanding of MGMP principles. Therefore, the researcher raises a question of how to help multiple-project managers to be effective.

The research method of the study is based on a questionnaire about MGMP effectiveness. The answers are rated on 1-7 Likert scales. The first part of questionnaire includes the question about the constructs which are represented by four independent variables – assignment, resources, interdependence management, MPM (multiple project management) competency - and the measurement of MGMP effectiveness which is described by two dependent variables – project performance in terms of time, cost and client satisfaction, and learning of multiple-project managers. The effect of constructs on MGMP effectiveness is studied by means of three contextual variables for each project manager: the number of projects, its durations, and the degrees of its technical uncertainty. The second part consists of demographic questions about respondents' and projects' background. In the third part a respondent is to describe his/her own effective approaches used in MGMP by his/her own words.

According to the figures in the article, the reliability of each construct is proven to have a good statistical result. Overall statistical investigation validates the results received in the survey. The results show that MPM competency, assignment and resources influence on project performance significantly, and MPM competency and interdependence management effects on learning.

Virtually, MPM competency is the main factor of project performance among analyzed variables. This finding emphasizes the importance of multitasking and multi-team management abilities for project managers. Multitasking competency enables better estimation of the resources capacity, priority setting, switching quickly between different projects contexts, and consequently more productive managing a group of projects. Multi-team management competency helps to rally and lead teams varying the leadership and management style depending on the individual project and the team's characteristics. The suggestion that both these skills influence directly on project performance is also evidenced in the third part of questionnaire where several project managers included MPM competency as an effective approach they used in MGMP.

The importance of effective project manager assignment is also proved to increase project performance. This means that person-job fit, project similarity, project manager's availability and career goals should be assessed thoroughly in the assignment process.

The next factor of project performance relates to resource allocation. It is quite common for multi-project environment to suffer from resource deficit which might lead to endangered projects and cause extras in project management. Consequently, sufficient resources have a direct relationship with project performance, especially in a group of multiple projects.

Last but not least, MPM competency and interdependence management occur to be significant factors of learning. This relationship between MPM competency and learning can be explained by assumption that frequent switching between projects contexts, leadership style changes and multitasking promote acquiring new knowledge and experience. Effective interdependence management requires more knowledge and experience from a project manager since simultaneous managing of two or more projects assumes working with shared resources and technologies, high coordination and collaboration. It does not only stimulate project managers to learn more effective approaches of MGMP but also enables transfer of knowledge between different project managers due to collaboration.

Conclusions

All things considered, the results of the study add knowledge to the theoretical and managerial implications of MGMP. The main findings of the research evidence that, first, multitasking and multi-team management competencies of multiple-project managers and sufficient resource allocation contribute significantly to MGMP efficiency, and second, management of the interdependence between multiple projects improves learning of project managers.

The contribution to the theory of MGMP seems to be more considerable as it shows the detailed model of estimating the impact of different factors on MGMP effectiveness that can be adjusted for further similar researches to investigate other aspects and relationships in project management. The managerial contribution is not so obvious because all offered measurements of MGMP effectiveness happened to influence significantly on both project performance and learning in one way or another. In other words, multiple-project managers should pay attention to all the factors investigated in the study regardless its specific effect on project performance or learning since the weights of project performance and learning for MGMP effectiveness are not clearly established.

Elena Betekhtina

Failure Factors of Software Projects at a Global Outsourcing Marketplace

M. Jørgensen, Journal of Systems and Software, volume 92, pages 157–169, 2014

Background

The paper presents a study conducted to understand the reason behind software project failures. The model was developed using 785,325 datasets of small software projects on vWorker.com. vWorker.com is an online global marketplace where clients outsource projects to developers from the entire world. The study outlines that previous collaboration between the provider and the client and less project failure by provider leads to high reduction in the risk of failure. Moreover, it was found that when clients give priority for low price and the project size increases the risk of project failure increases. The study presented in this paper differs from earlier studies on project failures it focuses on effects of potential project failures known at project start, uses observational data, and focuses on projects of global outsourcing marketplace. A binary logistics regression model of project failure was developed using 437,278 project data from 2001 to 2008. The model was tested on 346,047 dataset from 2009 to 2012 to assess the prediction accuracy of the model. Variables that have causal relationship to project failure were included. The reason is the model is not only to predict project failure but also explain why software fails in outsourcing market.

Results

The model for risk of project failure was developed based on a binary logistic regression where the output is a value between 0 and 1 and can be interpreted as the predicted probability of project failure. A cutoff value was calculated using ROC (Receiver operating characteristics) curve of the learning data set. The accuracy of the model was evaluated by applying the model with the learning data set and with cutoff value 0.2. It was found that the evaluation dataset have same results as the learning dataset. In addition to predicting project failure an understanding of the factors that contribute to the higher risk of project failure is important for reducing risk among other things.

Related work

Previous studies related to project failure confirm the results of the current study. The 11% cancellation is same with other studies and the skill of the provider and previous collaboration are crucial for explaining project failure. The role of the client and the size of the project contribute to the risk of project failure. The results of the study extend previous results and contradict to some extent. It was found that risk of project failure increases when client selected a provider with a lower than average bid price, a topic not much studied in software project contexts. However, it was found that selecting offshore projects does not increase risk of project failure.

Discussion and conclusion

Similar analysis was extended to include different project sizes to gain more generalization. The findings were the models are similar though might be limited to the global outsourcing market place. However, it's noted that it's difficult to use the results to predict project failure outside the global outsourcing market place. There was difficulty in defining project failure variable in an objectively measurable way. The causal relation between variables and risk of project failure were complex so difficult to understand reason for failure in cause-effect model. Moreover, developing logistic regression model is based on many assumptions and brings challenges when applied on software development scenario. It's concluded that the risk of project failures can be predicted by using factors known at project start-up. The prediction model works on small-scale projects on outsourcing marketplaces. However, it was found it is reasonably robust on different project sizes. The main suggestions of the results are good provider skill rather than law price reduce the risk of project failure, in selecting a skilled provider asses previous collaboration and previous project failure rate and, the skill of the client play key role in reducing risk of project failure.

Habteselassie Biruk Yemane

Performance on agile teams: Relating iteration objectives and critical decisions to project management success factors

M. L. Drury-Grogan, Information and Software Technology, volume 56, issue 5, pages 506-515, 2014

Background

Project management surveys show evidence that Information Technology projects frequently fail, even though golden triangle had established the project management success factors. Golden triangle, as a global standards for project management guideline established by Project Management Institute, comprised of *schedule* (*time*), *budget* and *quality* to determine their project management success. However only by these three parameters is hard to reflect success if projects were over time, over *budget* or under deliver.

In agile software development, regular and short period time of iteration development are used, which can lead to benefit in incorporates unpredictable events, cost-effective and user-driven software, and faster high-quality product deliver. Iteration objectives and critical decisions of project teams may reflect project management success factors, but little is known about this case. To identify the relationship of success factors in golden triangle and iteration objectives and critical decisions of agile software development teams, the authors decide to investigate in the research that what is iteration objectives and how do they relate to the project management success factors, and how do critical decisions in the agile teams relate to project management success factors.

The author of the paper analysed those above questions through three case studies in different agile teams by using interview protocol, collecting relevant data including team distribution, objectives for each iteration, objectives ranking by their priority. Then mapped this qualitative data back to the PM success factors, analysed it using standard coding techniques and classified the iteration objectives into *Schedule, Dividing work, Iteration amendments, Time satisfaction,* or *Quality* categories, according to PM success factors in the golden triangle. Therefore, the ranked order of importance of iteration objectives and the critical decision that affect PM success can be reviewed, by comparing the similarities and differences from data across cases studied.

Results

The finding from the interviews and analyses show that two PM success factors, namely *Schedule* and *Quality*, are both relate to agile teams' iteration objectives and critical decisions. Specifically, according to the data collected and analysed from the three agile cases, iteration objectives focus mainly on *Functionality, Schedule, Quality* and *Team Satisfaction*, but not on *budget.* Whereas critical decisions focus mainly on *Quality, Divining Work, Iteration Amendments,* and *Team satisfaction*, apart from budget and schedule of the golden triangle as well.

As data collected from cases show that Agile teams discuss four categories of iteration objectives, and each aspect includes its own sub-category within that iteration objective. *Functionality*, including *development, testing,* and *documentation of functionality* refers to completing tasks as planned in each iteration, which is the most listed objective by interviewees. *Quality* is considered as the most important objective as it is the largest category as it consist of *Product insurance*, *Bug fixing, client issue addressing, other members' code reviewing* and *client satisfaction insurance*, totally five sub-categories. *Schedule,* as a critical path for completion is strictly important in the iteration progress, is divided into two sub-categories as *planning the work* and *finishing the work on time*. The last iteration objective is *team satisfaction* which can ensure the team successfully plan and go through the next iteration.

In the research, the author found out that critical decisions from agile teams relating to the success factors can also sort in four categories. *Quality* impacts team members from making decisions in order to have project management success. *Dividing works* properly, to ensure team members can do their work, is also a decision that affects PM success. *Iteration amendments* should be accepted after sign-off, and avoiding iteration amendments lead to unnecessary work is the third decision category. *Team satisfaction,* similar in the iteration objectives, is reported by teams that trying to increase team satisfaction should be a decision to make, in order to make the whole team step into a robust iteration. Although the traditional golden triangle of PM success factors doesn't include team satisfaction, it would seem a viable objective and a proper decision for agile teams can definitely increase the success of their project management.

Conclusions

The author examined iteration objectives in agile project management iteration process, therefore concluded that four objectives are relate to the golden triangle of project management success factors, by real cases analysing. In addition, this research also suggested four critical decisions made by agile management team are related to success.

In this research, one of the three factors in the golden triangle are specifically discussed both as iteration objective and critical decision, namely Quality. Another category is also analysed as a factor both works as iteration objective and critical decision, namely Team satisfaction, above of those two overlapped factors are of important factors for the whole developing progress. Apart from that, Functionality as an iteration objective and Iteration amendments as a critical decision are similarly critical for project success.

Even though analysing only three cases is insufficient to make robust evidences, and the current research design used in the research is single point in time adds some limitation, this research has contributed to the software development and project management by providing important insight and mind-inspiring idea, which may lead to better understanding of project management success, especially for agile development.

Chenyu Wei

Communication Factors for Speed and Reuse in Large-Scale Agile Software Development

A. Martini, L. Pareto and J. Bosch, in Proceedings of the 17th Software Product Line Conference (SPLC ’13), pages 42-51, 2013

Johdanto

Ketterät ohjelmistokehityksen menetelmät ovat kasvattaneet suosiotaan suurten ohjelmistoyritysten keskuudessa. Myös uudelleenkäytettävyyttä painottavat strategiat, kuten Software Product Line Engineering (SPLE), auttavat organisaatioita varmistamaan pitkäaikaisen tuottavuuden ja tehokkuuden. Sisäiseen tiedonvälitykseen ja kommunikaatioon vaikuttavien tärkeimpien tekijöiden ymmärtäminen on tärkeää, jotta ketterien menetelmien käyttö projekteissa tuottaisi tulosta.

Tulokset

Martini, Pareto ja Bosch ovat tunnistaneet ja pisteyttäneet 23 erilaista kommunikaatioon vaikuttavaa osatekijää, jotka hidastavat ohjelmistoprojektien valmistumista ja vaikeuttavat komponenttien uudelleenkäytettävyyttä. Tutkimuksessa kerättiin kyselylomakkeella laadullista ja määrällistä tietoa kolmesta suuresta organisaatiosta, jotka käyttävät ketteriä menetelmiä ohjelmistoprojekteissaan. Kyselyyn vastaajat jaettiin tehtävänsä mukaan neljään osaan: projektipäälliköt, järjestelmäarkkitehdit, suunnittelijat ja testaajat.

Tutkimustuloksista nousi esille yhdeksän kysymystä, jotka vastaajat mielsivät aiheuttavan eniten ongelmia ohjelmistoprojekteissa. Kolmella havaitulla ongelman aiheuttajalla yhteistä oli se, etteivät kaikki projektiin kuuluvat jäsenet työskennelleet lähellä toisiaan. Ongelmia havaittiin jo järjestelmävaatimusten luonnissa, jos ohjelmiston suunnittelijat ja toteuttajat työskentelevät eri työpisteissä. Sivuyksiköissä ja –toimipisteissä työskentelevien projektiryhmäläisten välisessä yhteydenpidossa nähtiin puutteita. Myös erilaiset asenteet ja arvot voivat vaihdella toimipisteiden ja yksiköiden välillä ja se voi tuoda tullessaan kommunikaatioon vaikuttavia ongelmia. Samassa toimipisteessä, mutta eri tiimissä työskentelevien välillä nähtiin olevan kommunikaatiokatkoksia. Tämä tuottaa ongelmia varsinkin siinä tapauksessa, jos projektiryhmä työstää ohjelmistokomponenttia toisen tiimin järjestelmään.

Prosessit, joita projektiryhmä joutuu noudattamaan, eivät sovi kehitettävään tuotteeseen. Tähän ongelmaan tutkijat ehdottavat, että tuotekehitysprosessia muutettaisiin sopivaksi ketterän ohjelmistokehityksen ympäristöön sopivaksi. Paikallinen prosessi ja myös tiimin asenne pitäisi saada tuottamaan toistuvasti palautetta projektista ja sen kulusta.

Projektipäälliköt ja ohjelmistoarkkitehdit havaitsivat ongelmaksi sen, että uudelleenkäytettävät ohjelmistokomponentit ovat valittu jo projektin alussa väärin ja näin ollen myös resurssien jakaminen projektin sisällä on mennyt epätasaisesti. SPLE –

näkökulman mukaan myyntityöryhmän tehtävänä olisi tarkistaa komponenttien uudelleenkäytettävyys. Uudelleenkäytettävyyteen liittyviä ongelmana nähtiin myös se, että vanhoja käyttöönotettavia komponentteja ei ole aikaa parannella enää uudessa projektissa.

Kyselyyn vastaajat olivat sitä mieltä, että monesti selkeä ohjelmistodokumentointi puuttuu ja uuden työntekijän ja projektiryhmän välillä tapahtuva viestintä on puutteellista. Tämä taas osaltaan pitkittää uuteen järjestelmään tutustumista. Martini, Pareto ja Bosch ehdottavat, että kommunikaatio-osaaminen tulisi liittää osaksi organisaation koulutusstrategiaa.

Yhdessä kyselyyn osallistuneista yrityksistä tunnistettiin kommunikaatio-ongelmia tuottavaksi asiaksi projektiin ja ohjelmistoon liittyvät epäkohdat ja ohjelmointivirheet. Tutkijat tekivät päätelmän, että kysymys voi olla osatekijä vain joissain asiayhteyksissä ja organisaatioympäristöissä.

Kyselytutkimuksessa viimeisenä pyydettiin vastaajaa laittamaan tärkeysjärjestykseen alueet joihin kommunikaatioon liittyvät haasteet kuuluvat. Kategorioita oli viisi: arkkitehtuuri, teknologia, prosessit, organisaatio ja ihmiset. Tuloksena saatiin, että suurimpana kommunikaatioon liittyvien haasteiden aiheuttajana nähtiin ihmiset. Erilaiset ajattelutavat, asenne ja tietotaito vaikuttavat kommunikaation laatuun. Toiseksi tärkeimmäksi kehitysalueeksi nähtiin arkkitehtuuri, kun taas teknologian osuus kommunikaatio-ongelmiin arvioitiin vähäiseksi.

Yhteenveto

Tutkimuksen selvimpänä kommunikaatio-ongelmien osatekijänä ketterissä ohjelmistoprojekteissa havaittiin erilaiset rajat projektiryhmien ja – jäsenten välillä. Rajoina voivat olla niin maantieteellisesti erillään työskentelevät henkilöt tai samassa toimipisteessä, mutta eri osastolla, vaikuttavat projektityöntekijät.

Martini, Pareto ja Bosch painottavat kolmea tärkeintä kehitysaluetta yrityksen sisäisen kommunikaation parantamisessa. Ketterien tiimien tulisi panostaa strategiaan ja arkkitehtuuriin, jotka tukevat komponenttien uudelleenkäytettävyyttä ja tehokasta ohjelmistokehitystä. Akateemista eli koulutuksesta saatavaa kuin myös organisaation työntekijöiden kouluttamista kommunikaatiotaidoissa tulisi kasvattaa. Viimeisenä he tuovat esiin erilaisten arkkitehtuuristen ratkaisujen kehittämisen, jotta projekteissa voidaan keskittyä asiakasarvon tuottamiseen.

Elina Leino

A framework for identifying software project risks

M. Keil, P.E. Cule, K. Lyytinen and R.C. Schmidt, Communications of the ACM, volume 41, Issue 11, pages 76-83, 1998

Johdanto

Ohjelmistokehitysprojektin onnistumista pidetään hyvin usein itsestäänselvyytenä, eikä projektin suunnittelulle ja mahdollisten riskitekijöiden selvittämiselle anneta aina tarpeeksi painoarvoa. Projektipäälliköt olettavat usein, että projektin onnistuminen on taattua, kun varmistaa prosessin sujuvuuden yleisellä tasolla.

Projektiin voi kuitenkin liittyä monia riskitekijöitä. Projektin aikana ilmeneviä ongelmia ole helppo korjata, jos niitä ei ole tunnistettu ajoissa, eikä niihin olla etukäteen varauduttu. Seurauksia voivat olla esimerkiksi aikatauluongelmat ja kustannusten kasvu. Pahimmassa tapauksessa projekti muuttuu kannattamattomaksi tai jää kokonaan kesken. Keilin ja muiden mukaan pelkästään vuonna 1995 USA:ssa käytettiin miljardeja euroja kuluihin, jotka aiheutuivat pieleen menneistä projekteista. Suuri osa tällaisista kuluista voidaan välttää kunnollisen projektisuunnittelun avulla, arvioimalla mahdollisia riskejä jo ennen projektin alkua.

Ohjelmistokehitysprojektien riskeistä on aiemminkin tehty tutkimuksia ja julkaisuja, mutta Keilin ja muiden mukaan suurin osa niistä rajoittuu tarkastelemaan vain tiettyjä kehitysprosessin osa-alueita. Aiemmissa tutkimuksissa ei myöskään olla haastateltu tarpeeksi syvällisesti sellaisia henkilöitä, jotka työskentelevät kehitysprosessin parissa päivittäin. Tämän lisäksi monet tutkimusprojektit ovat keskittyneet tarkastelemaan ainoastaan perinteistä teollisuutta. Lukuisat epäonnistuneet projektit ohjelmistokehityksen alalla ovat Keilin ja muiden mukaan osoittaneet, että uudenlaiselle järjestelmälliselle tutkimukselle on tarvetta. Yhtenä tutkimuksen tärkeimmistä tavoitteista oli kehittää sellainen menetelmä, joka on tarpeeksi yleiskäyttöinen ja jolla pystytään tunnistamaan tehokkaasti yleisimmät riskit. Keil ja muut painottavat, että heidän tutkimuksensa on ensimmäinen laatuaan, sillä vastaavia tutkimuksia ei ole aiemmin suoritettu samassa laajuudessa tai tarpeeksi asianmukaisessa kontekstissa.

Tutkimuksessa kartoitettiin ohjelmistokehitysprojektien yleisimpiä riskitekijöitä ja arvioitiin niiden vakavuutta. Tämän lisäksi arvioitiin kuinka usein eri riskit esiintyvät ja mitä kriteerejä projektipäälliköt käyttävät arvioidessaan riskien vakavuutta. Tutkimuksesta saatuja tietoja käytettiin riskien arvioimiseen tarkoitetun menetelmän kehittämiseen. Keilin ja muiden mukaan kehitetty menetelmä helpottaa projekteihin liittyvien riskien tunnistamista sekä analysointia, ja tarjoaa strategioita erilaisten riskityyppien hallintaan.

Tulokset

Tutkimusmenetelmänä käytettiin haastatteluja eri ryhmien välillä. Haastatteluista saatua tietoa käytettiin riskityyppien asianmukaiseen luokitteluun. Tavoitteena oli luoda luokittelu, jonka pohjalta voidaan kehittää sopivia kaavoja ja strategioita riskienhallintaa varten.

Haastatteluun kutsuttiin projektipäälliköitä eri puolelta maailmaa, Suomesta, Hong-Kongista ja Yhdysvalloista. Haastateltavia pyydettiin arvioimaan erilaisia riskitekijöitä ja näiden riskitekijöiden vakavuutta ja merkitystä ohjelmistokehitysprojektin kannalta.

Haastattelujen perusteella identifioitiin kymmenkunta erilaista riskitekijää. Vaikka eri maiden ryhmät haastateltiin erillään toisistaan, esille tulleet riskit olivat hyvin samanlaisia. Keil ja muut korostavat, että tulos viittaa samojen ongelmien yleisyyteen. Kaikissa projekteissa esiintyy jokseenkin samat ongelmat.

Kolme riskityyppiä tunnistettiin selvästi vakavimmiksi. Näitä olivat ylimmän johdon puutteellinen sitoutuminen, käyttäjien sitoutumattomuus kehitysprosessiin, sekä vaatimusmäärittelyyn liittyvät väärinkäsitykset. Ylimmän johdon puutteellinen sitoutuminen oli yksiselitteisesti vakavimpana pidetty riski. Se vaikutti sekä prosessin sujuvuuteen, että ohjelmistokehitysprojektin eri osapuolten motivaatioon. Käyttäjien sitoutumattomuudella tarkoitetaan sitä, että käyttäjät eivät olleet tarpeeksi aktiivisesti mukana vaatimusmäärittelyprosessissa. Tämä puolestaan johti puutteellisiin tai harhaanjohtaviin vaatimusmäärittelyihin, jotka eivät onnistuneet kuvaamaan todellista tarvetta.

Tutkimuksen yksi mielenkiintoisimmista havainnoista oli se, että vakavimpina pidetyt riskit olivat niitä, joihin projektipäälliköt eivät kokeneet pystyvänsä helposti vaikuttamaan. Tämä riskeihin vaikuttamisen näkökulma auttoi arvioimaan riskien vakavuutta ja kehittämään tehokkaan menetelmän eri riskityyppien kartoitukseen.

Johtopäätökset

Tässä tutkimuksessa tunnistettuja riskitekijöitä ja niiden ennaltaehkäisyyn ehdotettuja menetelmiä voi käyttää suuntaa antavana mallina ohjelmistokehitysprojekteissa. Tutkimus osoitti, että monet riskit voivat olla täysin projektipäällikön vaikutusalueen ulkopuolella ja niitä on hyvin vaikea valvoa, erityisesti jos niiden mahdollisuudesta ei tiedetä etukäteen.

Tutkimuksessa esitelty riskienhallintamalli tarjoaa hyvät perusteet projektinhallinnalle. Mallin avulla pystyy hyvin varautumaan tyypillisimpiin ohjelmistoprojektin riskeihin, mikä Keilin ja muiden mukaan lisää projektin onnistumisen todennäköisyyttä.

Evgeni Pajunen

Resource Constrained Multi-Project Scheduling with Priority Rules & Analytic Hierarchy Process

Amol Singh, Procedia Engineering, volume 69, 725 – 734, 2014

Background

Project management can be distributed into 9 parts: time, cost, scope, quality, risk, procurement, human resource, communication, and integration. According to those parts, project management can be considered as a complex decision making process. Most problems of project management occur in the process of planning and scheduling decisions. The manager need to control the allocation of the limited resource, meet the demand of its stakeholders and finished the project on time. Sometimes, in order to meet the peak demands, the managers have to delay doing other activities. and that action may extend the duration of the project. So scheduling and allocating resource properly are crucial during the process of the project management.

There were many traditional optimization methods about solving the scheduling problems, but all of them have a common feature that these studies were finished in the small size projects. However, as the number of the projects and the number of activities in projects increase. the scheduling problem is too difficult and complex to solve by the traditional method. Therefore, researchers are finding more effective methods to solve the scheduling problem, and they have developed heuristic and Meta heuristic methods for multi project scheduling. Now, some researchers are trying to improve the efficiency of the heuristic methods. However, the efficiency of heuristic methods also decrease with the number and size of project increase.

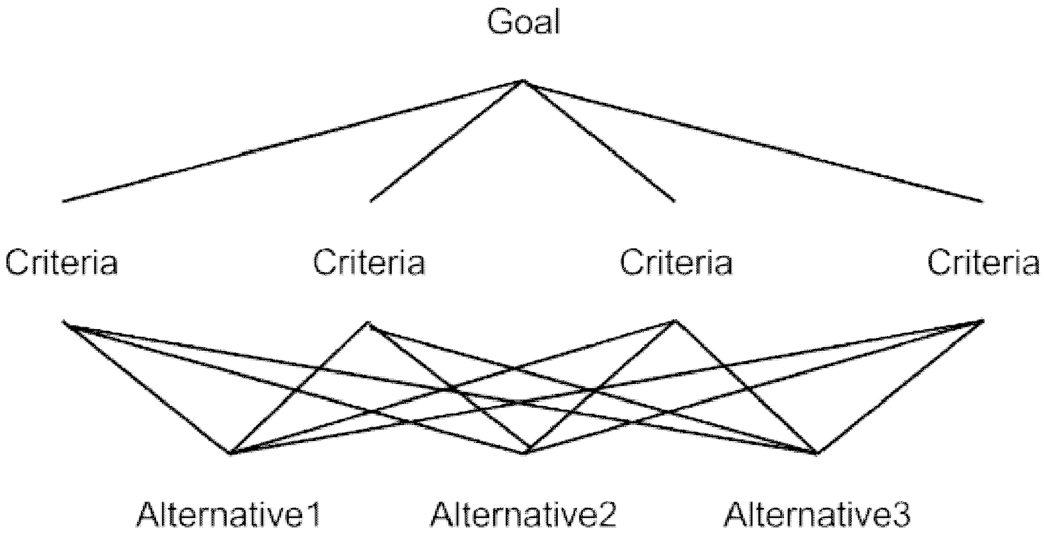
The greatest challenge of project managements is to finish the project within the deadline and the given budget. Many researchers have succeeded in developing the best algorithms in different environments. However there exist some problems in their research results. First, most current scheduling methods are only available in single project. Second, scheduling and allocating resource will become more difficult and complicated in multiple projects, current algorithms and methods can not deal with multiple projects. Third, some researchers have considered about problem of multiple projects, but they neglected constraint from a variety of resources., because in real life, complete an activity needs more than one resource.

Results

For addressing these research issues, three major tasks are outlined. Firstly, a complex multi project scheduling problem with resource constrained environment is considered. The performance of multi-project with resource constrained schedule is enhanced in terms of customer requirement by integrating the project criticality index with the activity priority index. Secondly, an efficient hybrid algorithm has been developed for generating the schedule of multi-project resource constrained problem. This algorithm integrates the project criticality index with the activity priority. Thirdly, in order to address the real requirement of the projects, a variety of resources are considered for each activity during the schedule development.(p.728)

The writer using the analytic hierarchy process(AHP) method to calculate the priority index of the project.

The first step in the analytic hierarchy process is to model the problem as a hierarchy. In doing this, participants explore the aspects of the problem at levels from general to detailed, then express it in the multileveled way that the AHP requires. As they work to build the hierarchy, they increase their understanding of the problem, of its context, and of each other's thoughts and feelings about both.

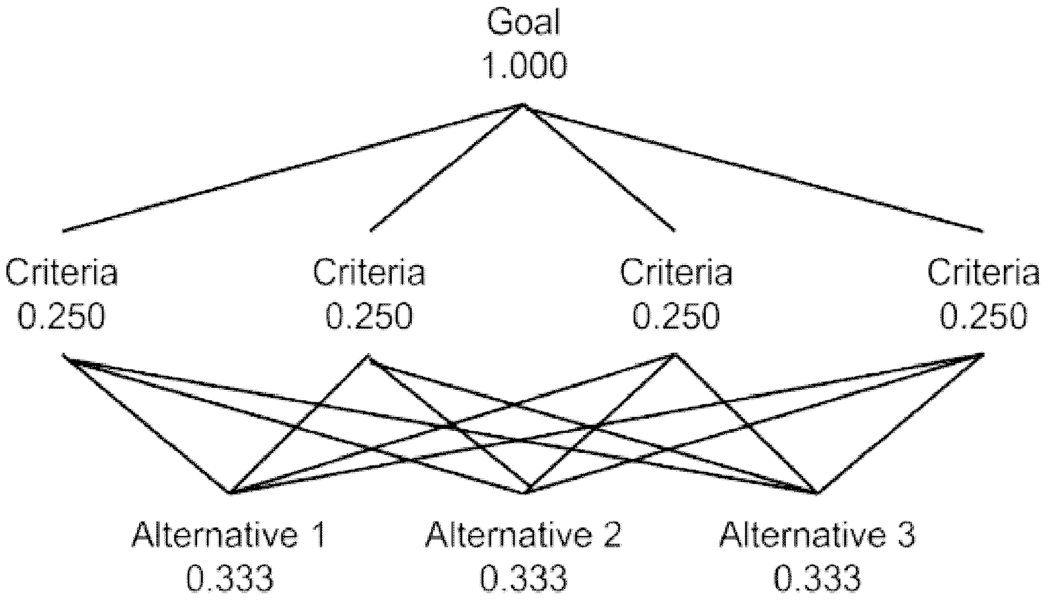


Once the hierarchy has been constructed, the participants analyze it through a series of pairwise comparisons that derive numerical scales of measurement for the nodes. The criteria are pairwise compared against the goal for importance. The alternatives are pairwise compared against each of the criteria for preference. The comparisons are processed mathematically, and priorities are derived for each node.

Finally, we should establish priorities. Priorities are numbers associated with the nodes of an AHP hierarchy. They represent the relative weights of the nodes in any group. Like probabilities, priorities are absolute numbers between zero and one, without units or dimensions. A node with priority .200 has twice the weight in reaching the goal as one with priority .100, ten times the weight of one with priority .020, and so forth. Depending on the problem at hand, "weight" can refer to importance, or preference, or likelihood, or whatever factor is being considered by the decision makers.

Priorities are distributed over a hierarchy according to its architecture, and their values depend on the information entered by users of the process. Priorities of the Goal, the Criteria, and the Alternatives are intimately related, but need to be considered separately.

By definition, the priority of the Goal is 1.000. The priorities of the alternatives always add up to 1.000. Things can become complicated with multiple levels of Criteria, but if there is only one level, their priorities also add to 1.000. All this is illustrated by the priorities in the example below.



Conclusions

In summary, the writer try to use Ahybrid algorithm to get the priorities of each project and activity, then set up the schedule according to the priorities of projects and activities. The Ahybrid algorithm is mature and efficiency, but it also need managers to decide the intensity of importance, so the managers should have enough experience and specialized knowledge.

Fan Zhaodong

The Impact of Agile Methods on Software Project Management

M. Coram and S. Bohner, in Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems, pages 363-370, 2005

Background

As opposed to other engineering disciplines, software engineering deals with products which change and evolve at a very high rate. Frequent changes often mean increased production costs as well, especially if changes occur in later phases of the development process. This is, however, true to a lesser degree in tangible products and this discovery led to the apparition of agile development methods which manage changes and related risks better than traditional methods.

Numerous agile methods have been introduced in the last decades which, despite small differences in specifics, share a common goal: enabling teams to respond more rapidly to changes. Where agile methods can prove to be highly effective in certain contexts, in some cases the overhead they generate might overpower the benefits they bring. Therefore it is important to consider the risks of each method and to determine whether an agile method is appropriate for a given project; which is often a real challenge. The aim of this paper is to help project managers in this challenge by examining the impact of agile methods on software project management and by highlighting some of the strengths and weaknesses.

"To provide a flavor" of the agile principles, some of the most important agile methods are presented. These methods are Extreme Programming (XP), SCRUM and Dynamic System Development (DSDM). XP concentrates on "getting the project at hand done" by applying a series of principles which proved to be successful. The development lifecycle consists of five phases: Exploration (obtaining requirements from customers; few weeks – few months), Planning (prioritizing, scheduling; few days), Iterations to Release (producing first release; one – four weeks), Productionizing (performance testing, delivering the release), Maintenance (implementing changes) and Death (completing documentation, disposition of the system). In SCRUM the focus is on organizing teams in such a way that they are able to produce software in a constantly changing environment. The SCRUM lifecycle consists of three phases: Pre-game (Planning and High-level design), Development (iterative development cycles, "Sprints"; one week – one month/sprint, three – eight sprints/project) and Post-game (concluding the effort and delivering the release). A distinguishing feature of DSDM is that it fixes time and resources first and then adjusts the rest accordingly. A DSDM process consists of five phases: Feasibility Study (project assessment), Business Study (business and technology assessment, architecture and prototype planning), Functional Model Iteration (functional iterations with enhancements), Design and Build Iteration (production of minimal system) and Implementation (delivering the system).

The primary intent of agile methods can be well determined from the "Agile Manifesto".

The following lines express clearly the core values of the agile principles:

- *"individuals and interactions over processes and tools"*
- *"working code over comprehensive documentation"*
- *"customer collaboration over contract negotiation"*
- *"responding to change over following a plan"* (p. 3)

Certainly, theses values do not suggest that the second items are of no significance, but they are nevertheless less important than the first items. For example, rigid processes and tools do not accommodate changes well, whereas putting more focus on people allows for more creativity in solutions. Also, while documentations are valuable, writing and maintaining them is highly time-consuming; a working product, on the other hand, is much more valuable. Customer collaboration is also highly beneficial and promoted by all agile methods: as opposed to contracts, it allows the customer to change their mind. Finally, responding to changes is often more important than sticking to a written plan, because in most cases changes occur more frequently than the modification of a plan would be possible. Supporting these four values leads to the following common features in agile methods: collaboration (both inside and outside the development group), code reviews, small teams (XP: three – sixteen developers; DSDM: six teams of two – six members), small release schedules (two weeks – six months), time boxing and constant testing.

Results

In the article the impact of agile methods on several parts of the project (such as, for example, the people, processes and other project elements) was examined. For this research, the authors relied mainly on related literature and specific background knowledge.

When examining the impact on the stakeholders of a project, the following groups where inspected: developers, testers, project leaders, customers and executive management. Developers might be the ones on whom agile methods have the largest impact, as the success of a project depends greatly on having strong developers; however, skilled workers are quite rare. This is a risk to consider, especially in the case of long-term projects. The impact on testers varies among agile methods. In their case the project management challenge usually lies in finding testers with appropriate skills and reallocating those who do not fit the group. Project leads can be divided into two groups: team leads and project managers. Project managers are often deeply involved in the development process and have to collaborate intensively with the team and the customer; if the project manager cannot take such a role for some reason, agile methods should not be selected. Customers are highly important in agile methods and frequent involvement is required from their side; however, some customers might not want to be that involved or for some projects it might be hard to even identify the targeted customers. The executive management also has an important in projects: their support is essential for success. However, it is often a real challenge to convince them about the benefits of using agile methods. All these stakeholders have to work in team to ensure success. High turnover rate can pose a serious risk in agile projects; in the case of frequently changing personnel agile methods might not be appropriate.

As for the impact on processes, it was found that agile methods have a great impact on the organization's processes: old, traditional processes often need to be replaced by agile ones. The planning process in agile projects, for example, is less emphasized (or is not considered important at all) and is relatively informal. Also the documentation is quite sparse, consisting often only of source code and user stories. While the reduced amount of documentation might enhance productivity, it might generate some risks as well, for example, in the case of new employees. Agile methods have a great impact on the development process as well, as they employ principles which might lead to dramatic changes. The key development processes are refactoring, minimalist development, code reviews and continuous integration.

The applicability of agile methods for different types of projects and the influence of business factors and other project characteristics was also subject of research. It was found that agile methods are most applicable to projects where the requirements are poorly defined and change frequently, whereas projects which require rigorous analysis (e.g. safety- and life-critical systems) are not supported. Among business factors, contractual obligation is a key factor affecting the appropriateness of agile methods. If requirements are set in a legal contract, use of agile methods is probably inappropriate. Also, projects with fixed release dates or with road-map for features may not be fit for agile methods. If detailed documentation is required, that also might have an impact on the applicability of agile methods. As for other project characteristics, time span might have a significant impact on the effectiveness of agile methods. Long-running projects might benefit greatly from agile methods. However, there are certain risks which should be considered in the case of long-term projects: loss of team members might be a serious problem and also the larger size of the product and the required maintenance might cause some issues. Project road map might be another characteristic to consider. Projects with well-defined road maps might not benefit from agile approaches, as they work well with architectures that consider more than just the current release.

Conclusions

Agile methods can provide solutions to manage effectively frequent changes and uncertainty in software development; however, they might not be appropriate for all projects. It is the project manager's responsibility to decide whether agile methods should be applied or not and in this decision several aspects of the project (people, processes, project characteristics) have to be taken into consideration. All in all, agile methods might offer a good alternative for managing ill-defined or rapidly changing requirements and even in projects where their applicability is questionable some of the principles might provide useful guidelines. With a competent team capable of adopting agile principles and implementing the processes, application of agile methods should be considered; otherwise traditional methods might be more appropriate.

Hang Do Minh

Knowledge sharing in information systems development projects: Explicating the role of dependence and trust

J.-G. Park and J. Lee , International Journal of Project Management, volume 32, number 1, pages 153 – 165, 2014

Johdanto

Tietojärjestelmiä kehitettäessä tietojen ja taitojen jakaminen kehittämisprojektin eri osapuolten välillä vaikuttaa sekä toteutettavan järjestelmän, että kehitysprojektin onnistumiseen. Tietojen ja taitojen jakamisen on todettu muunmuassa tehostavan työtehokkuutta sekä parantavan luovuutta ja innovatiivisuutta.

Onnistuakseen tietojen ja taitojen jakaminen edellyttää projektin eri osapuolten välistä riippuvuutta toisistaan. Tietynlaisen riippuvuuden lisäksi tarvitaan myös luottamusta osapuolten välillä. Ilman riippuvuutta ja luottamusta osapuolilla ei ole halukkuutta osallistua tietojen ja taitojen jakamiseen.

Park ja Lee keskittyvät tutkimuksessaan käsittelemään riippuvuutta ja luottamusta, niiden vaikutusta tietojen ja taitojen jakamiseen, sekä myös niihin vaikuttavia tekijöitä. Tutkimuksessaan Park ja Lee käsittelevät olemassaolevaa kirjallisuutta, joista he koostavat mallin, joka esittää tietojen ja taitojen jakamiseen vaikuttavat tekijät. Lisäksi Park ja Lee esittävät kirjallisuuden pohjalta 11 hypoteesia, jotka kuvaavat mallin eri kokonaisuuksien välisiä suhteita.

Esittämäänsä mallia ja sen oikeellisuutta testatakseen Park ja Lee analysoivat tilastollisesti suorittamaansa kyselyyn saamiaan vastauksia. Analyysin pohjalta Park ja Lee käsittelevät hypoteesiensa ja niiden kautta myös esittämänsä mallin paikkansapitävyyttä.

Tulokset

Park ja Lee pyrkivät tutkimuksessaan selvittämään, miten hyvin kirjallisuudessa yleisesti hyväksytty ajatus siitä, että luottamus ja riippuvuus ovat edellytyksiä projektia hyödyntävälle tietojen ja taitojen jakamiselle, pitää paikkansa. Lisäksi Park ja Lee tarkastelevat tarkemmin luottamusta ja riippuvuutta projekteissa, sekä näiden ominaisuuksien omia edellytyksiä.

Selvittääkseen mitkä tekijät todella vaikuttavat luottamuksen ja riippuvuuden syntyyn tietojärjestelmän kehittämisprojektissa, Park ja Lee muodostavat kirjallisuuden pohjalta mallin, joka kuvaa näitä eri ominaisuuksia ja niiden keskinäisiä suhteita. Parkin ja Leen esittämä malli koostuu kolmesta eri tasosta.

Alimman tason ns. ennakkovaatimukset muodostavat kolme perusominaisuutta, jotka ovat ympäristö, liikekumppani, ja kanssakäyminen. Tietojen ja taitojen jakamisen kannalta projektin kompleksisuus vastaa ympäristöä, asiantuntemus ja yhteiset arvot kuvaavat liikekumppania ja varsinainen tietojen ja taitojen mahdollistava kommuniointi vastaa kanssakäymistä.

Keskimmäinen tason muodostavat luottamus ja riippuvuus. Alemman tason ominaisuudet mahdollistavat luottamuksen ja riippuvuuden synnyn projektin osapuolten välillä.

Ylimmän tason muodostavat lopputulokset, eli tietojen ja taitojen jakaminen ja siitä seuraava projektiryhmän tehokkaampi toiminta.

Mallin eri tasojen välisiä suhteita kuvaavat Parkin ja Leen kirjallisuuden pohjalta esittämät 11 eri hypoteesia. Parkin ja Leen esittämät hypoteesit ovat:

(1) Tietojen ja taitojen jakamisella on yhteydessä projektiryhmän suorituskykyyn.

(2) Riippuvuus yhteistyökumppaniin on yhteydessä tietojen ja taitojen jakamiseen.

(3) Luottamus yhteistyökumppaniin on yhteydessä tietojen ja taitojen jakamiseen.

(4) Riippuvuus yhteistyökumppaniin on yhteydessä yhteistyökumppanin luottamiseen.

(5) Projektin kompleksisuus on yhteydessä riippuvuuteen yhteistyökumppanista.

(6) Yhteistyökumppanin asiantuntemus on yhteydessä riippuvuuteen yhteistyökumppanista.

(7) Yhteistyökumppanin asiantuntemus on yhteydessä yhteistyökumppanin luottamiseen.

(8) Yhteiset arvot on yhteydessä riippuvuuteen yhteistyökumppanista.

(9) Yhteiset arvot on yhteydessä yhteistyökumppanin luottamiseen.

(10) Kommunikoinnin frekvenssi on yhteydessä riippuvuuteen yhteistyökumppanista.

(11) Kommunikoinnin frekvenssi on yhteydessä yhteistyökumppanin luottamiseen.

Kaikkien hypoteesien kohdalla pätee, että yhteys on olemassa nimenomaan positiiviseen suuntaan, eikä välttämättä negatiiviseen, haitalliseen suuntaan. Parkin ja Leen esittämät hypoteesit nousevat olemassaolevasta kirjallisuuden totuuksina pidetyistä ajatuksista.

Testatakseen kirjallisuuden pohjalta esittämänsä mallin paikkaansapitävyyttä Park ja Lee keräsivät empiiristä dataa tutkimuskyselyn avulla. Tutkimuskyselyyn osallistui 139 paria, joista jokainen pari muodostui projektijohtajasta sekä tietojärjestelmä konsultista. Tutkimuskyselyyn osallistuneet olivat kahden suuren korealaisen IT-alan yrityksen työntekijöitä. Tutkimuskysely koostui useista kysymyksistä, joiden avulla mitattiin vastanneiden sen aikaisen käynnissä olevan projektin ominaisuuksia, eli Parkin ja Leen esittämän mallin kahdeksaa eri kokonaisuutta ja niiden mahdollisia keskinäisiä suhteita.

Keräämänsä datan analysoimiseksi Park ja Lee käyttävät PLS metodia. Tilastollisen analyysin perusteella Park ja Lee toteavat datan tukevan kaikkia yhtätoista kirjallisuuden pohjalta esittämäänsä hypoteesia.

Yhteenveto

Park ja Lee huomauttavat, että useimmiten tietojen ja taitojen jakamista käsittelevässä tutkimuksessa keskitytään vain pieneen osaan kokonaisuutta, minkä takia tämän tyyppinen ilmiön kokonaiskuvaa maalaavaa tutkimustulosta voidaan pitää merkittävänä ja tiedettä edistävänä.

Tutkimustuloksen perusteella on selkeää, että projektiryhmän on hyödyllistä yrittää kaikin puolin toimia siten, että luottamus- ja riippuvuussuhde yhteistyökumppanin kanssa on voimakas, sillä niiden avulla myös tietojen ja taitojen jakaminen projektissa mahdollistuu. Yhteistyökumppanin asiantuntevuus ja sen hyödyntäminen, yhteiset arvot projektin suhteen sekä kommunikoinnin frekvenssi ja monimuotoisuus korostuvat, sillä ne vaikuttavat sekä luottamus- että riippuvuussuhteeseen osapuolten välillä.

Janne Heikkilä

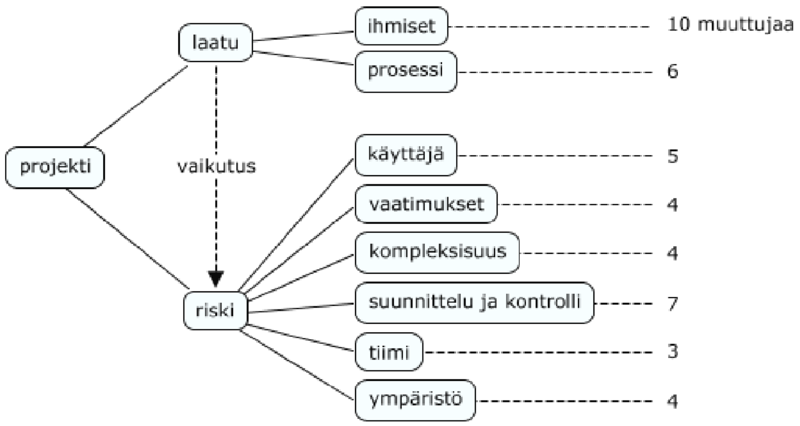
Quality vs risk: An investigation of their relationship in software development projects

L. Sarigiannidis and P. D. Chatzoglou, International Journal of Project Management, in press, 2014

Tausta

Artikkelissa käsitellään ohjelmistoprojektien laadun ja riskin suhdetta sekä niiden vaikutusta projektien onnistumiseen. Tavoitteena on luoda kyseisestä aiheesta uusi tutkimuksellinen malli perusteellisesti läpikäydyn kansainvälisen tutkimuskirjallisuuden pohjalta. Mallin testaamiseen tarvittava data kerättiin kreikkalaisista ohjelmistokehitykseen keskittyneistä yrityksistä.

Mallissa (kuvio 1) on kuusi riskidimensiota: käyttäjä, vaatimukset, projektin kompleksisuus, suunnittelu ja kontrolli, tiimi sekä organisaation ympäristö. Vastaavasti projektin laatu jaettiin kahteen osaa: ihmisistä ja prosessista johtuviin laatutekijöihin. Edellisten lisäksi olennaista on testata laadun vaikutusta riskeihin. Tutkijoiden mukaan päätutkimuskysymys on: kuinka paljon projektin laatuun liittyvät tekijät vaikuttavat projektin riskeihin?



Kuvio 1. Tutkimusmallin yksinkertaistus

Empiirinen aineisto analysoitiin kvantitatiivisesti käyttäen rakenneyhtälömallimenetelmää (structural equation modelling technique, SEM), jolla tutkitaan miten teoreettinen malli sopii aineistoon. Rakenneyhtälömallissa yhdistyvät sekä faktori- että regressioanalyysi, jolloin faktoreiden välistä yhteyttä tutkittiin regressioanalyysin avulla.

Tulokset

Tutkimusaineiston perusjoukko koostui kreikkalaisista ohjelmistoyrityksistä, jotka olivat mukana jossain ohjelmistokehitysprojektissa. Tällaisia yrityksiä oli tutkimusajankohtana 220. Perusjoukon yrityksistä 72 vastasi myöntyvästi, ja niistä palautettiin 124 kyselylomaketta. Lomakkeista 12 hylättiin (= 9 yritystä) riittämättömien vastausten vuoksi. Lopulliseen näytteeseen kuului 112 hanketta 63 yrityksessä (eli vastausprosentti oli lähes 29).

Yritykset määrittelivät käytetyn ohjelmistoprojektin ja siinä mukana ollut henkilöstö vastasi kysymyksiin. Vastaajista 42 prosenttia oli projektipäälliköitä, 40 prosenttia kehitystyötä tekeviä (pääasiassa ohjelmoijia) ja loput 18 prosenttia oli muita projektiin liittyviä henkilöitä (laatupäälliköt, suunnittelijat, auditoijat jne.).

Ennalta luodun mallin muuttujiin haettiin arvoja kyselylomakkeella. Myös kysymykset pohjautuivat kirjallisuuteen ja alan tutkijoiden keskusteluihin, ja niitä oli testattu aiemmissa tutkimuksissa. Osallistujia pyydettiin arvioimaan 27 ehdotettua[1] riskiä sen mukaan, mikä on niiden esiintymismahdollisuus ja vaikutus (kustannuksiin, aikatauluun, tekniseen suorituskykyyn ja projektiryhmän yhteistyöhön). Skaala oli välillä 1–10, jossa tapahtuman mahdollisuudessa 1 tarkoitti harvinaista ja 10 ”varmaa” mahdollisuutta. Vaikutusten osalta 1 kuvasi minimaalista tai olematonta vaikutusta ja 10 suurta vaikutusta.

Kysymykset testattiin etukäteen validiteetin vuoksi. Näin pyrittiin välttämään epätarkoituksenmukaisia, osittaisia, epämääräisiä ja monimerkityksellisiä kysymyksiä, jotka olisivat saattaneet vaikuttaa vastaajien mielenkiintoon ja turhautumiseen kysymyksiin vastatessa. Lisäksi reliabiliteetin kannalta varmistettiin, että faktoreiden mittaamiseen käytetyt muuttujat toimivat myös Kreikan osalta. Näin ollen tutkijat tekivät aluksi rakenteesta eksploratiivisen faktorianalyysin, jonka pohjalta he poistivat kaksi muuttujaa prosessin laadusta (liittyivät vaatimusten hallintaan) alhaisen faktorilatauksen[2] (< 0.5) vuoksi. Cronbachin alfa -tunnusluvun mukaan tuloksia voidaan pitää reliaabelina ja yhtenäisinä.

Kokeellisen faktorianalyysin jälkeen tutkijat tekivät konfirmatorisen faktorianalyysin, jonka tarkoitus on vahvistaa tai kumota ennalta määritelty faktorirakenne. Testien tulokset tukevat ehdotettua mallia. Myös toisen kertaluvun faktorianalyysin riittävyysarvioinnissa kaikki arvot olivat hyväksyttäviä: esimerkiksi GFI-arvo (goodness of fit -indeksi) on yli 0.90, joten mallia voidaan pitää riittävänä. Myös RMR (root mean square residual) -indeksin lähellä nollaa olevat arvot osoittavat mallin toimivuutta (projektin riski 0.002 ja molemmat laatutekijät 0.000).

Regressioanalyysin mukaan tulokset viittaavat siihen, että laadun ja riskin välillä on tilastollisesti merkitsevä negatiivinen riippuvuus (−0.293). Kun laatu-faktori otetaan erikseen huomioon ja jaetaan ihmisistä ja prosesseista johtuviin tekijöihin sekä riski kuuteen dimensioon, niin silloin ihmisistä johtuvan laadun ja viiden kuudesta riskidimensioista huomataan vaikuttavan negatiivisesti (ja tilastollisesti merkittävästi). Prosessin laatu vaikuttaa negatiivisesti vain tiimi-ulottuvuuteen (-0.177). Kompleksisuuteen kummallakaan ei ollut tilastollisesti merkitsevää vaikutusta. Tulokset vastaavat muuta kirjallisuutta, jossa alleviivataan laadun tärkeyttä riskien vähentämisessä. Tärkeimpänä tuloksena voidaan pitää sitä, että erityisesti ihmisiin liittyvillä laatutekijöillä on kielteinen vaikutus

[1] Esimerkiksi muuttujan User1 ehdotettu riski oli ’Käyttäjien muutosvastaisuus’ ja muuttujan Team1 ehdotettu riski ’ Kokemattomat tiimin jäsenet’. Tarkemmin tutkimuksen liitteessä: Appendix A. Questionnaire.

[2] Kyseisissä poistetuissa tapauksissa se oli < 0.5. Faktorilatauksen arvo kertoo kuinka paljon ko. faktori pystyy selittämään muuttujan vaihtelusta.

projektin laatuun.

Jotta tuloksia voitaisiin yleistää, niin vastaavia tutkimuksia olisi tehtävä myös muissa maissa. Toinen tärkeä kysymys liittyy tutkijoiden mukaan strukturoidun kyselylomakkeen käyttöön tietojen keräämisessä. Huolimatta sen tarkasta suunnittelusta ja testaamisesta, on mahdollista, että joihinkin vastauksiin on vaikuttanut kyselylomakkeen muotoilu ja koko. Ihannetapauksessa strukturoitu tai osittain strukturoitu kaikkien osallisten haastattelu olisi selventänyt joitakin ongelmakohtia.

Johtopäätökset

Ainutlaatuiseksi tutkimuksen tekee se, ettei Kreikassa ole pyritty testaamaan riskienhallintaan liittyvissä tutkimuksissa empiiristä pätevyyttä ja tarkkuutta teoreettisia lähestymistapoja käyttäen. Riskeille altistumisessa (risk explosure, RE) käytettiin ensimmäistä kertaa (Kreikassa) matemaattista kaavaa[3] $RE = P + C - (P * C)$ (missä $P =$ esiintymismahdollisuus[4] [possibility] ja $C =$ vaikutus [consistency]; empiirinen arviointi varmisti sen pätevyyttä ja tarkkuutta.

Tutkimus perustuu aineistoon, joka on kerätty usealta projektitiimin jäseneltä joilla on erilaisia ominaisuuksia. Tämä voi parantaa tutkimuksessa käytettyjen parametrien arviointia ja vähentää subjektiivisia päätelmiä, koska eri näkökulmia eri päätöksentekotasoilla on otettu väkisinkin huomioon. Tutkimuksessa henkilöiden asema ei kuitenkaan näyttäisi johtavan toisista poikkeaviin johtopäätöksiin.

Muuta huomioitavaa tutkimuksessa on, että riskien mittauksessa käytetään peräti kuutta ulottuvuutta, joissa on yhteensä 27 mitattua muuttujaa. *Vaatimus*-ulottuvuuden muuttujilla oli korkeimmat keskiarvot, mikä tarkoittaa sen suurinta altistumista riskeille (vastaajien mielestä). Sitä vastoin *organisaation ympäristö* näyttäisi olevan vähiten merkittävä riskitekijä. Tämä voi tutkijoiden mukaan johtua siitä, että Kreikan ohjelmistokehitysyritykset toimivat vakaissa organisaatioympäristöissä, joissa ei ole merkittäviä hallinnollisia muutoksin projektien toteuttamisen aikana.

Keskiarvot laatuun liittyvissä muuttujissa ovat melko samanlaisia, mikä viittaa siihen, että osallistujat ovat kaiken kaikkiaan erittäin tyytyväisiä laatuun. Henkilökunnan laadullisen ulottuvuuden (henkilöstön yleinen kokemus, kokemus käytettävästä ohjelmointikielestä, alhainen henkilöstön vaihtuvuus) osalta keskiarvo on erittäin tyydyttävä (korkeampi kuin muilla). Tämä osoittaa muun muassa sen, että yrityksissä ollaan tyytyväisiä työntekijöiden tietämystasosta sekä kyvyistä. Edellinen selittää jossain määrin, miksi ulottuvuutta *organisaation ympäristö* pidetään pienenä riskinä.

Projektin laatu jaettiin kahteen pääosaan, joilla mitattiin yksittäisiä vaikutuksia kuuteen riskidimensioon. Ihmisiin liitetyllä laadulla on negatiivinen ja tilastollisesti merkittävä suhde kaikkiin riskidimensioihin (ilman riskejä, jotka liittyvät projektin kompleksisuu-

[3] Yleisemmin käytetty kaava on $RI = P * C$, jolloin korkea vaikutus mutta pieni mahdollisuus tuottaa suhteellisen korkean riskiesiintymisen.

[4] Suomalaisissa projektioppaissa käytetään usein termiä *todennäköisyys* (probability), vaikka sitä ei voi laskea matemaattisesti. Kun kyse on subjektiivisesta arviosta, niin silloin olisi parempi käyttää termiä *mahdollisuus* (possibility). Esimerkki tästä käsitteellisestä epämääräisyydestä löytyy kurssin projektiwikin dokumenttirungosta (http://www.cs.tut.fi/~projekti/dokumentit/proj-sisalto.txt).

teen). Tämä havainto viittaa artikkelin mukaan siihen, että Kreikan ohjelmistokehitystä tekevät yritykset kiinnittävät huomiota asianmukaiseen henkilöstöön, koulutukseen ja henkilökunnan viestintään. Prosessiin liittyvät laatuominaisuuksilla ei näyttäisi olevan merkittävää vaikutusta riskialttiuteen. Ainoastaan tiimiin liittyviin riskeihin on negatiivista vaikutusta prosessin laadulla. Tutkijoiden mukaan syynä tähän voi olla Kreikan yritysten suhteellisen pieni koko.

Kokonaisuuden kannalta mallin yksittäisillä ulottuvuuksilla on vain pieni merkitys. Kuitenkin riskienhallinnan menetelmät, jotka keskittyvät vain yhteen riskinäkökulmaan, voivat johtaa suurempiin riskien aliarviointeihin ja sitä kautta epäonnistumisiin. Artikkelin kirjoittajien mielestä projektien johdon tulisikin käyttää vastaavia välineitä, joiden avullaan voidaan arvioida ja hallita riskejä. Tulokset osoittavat, että 27 tutkimuksessa ehdotetun riskin mahdollista esiintymistä ja luokitusta voidaan hyödyntää keskittymällä suhteelliseen pieneen määrään riskejä, joista saadaan suurin mahdollinen hyöty (Pareto-jakauma).

Kirjoittajat ehdottavat, että ohjelmistoyritykset kehittäisivät tämän tutkimuksen pohjalta riskiprofiileja jokaiselle ohjelmistoprojektilleen. Riskienhallintamallia voitaisiin käyttää riskien ositteluun ja evaluaatioon projekteissa kahdellakin tavalla:

1. Työkaluna eri projektisyklin vaiheissa, jolloin valvottaisiin projektin riskitason muutoksia.
2. Analyysityökaluna valmistuneisiin projekteihin. Jatkuvan seurannan tulokset hyödynnettäisiin siten tulevissa projekteissa.

Ajan myötä projektien johto saisi täten arvokasta tietoa, jonka avulla voidaan välttää tai resursoida riskihankkeet oikein. Projektisalkkua voisi myös painottaa tasaisemmin korkean ja alhaisen riskin projekteilla.

Jari Smedberg

Project management, governance, and the normalization of deviance

J.K. Pinto, International Journal of Project Management, volume 32, issue 3, pages 376-387, 2014

Background

In the year 1986 the Challenger space shuttle disintegrated 73 seconds after launch. In the year 2003 the Columbia space shuttle fell apart while reentering atmosphere. Cruise ship Costa Concordia drifted to shallow water near an Giglio Island in the year 2012 killing 32 of the 4252 passengers. What all of these have in common? Each of the disasters could have been easily avoided.

During the research and development of Challenger shuttle subcontractor's engineers discovered that O-rings of solid rocket boosters were leaking due to weak putty. The engineers tested for different options to fix the putty and found satisfying solution for the problem. Even though the O-rings still occasionally leaked during test flights they still approved the shuttle for the final launch. The production got so accustomed to the flaw that it became normal and acceptable.

The second shuttle, Columbia, had made 28 successful flights before the disaster. During the flights there was numerous reports that foam insulation batters during the liftoff against external fuel tanks. Ultimately this caused underside heat shields break off during the take-off of the flight that ended up in the disaster.

In the case of Costa Concordia it was accustomed to cruise ships to go unapproved course near the island which lead to the disaster. Further investigation showed that it was not only approved behavior but also promoted by the company's directors as a "convenient, effective marketing tool" (p. 377).

These incidents have in common that people involved tolerated deviance. This behavior is named as normalization of deviance which is best described as "unexpected becomes the expected, which becomes the accepted" (p. 382). The paper explores through interviews with project managers how normalization of deviance affects projects and ways to minimize the problems caused by normalization of deviance.

Results

The research was conducted by interviewing 21 project managers from 3 corporations. After the interview Pinto analyzed the answers and identified characteristics of different types of normalization of deviance phenomenons using Q-sort method. After that Pinto suggests a way to identify normalization of deviance in own company and how to stop doing it. Pinto found out three different categories where normalization of deviance occurs. First being at project proposals and strategic misrepresentation. The next is client/contractor relationships and lastly planning and scheduling dynamics.

The first category project proposals and strategic misrepresentation relates to the "project proposal, bidding, and business stage development" (p. 378). This was usually achieved by holding information related to project planning and bidding. Pinto gives and example from the year 1972 in the city of Calcutta. At that time the city was planning of building metro system. The Prime Minister of India Indira Gandhi was about to visit planned metro terminal site. Right before her visit West Bengal Chief Minister Siddharta Ray ordered the sites ground dug up and leave huge piles of dirt near the site. When Gandhi got to the site she asked about the state of the construction on which Ray replied that the state decided to continue the project no matter what government decided. This caused the government act and give funds to the project which ultimately ended up funding the whole project.

The second category found was about client/contractor relationships where the two parties perceive each other as the "rival camp" (p. 380). Through out the project relationship varies between positive and negative. At the beginning the relationship is positive and the contractor promises that almost everything is possible while some of the promises end up being the subject of conflicts later on the project. When the project nears up the delivery the relationship goes to positive again which again drops when final agreements are done upon what is included in the delivery. The fact that customers rarely repeatedly order new projects from the company resulted in decreased the need of maintaining positive relationship with the client.

The last category found was about planning and scheduling dynamics which occur within a company. As for planning here the same atmosphere of "rival camps" occur against for example the developing team and marketing. One example points out that one project manager made clear to production team that all communication and change requests go through the project managers office. The same project manager also emphasizes the fact that competitors company is not the enemy but the marketing is. For the scheduling the deviant behavior usually was that managers usually cut some percentage off the time estimates, and when schedule planner eventually catches on this cutting behavior they usually add the same or some more percentages the estimate before presenting it to the manager.

Pinto suggests five step process for countering normalization of deviance in governance process. Step one is process analysis which is crucial first step to find out what kind of normalization of deviance behavior does the company conduct. This step usually requires outsiders help in order to figure out the devious behavior. The second step is to educate organizational members. The idea here is to teach the organizational actors to identify normalization of deviance from the operating processes and what motives lies under them. The third step is to clarify standards of appropriate behavior throughout the organization. The fourth step is to ensure transparency throughout the organization. This means that the actions of all the organization's members are judged with the standards in mind. The last step is reward compliance with the new standards. Reward system will support newly introduced standards and it will work better than punishing of wrong doings.

Conclusions

The paper points out three different ways that normalization of deviance appears in organizations. How much they will ultimately affect on average can be argued. For instance extra padding in planned schedule which is cut by manager will lead up to actually realistic schedule. The client/contractor relationship fluctuation was pointed out by only one company which actually made mostly contracts worth billions which might explain the lack of returning customers. Customers coming back for more projects compels the organization to maintain customer relationship. Ultimately the tolerance of deviant behavior of people or developed system can lead to catastrophic disaster.

Jarno Mynttinen

IT Project Management: developing on-going skills in the management of software development projects

A. Tatnall and P. Shackleton, in Proceedings of the Inaternational Conference of Software Engineering: Education and Practice, pages 400-4005, 1996

Johdanto

Projektinhallintaa tai projektinhallinta-ohjelmistojen käyttöä ei yleensä opeteta korkeakouluissa omana kurssinaan, vaan se on sisällytetty muihin kokonaisuuksiin. Opiskelijoille syntyy usein käsitys, että projektinhallinnan tarjoamat työkalut ovat tarpeellisia vain projektin suunnittelussa ja hylkäävät niiden käytön projektin edetessä. Tatnall ja Shackleton ehdottavat artikkelissaan opetusmuotoa, jossa fiktiivisen projektin ja roolipelaamisen avulla voidaan tuottaa riittävän monimutkainen projekti, jonka dynaamiset tilanteet kannustavat opiskelijoita käyttämään Microsoft Project-ohjelmiston tyyppistä työkalua projektinhallintaan liittyvien ongelmien ratkaisussa.

Tulokset

Tatnall ja Shackleton opettivat tietojenkäsittelyä Victorian teknisen yliopiston (VUT) kauppatieteen laitoksella. He opettivat projektinhallintaa ja Microsoft Project-ohjelmiston käyttöä 'Systems Implementation' -kurssilla, jonka tarkoitus oli tukea laajempaa 'Computing Project' -kurssia. Jälkimmäisellä kurssilla kuitenkin huomattiin, että vaikka opiskelijat osasivat käyttää projektinhallintaohjelmistoa projektin suunnitteluun, sen käyttö lopetettiin suunnitteluvaiheen jälkeen.

Tekijät päättelivät, että esimerkkiprojektit eivät olleet riittävän monimutkaisia, jotta projektinhallintaohjelmiston käyttö olisi tuntunut opiskelijoista tarpeelliselta. 'Systems Implementation' -kurssin toteutusta muutettiin seuraavana lukuvuonna siten että se sisälsi fiktiivisen projektin, jonka asiakkaana toimivat kurssin opettajat ja projektipäällikköinä opiskelijat. Opiskelijat loivat ensimmäisenä suunnitelman ohjelmistoprojektille ja sen jälkeen he saivat viikoittain raportoinnin fiktiiviseltä kehitysryhmältä projektin etenemisestä. Kehityksessä tuli vastaan ongelmia, jotka pakottivat opiskelijat muokkaamaan alkuperäistä suunnitelmaa ja allokoimaan resursseja uudestaan.

Opiskelijoiden kokemuksia kerättiin yksittäis- ja ryhmäkeskusteluissa. Kurssiin liittynyt epävarmuus oli monelle uusi kokemus. Epävarmuustekijöitä olivat esimerkiksi se, etteivät opiskelijat voineet vertailla toistensa vastauksia, oikean vastauksen ollessa monikäsitteinen. Myös projektin dynaaminen luonne loi epävarmuutta ja turhautumista, koska kaavioita ja suunnitelmia jouduttiin päivittämään viikkoraporttien luoman uuden informaation pohjalta. Myöskään opiskelijoiden yleensä käyttämä viime tipassa -strategia ei toiminut tällä kurssilla, koska viime hetkellä havaittuihin ongelmiin ja niistä muodostuviin kysymyksiin ei enää ehtinyt saamaan ajoissa vastauksia.

Projektinhallintaohjelmiston käytön korkea oppimiskynnys ja aloittelijalle epäintuitiiviset konseptit aiheuttivat myös ongelmia. Jos ohjelmistoa halusi käyttää ongelmitta, joutui selvittämään miten ohjelmiston suunnittelijat olivat sitä kuvitelleet käytettävän. Suurin osa opiskelijoista piti kuitenkin kurssia hyvänä opetuksena oikean elämän realiteeteista. Tuloksia on kuitenkin hankala arvioida ennen seuraavaa 'Computing Project' -kurssia, jossa tullaan näkemään käyttävätkö opiskelijat projektinhallintatyökaluja koko projektin ajan.

Johtopäätökset

Roolipelaamista muistuttava menetelmä päästi opiskelijat johtamaan projektia, jonka toteuttaminen ei olisi käytössä olleilla todellisilla resursseilla onnistunut. Tämä mahdollisti projektin monimutkaistamisen tasolle, jolla projektinhallintaohjelmiston käytön hyödyt korostuivat ja opiskelijat oppivat käyttämään ohjelmistoa apunaan myös projektin aikana. Oppimistuloksien arviointi perustui osin opiskelijoiden toimintaan seuraavalla kurssilla ja ei näin ollen sisältynyt artikkeliin.

Johannes Salminen

When agile meets the enterprise

G. van Waardenburg and H. van Vliet, Information and Software Technology, volume 55, issue 12, pages 2154-2171, 2013

Background

In the last ten years there have been increase in the use of agile methods across the field of software development. Now large traditional software development corporations are also looking to use these methods for quicker development cycles and therefore increased profitability.

Agile methods are originally meant to be used by small sized development teams, not by enterprises of hundreds or thousands of workers spread throughout the world in different offices. Taking the agile methods into these enterprises lead to clash of two significantly different methods and this does not come without consequences.

Tradition plan-driven methods emphasizes planning and clearly defined phases as agile methods are based on uncertainty of the future and “just do it” thinking. Changing plans in plan-driven methods is slow and therefore avoided as in agile methods are made to react fast for changes. Agile and plan-driven methods are like black and white – they are opposites of each other. This research takes a look at what happens when these two collide.

Results

The research presented in this article conducted by G. van Waardenburg and H. van Vliet takes a look at the consequences of the clash of these two development methods. The research was done using Grounded theory method and the data was collected by interviewing 21 members of two large companies. The interviews were semi-structured. These enterprises both use traditional plan-driven methods alongside agile development methods. Agile methods had been used in Company A for one year and three years in Company B.

The biggest downside of including agile methods in the enterprise’s development methods was increase of IT landscape complexity, especially for the Company A. The landscape was increased by number of reasons. These include complexity increased by simultaneous development streams, which is already a challenge in traditional development methods. Constant communication between project members is essential part of agile methods, which lead to problems with companies having separate teams for front-end and back-end parts of the project. Different development methods in different parts of the project also increased the complexity of IT landscape as some part were done in traditional methods and others in agile methods.. Using of independent contractors for some parts of the project was also problematic in the agile environment as those contractors had different development culture.

The most critical consequences of increased IT landscape complexity were defined as problems in communication, defining of what was meant with the word “done” and implementing changes to the requirements. Changes to requirements are especially hard as those parts of the project which use tradition development methods wants to lock down requirements in early stages as the agile methods emphasize evolving the requirements as the project goes on.

These two enterprises have developed strategies to manage the increased IT landscape complexity. One of these strategies is to combine the product backlogs of all the teams working on the project. This helps with planning and keeping track with project priorities. Communication problems are countered with project teams having representative team members working within other project teams to provide knowledge of what is going on in other parts of the project. Project Managers must renew themselves by shifting their role from directive role to more facilitating one. The manager must also be more involved in the team process to maintain his influence over the team.

Lack of business involvement was one of the challenges confronted in these two enterprises after taking the agile methods to use. In agile development method business involvement is necessary as changes to the project often happen. The lack of business involvement was seen to be caused in the research by two causes. Businesses are not used to be involved in the project after the requirements have been agreed on. Centralized IT departments in the companies divide a larger gap between business and the development team.

The lack of business involvement led to problems especially with the requirements. Gathering the requirements from business representatives after each iteration was seen problematic and businesses didn’t want to prioritize requirements as they saw each requirement equally important. Lack of feedback from businesses was also a large problem as this also affected how quickly changes to the requirements could be made.

A few solutions to the lack of business involvement was presented in the research. Businesses must be informed about the agile process and explained how it works to get them to realize the importance of communication between the two parties. Another solution was to get one representative from the business to act in the Product Owner role of the Scrum development method and therefore channel the business knowledge. Intensive stakeholder communication was also seen necessary to increase business involvement. This included collaborative demonstrations of product functionality, increased face-to-face communication and weekly stakeholder sessions.

Conclusions

Introducing agile development methods into large traditional software enterprises leads to increased IT landscape complexity and lack of business involvement. These challenges can be mitigated with strategies, but these needs an effort from both the development team and the business stakeholders of the project to succeed.

Lauri Nykänen

From dynamic capabilities to ERP enabled business improvements: The mediating effect of the implementation project

Edward W.N. Bernroider, Christina W.Y. Wong , Kee-hung Lai, International Journal of Project Management, volume 32, pages 350–362, 2014

Background

Enterprise Resource Planning, initially proposed by Gratner Group Inc. in 1990 as the application software, is now adopted by commercial enterprises all over the world and became increasingly important. However, many ERP adoption often leads to failure, with only 13% reaching customer expectations and more than 50% of firms believe that their ERP adoption were unsuccessful(Panorama Consulting Group, 2009). In this case, seeking the reason to promote ERP adoption is particularly important.

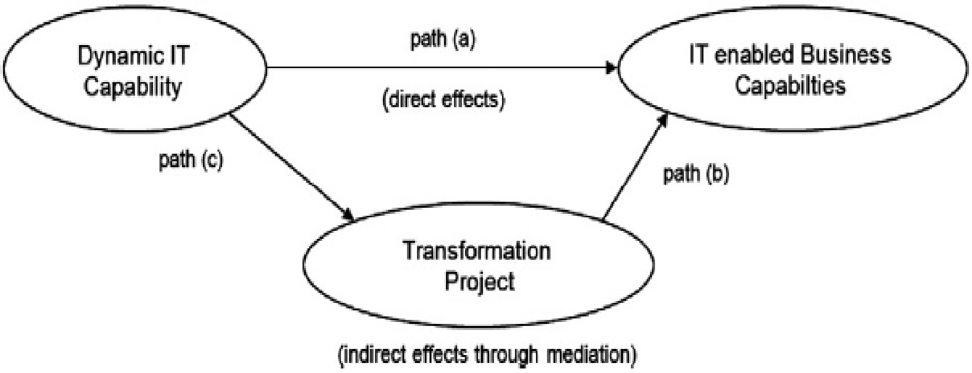
In the introduction, the author addresses the point that previous study mainly focuses on the influences of implementation project capabilities like specific IT project related criteria, project management practices and organizational pre-adoption characteristics, but fails to clearly point out how these capabilities effect on improving IT enabled business capabilities.

To begin with, the paper introduces three dynamic pre-adoption capabilities(external information acquisition, IT decision making, IT governance) essential to ERP implementation; Furthermore, other than the prior research, it suggests that project management needs to be considered in dynamic transformation capability and reveals its relationships between dynamic capabilities; Last but not least, it highlights the positive effects of considered dynamic capabilities, and furthermore, gives that how external information acquisition and IT governance affects as a mediator depend on the implementation.

Results

Based on the literature review, several variables are conceptualized and measured as evaluation criterion. In this study, 57 Austrian ERP adopters, with more than 249 employees or an annual turnover exceeding EUR 50million are surveyed. Respondents were asked to assess given questions for various indicator variables, covering external information acquisition (IA), Decision Making (DM), IT governance (ITG), IT enabled business capabilities (BC) as well as Project performance (PP).

The statistical methods applied in this study includes calculating descriptive statistics with SPSS (version 19) and structural equation modeling (SEM) combined with mediation analysis.

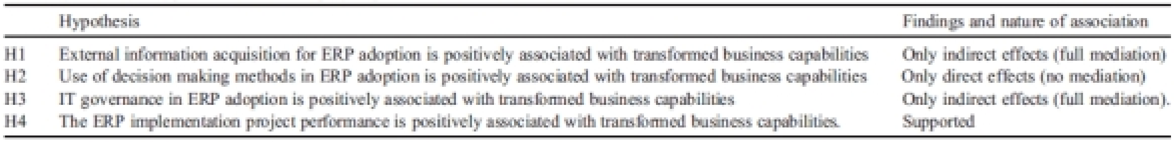


The main intention of this study is to identify the mediating effect of the implementation project between dynamic capabilities and ERP-enabled business improvements. The paper developed a research model to study how the transformation of organization mediates the anticipated positive effects that shown in figure above (p4).

Furthermore, four hypothesis are given and further identified (Table 6) based on this model.

1. External information acquisition (INF) for ERP adoption is positively associated with transformed business capabilities is positively associated with transformed business capabilities (directly and indirectly).
2. Use of decision making methods is positively associated with transformed business capabilities (directly and indirectly).
3. IT governance in ERP adoption process is positively associated with transformed business capabilities (directly and indirectly).
4. The ERP implementation project performance is positively associated with transformed business capabilities.

Table 6
Summary of tests of hypotheses (with ERP project performance as mediator).



	Hypothesis	Findings and nature of association
H1	External information acquisition for ERP adoption is positively associated with transformed business capabilities	Only indirect effects (full mediation)
H2	Use of decision making methods in ERP adoption is positively associated with transformed business capabilities	Only direct effects (no mediation)
H3	IT governance in ERP adoption is positively associated with transformed business capabilities	Only indirect effects (full mediation).
H4	The ERP implementation project performance is positively associated with transformed business capabilities.	Supported

As a result, the survey suggests that ERP projects are necessary condition for positive effects of information acquisition and IT governance effects, while decision making showed insufficient connection to business capabilities gained by ERP.

Conclusions

This research draws a conclusion that the capacity for external information acquisition project and IT governance mechanisms influence ERP enabled business capabilities indirectly by the ERP implementation project. For external information acquisition, since it is regularly supported by vendor-driven information channels and consultants, the information acquisition capability for ERP might be a function of a dominating external gatekeeper's role; For IT governance, the ERP project seems to be a full mediator for improving ERP business capabilities. Thus, the transformation project helps to clarify the relationships between these two dynamic capabilities and ERP enabled business capabilities as mediator.

Conversely, although the third capability decision making exerts a direct positive association and no indirect relationship with ERP enabled business capabilities, it is not proved to have the same mediational role, which has not been referred in prior research.

Ma Xiaolan

An architectural model for software testing lesson learned systems

J. Andrade, J. Ares, M.-A. Martínez, J. Pazos, S. Rodríguez, J. Romera, and S. Suárez , Information and Software Technology, volume 55, pages 18–34, 2013

Background

Software testing is the dynamic verification of actual program behavior on a finite set of test cases.(p18) The altitude of people has changed from these years. Software testing has become increasingly important, critical and complex. Also, there are many testing techniques applied in kinds of testing models, such as machine learning techniques, adaptive random techniques.

But at the same time, testing engineers perform similar tasks and have similar problems day in day out as they work on different projects. (p19)Because the experiences are not gained by others, therefore, the same mistakes are made over again. In this paper, the authors propose that the experience can be applied in the future by other team if the organization is capable if extracting that individual experience and making it available to use by anyone who need it.

Thus, this paper defends the use of a lesson learned system for software testing. This system is an effective knowledge management resource enabling testers and managers to take advantage of the experience from the others. By this way, the experience has to be gathered, disseminated and reused.

Results

After analyzing the proposals for managing software testing experience, we have notified the weaknesses from current methods. The architectural model proposed here for lesson learned systems is designed to avoid these weaknesses.

"A lesson learned is a knowledge or understanding gained by experience."[5][1] Its aim is to convert people's experience-derived individual knowledge into organizational knowledge through reuse.(p21) To manage these experiences, lesson learned systems implement the processes including collection, verification, storage, dissemination and reuse.

The architectural model proposed here manages knowledge gained by experience that has the potential for future testing projects. Here is the representation scheme used for the lesson learned repository design, which includes five main blocks: Generic, Context, Experience, Lesson Learned and Reuse.

[1] P.. Secchi , R. Ciaschi, D. Spence, A concept for an ESA lessons learned system, in: Proceedings of Alerts and LL: An Effective Way to Prevent Failures and Problems, The Netherlands, 1999, pp. 57–61.

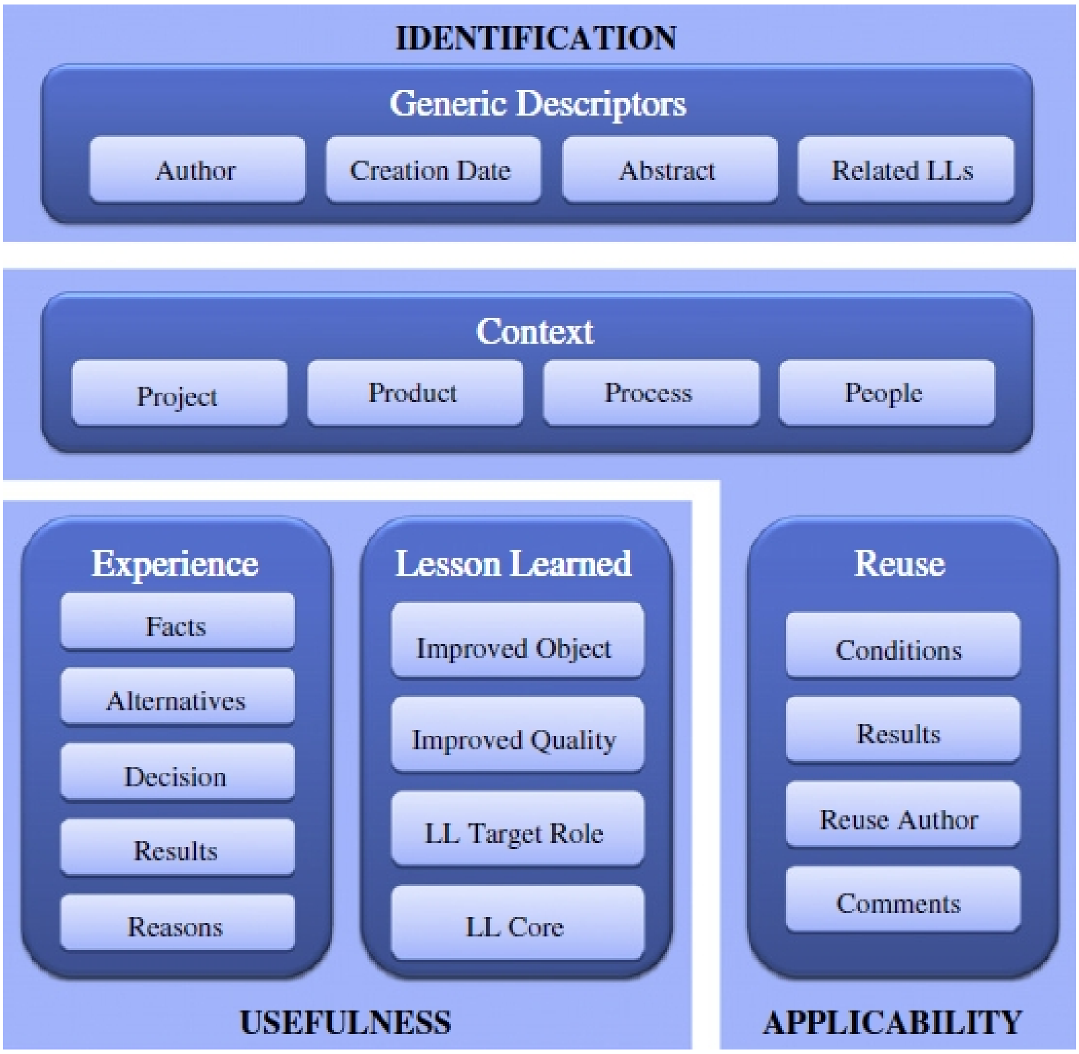


Fig.1 Software testing LL scheme.(p23)

Also, this paper proposes a web-based software testing lesson learned system prototype with the aim of evaluating the strengths and weaknesses of the core of the proposed model. In sum, this paper proposes a different approach, based on the management of the lessons learned that software testing engineers gather from everyday experience.

Conclusions

The architectural model proposed here provides guidance for developing software testing lesson learned systems. And the model designed to avoid the current weakness and take into account two basic goals: usefulness and applicability. (p32)At the same time, we find there are still some problems existing. It ware hard to use in a real lesson learned system, and knowledge sharing increased the employee workload. Also, the dissemination of relative experience need more efficient. It should make team member be interest of it.

Menglin Xu

An optimization method for selecting project risk response strategies

Y. Zhang and Z.-P. Fan, International Journal of Project Management, volume 32, number 3, pages 412–422, 2014

Background

Risk is a harmful factor that can manifest at any point in a project. Due to its negative consequences, specifically, the failures in meeting the budget, schedule or quality requirements, project risk management is seen as an important discipline. It consists of three phases: risk identification, dealing with the documentation of possible risks; risk assessment, the close examination and evaluation of the identified risks' probabilities and possible impact; risk response, the selection and implementation of actions that could lower those risks' likelihood and impact.

Selecting appropriate risk response strategies is considered an important part of project risk management, yet it is also the least studied one. Project managers generally rely on prior experience when selecting these strategies, but they do not have quantitative models to assist in their choice. The article proposes one such model on the basis of the project's work breakdown structure and identified risks. The model represents an integer programming problem that, when solved, provides the optimal solution in terms of the risk response strategies. Should this solution be unacceptable to the project managers, an iterative process is utilized to lead to a more desirable selection. The model and this process rely on trade-offs between three factors: the cost of implementing chosen strategies, project schedule and project quality.

Previous studies related to the selection of adequate risk response strategies are classified into four major groups.

Firstly, the zonal-based approach suggests mapping two risk-related criteria to two axes. Pairs that have been chosen for this method include probabilities of immediate and external project risks, the controllability and generality of the risks. Different values of the chosen criteria allow splitting the graph into multiple zones and populating them with possible risk response strategies. Ultimately, those strategies which yield the desired values of the two criteria are selected. However, this approach cannot be extended to more than two criteria.

Secondly, the trade-off approach involves making trade-offs between risk-related criteria such as cost, duration, quality, success probability, etc. with respect to objective requirements and subjective preferences. Out of the candidate response strategies obtained in this process, a selection is then made according to the principles of efficient frontier, Pareto optimality or a similar concept. When more than two factors are involved, this method is restricted to qualitative analysis only.

Thirdly, the WBS-based approach attempts to relate risk response strategies to actual work activities within the project. (A work breakdown structure, or WBS, is a tree diagram that describes these work activities and displays their scope in terms of schedule, quality, and cost.) Strategies can be produced for actual activities or prototype activities that somehow relate to actual ones; however, whether these strategies will be optimal is unknown.

Finally, the optimization-model approach seeks the answer to the strategy selection problem in optimization theory. The objective function is the cost of implementing chosen strategies, and it is constrained by the possible combinations of strategies, acceptable risk reduction, the budget allocated for the implementation and other factors.

The need for a new approach to risk response strategy selection, in addition to all of the listed methods, is due to their various limitations, such as the necessity of considering only two criteria or, on the other hand, the lack of a rigorous mathematical procedure. The model proposed in the paper, while based on the WBS approach, represents an optimization problem, seeking to maximize the obtained risk response effects, considering the cost of implementing strategies, relations between them and schedule and quality requirements.

It is noteworthy that the risk response strategies covered in the article fall into the category of risk mitigation – that is, attempting to counter identified risks and reduce their probability of occurrence or potential impact, thus also reducing expected loss from such risks. Other forms of risk response include avoidance, acceptance and transfer.

Results

The model makes use of binary decision variables in the optimization problem. That is, variables corresponding to candidate risk response strategies take the value 1 if the strategy is chosen and 0 if it is rejected. Consequently, the problem can be solved with the so-called zero-one integer programming methods.

The solution to the problem posed by the model is a set of decision variables that yields the maximum effect from implementing risk response strategies. The model also includes two classes of constraints: one handles the available budget for the implementation, as well as the required quality and duration of every activity in the project, while the other specifies possible dependencies within the strategies. For every possible pair of strategies, there can be weak exclusion, meaning that both cannot be selected at once, strong exclusion, meaning that one and only one of the two must be selected, or cooperation, meaning that selecting one necessitates selecting the other.

The model relies on four assumptions reducing the complexity of the actual project risks: that risk events are mutually independent; that they have a negative impact on the work activities; that risk response strategies have a positive impact on those; and that money is the only resource constraint.

Ultimately, the optimization problem can be expressed in the following manner:

Select the values of the decision variables corresponding to selected risk response strategies, maximizing the cumulative risk response effect (or minimizing the expected loss from risk events), so that

- the cost of implementing the chosen strategies does not exceed the allocated budget;

- implementing the strategies, together with the occurrence of risk events, will not delay subsequent activities beyond the minimum amount of time between them;
- implementing the strategies, together with the occurrence of risk events, will not degrade the quality of the activities beyond the point when subsequent activities are affected;
- the final activity will be completed before the delivery time plus the acceptable delay;
- the quality of the final activity satisfies the maximum quality reduction for the project;
- out of any two risk response strategies bound by weak exclusion, at most one is chosen;
- out of any two risk response strategies bound by strong exclusion, one is always chosen;
- out of any two risk response strategies bound by cooperation, both or none are chosen;
- all the decision variables take the values 0 or 1.

Before the model is utilized, the project managers and the development team must discuss possible risk response strategies. They will likely rely on experience in previous projects both when suggesting candidate strategies and when estimating their effects. The preparations involve an analysis of the project scope and WBS, as well as risk identification and assessment.

If the obtained solution of the problem is not satisfactory, a stopping rule corresponding to an acceptable solution is proposed. Trade-offs are then made between the budget, time and quality parameters of the model, and the optimization process is conducted upon every such adjustment until the rule is satisfied.

The paper provides an extensive example covering the construction of a ventilation and air conditioning system, starting from the creation of the project WBS, identification of risks and assessment of their costs, and concluding with the optimization itself. Sensitivity analysis indicates that the overall risk response effect is robust with respect to project schedule and quality for certain budget sizes, and grows more sensitive to these criteria as the budget decreases. Similar dependencies are observed between other parameters with varying acceptable quality or schedule. The model's application yields several strategy selections, one of which is to be undertaken according to the project managers' preferences (high quality, low budget, etc.).

Conclusions

The article proposes the usage of an optimization model (integer programming) as a method of selecting the most appropriate risk response strategies in project risk management. Based on WBS analysis, the model integrates three common constraints in the field, namely, project cost, schedule, and quality.

It is shown that the resulting risk response effect exhibits robust behaviour when these constraints are sufficiently relaxed. If this is not the case, the model allows for trade-offs to be made between the current values for the risk management budget, project schedule and quality; the optimization process is repeated after every such alteration until a feasible selection of risk response strategies is obtained.

One drawback of the suggested approach is the interference of the human factor, which manifests itself in that the risk response strategies are handled only by the project managers, yet the risk situations themselves will be likely seen differently by the team members. Ideally, the choice of feasible strategies shall account for individual preferences and feelings.

Moreover, the suggested model relies on the assumption that all risk events are independent from each other for simplification purposes, and incorporating interdependencies between risks is seen as a research perspective. Specifically, candidate strategies may have to account for such dependencies, their strength and the way they change over time. Another problem is the growing complexity of the integer programming problem that comes with large numbers of variables and constraints (at least in terms of finding an exact solution), which may call for other optimization methods such as genetic algorithms.

Mykola Andrushchenko

Managing project changes: Case studies on stage iteration and functional interaction

L. Zhang, International Journal of Project Management, volume 31, issue 7, pages 958-970, October 2013

Background

Project change in complex project is always a big problem to solution-oriented companies, and is as well a popular topic in software development research. Some critical changes are unavoidable and unpredictable in complex projects. How to deal with these critical problems can be an important factor, influencing deeply on the project quality.

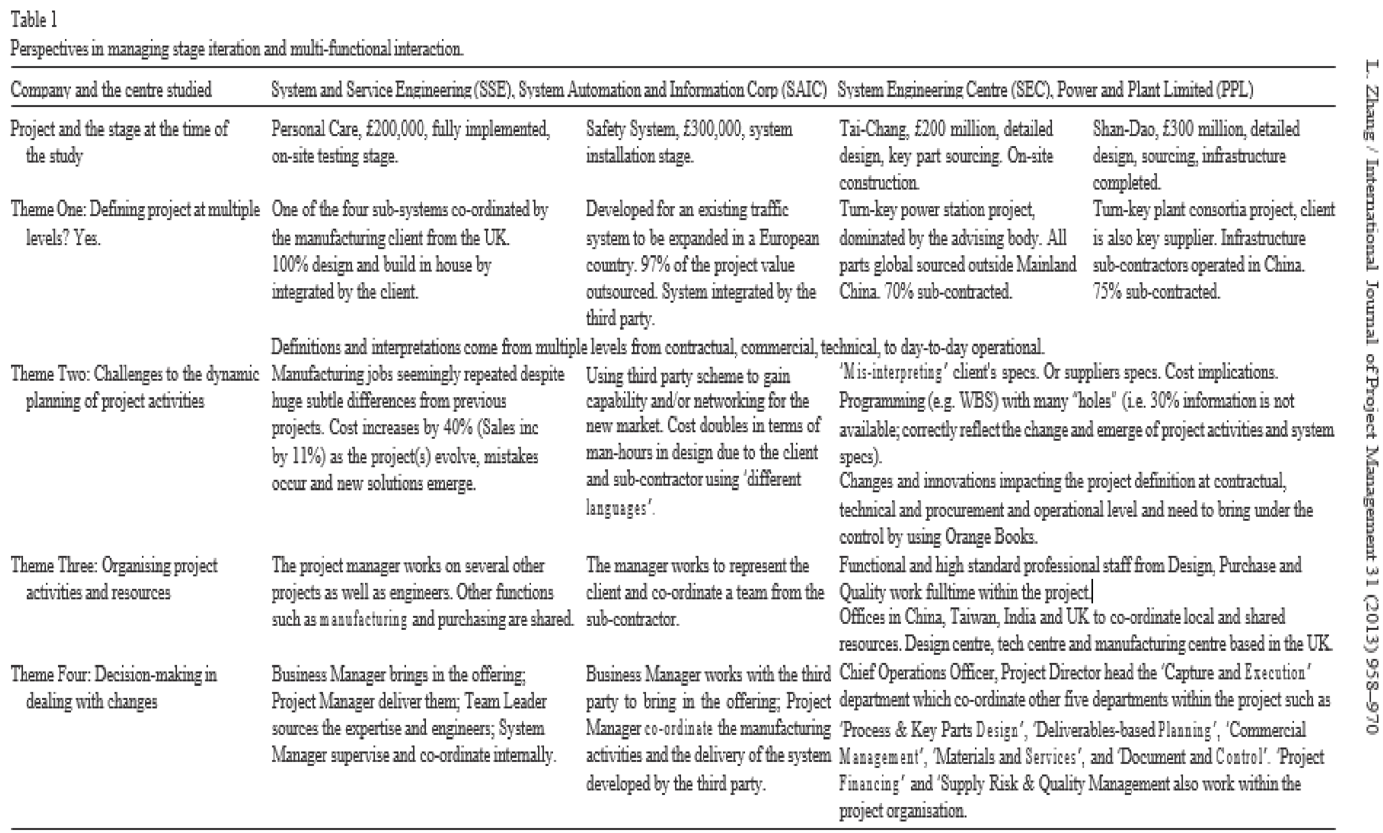
Project planning for complex projects is based on future assumptions, which are not stable and even wrong. When changes occur in a complex project, a part of related issues should be maintained and resolved. Commonly, a project could be regarded as a multi-stage process and a multi-functional effort. Stage iteration and functional interaction can be the key to solve this kind of problems.

In order to resolve problems caused by changes in complex projects, this paper introduces what stage iteration and functional interaction mean, and what the stakeholders can do with stage iteration and functional interaction during a development life cycle.

Results

The main methodologies used in this paper are case study and data analysis. Two related cases, SAIC company (System Automation and Information Corp) and PPL company (Power and Plant Limited) are respectively introduced in research design section. They are chosen for their full knowledgement of project stage iteration and functional interaction. Data is collected through documents, interviews, even projects tracking within 6 months in these two companies. All data analysis is based on the information collected in this period.

As the result of this research, four critical points for complex project changes are introduced and discussed from four views: in the following sequence: 1. How project activities were defined to accommodate potential and unexpected changes; 2.What were the challenges to project planning and programming, and why; 3. How a project organisation was formed and behaved to cope with project development and changes; 4. How project participants interacted in the decision-making process.



L. Zhang / International Journal of Project Management 31 (2013) 958–970

Table 1
Perspectives in managing stage iteration and multi-functional interaction.

Company and the centre studied	System and Service Engineering (SSE), System Automation and Information Corp (SAIC)		System Engineering Centre (SEC), Power and Plant Limited (PPL)	
Project and the stage at the time of the study	Personal Care, £200,000, fully implemented, on-site testing stage.	Safety System, £300,000, system installation stage.	Tai-Chang, £200 million, detailed design, key part sourcing. On-site construction.	Shan-Dao, £300 million, detailed design, sourcing, infrastructure completed.
Theme One: Defining project at multiple levels? Yes.	One of the four sub-systems co-ordinated by the manufacturing client from the UK. 100% design and build in house by integrated by the client.	Developed for an existing traffic system to be expanded in a European country. 97% of the project value outsourced. System integrated by the third party.	Turn-key power station project, dominated by the advising body. All parts global sourced outside Mainland China. 70% sub-contracted.	Turn-key plant consortia project, client is also key supplier. Infrastructure sub-contractors operated in China. 75% sub-contracted.
	Definitions and interpretations come from multiple levels from contractual, commercial, technical, to day-to-day operational.			
Theme Two: Challenges to the dynamic planning of project activities	Manufacturing jobs seemingly repeated despite huge subtle differences from previous projects. Cost increases by 40% (Sales inc by 11%) as the project(s) evolve, mistakes occur and new solutions emerge.	Using third party scheme to gain capability and/or networking for the new market. Cost doubles in terms of man-hours in design due to the client and sub-contractor using 'different languages'.	'Mis-interpreting' client's specs. Or suppliers specs. Cost implications. Programming (e.g. WBS) with many "holes" (i.e. 30% information is not available; correctly reflect the change and emerge of project activities and system specs). Changes and innovations impacting the project definition at contractual, technical and procurement and operational level and need to bring under the control by using Orange Books.	
Theme Three: Organising project activities and resources	The project manager works on several other projects as well as engineers. Other functions such as manufacturing and purchasing are shared.	The manager works to represent the client and co-ordinate a team from the sub-contractor.	Functional and high standard professional staff from Design, Purchase and Quality work fulltime within the project. Offices in China, Taiwan, India and UK to co-ordinate local and shared resources. Design centre, tech centre and manufacturing centre based in the UK.	
Theme Four: Decision-making in dealing with changes	Business Manager brings in the offering; Project Manager deliver them; Team Leader sources the expertise and engineers; System Manager supervise and co-ordinate internally.	Business Manager works with the third party to bring in the offering; Project Manager co-ordinate the manufacturing activities and the delivery of the system developed by the third party.	Chief Operations Officer, Project Director head the 'Capture and Execution' department which co-ordinate other five departments within the project such as 'Process & Key Parts Design', 'Deliverables-based Planning', 'Commercial Management', 'Materials and Services', and 'Document and Control'. 'Project Financing' and 'Supply Risk & Quality Management also work within the project organisation.	

As the table 1 (p.963) shows: First, defining project activities at multiple levels; second, challenges to the iteration of project management functions; third, organising project resources and activities; fourth, patterns of organising decision-making activities. In the end, all of these solutions could be regard as a preliminary conceptual model, aiming at solve problems caused by changes in complex projects.

1. Defining project activities at multiple levels. In some complex projects, project activities need to be defined and planed many times in different project stages and levels. It makes stakeholders understand the certain activities more clearly, besides, makes the tasks and duty more clearly.

2. Challenges to the iteration of project management functions. Projects are not finished in one or two days. A well-planed project should take many challenges, including the iteration of project management functions, into account.

3. Organising project resources and activities. Good management on resources and activities is helpful to project time schedule. Sometimes, enough time is always a good news to unpredictable problems.

4. Patterns of organising decision-making activities. Who and how the decessions are made in a complex project should get more alternation in every project. Not all group members have a full view of a complex project. The make up of a decision-making group is relevant closely to the solutions to project changes.

Conclusions

This paper studied two cases on exploring how solution-oriented companies develop and implement complex bespoke systems with stage iteration and functional interaction. A preliminary model, produced from literature review, shows a conceptual framework to readers. These findings and contribution would be helpful to projects changes for groups of complex projects.

Pengfei Lu

Project risk management methodology for small firms

S. Marcelino-Sádaba, A. Pérez-Ezcurdia, A. M. Echeverría Lazcano, and P. Villanueva, International Journal of Project Management, volume 32, number 2, 2014

Background

Within the European Union 66,7% of the jobs are generated by small and medium sized enterprises (SMEs). Those businesses usually generate growth based on projects and undertakings which are outside of their normal activities. The main issue is that, due to their limited resources and often less experienced stuff, industrial standards for project management are not or only partially followed. To overcome those issues and to provide an alternative to the heavy weighted common industrial project management standards such as PMBoK, PRINCE2R or ICB, Marcelino-Sádaba, Sara, et al. developed a risk management methodology especially designed for SMEs.

In their article “Project risk management methodology for small firms”, which was published in the International Journal of Project Management 32, no. 2 (February 2014), Marcelino-Sádaba, Sara, et al. discuss the differences between SMEs and large scale businesses. Their main criticism is that the common standards for software management were mainly developed for big companies and projects. On the other hand the needs of SMEs were widely ignored (p.3). But that there is a need to specify a separate methodology for SMEs has been already discussed and proven by Pérez-Ezcurdia and Marcelino-Sádaba (2012) (p.2).

Based one a lack of specific project management standards for SMEs and because risk management is one of the crucial parts of a project, the paper of Marcelino-Sádaba, Sara, et al. provides an important methodology for improving the overall quality of software products.

Results

As already mentioned SMEs often running projects outside of their daily activities to generate growth. To find out which concrete problem they are facing during the development process, Marcelino-Sádaba, Sara, et al. carried out a field study which analyzed 79 companies. Those companies included various small and medium sized businesses which were active in different business segments (p.3-4). Based on interviews and meetings three major problems were detected:

1. First of all SMEs tend to overlook the initial and final phase of a project (p.4)
2. Second, SMEs don't chose their projects wisely from a long term strategy point of view (p.4)

3. Finally, SMEs often ignore the closing face of a project in terms of lessons learned and process improvements (p.4)

The methodology suggests a generic project structure which divides it into 4 phases: "*definition, planning, execution and control, and closure and results management*" (p.5). In each phase the new methodology puts special attention on those issues mentioned above.

Definition phase

One of the major issues is that SMEs often don't choose their projects from the strategical point of view. This is a very critical and important point because all the risks which are identified in this phase are strategical risks. That means that they have a big impact on the decision if a project is going to be implemented or not. Project manager have to think especially about how profitable the project is going to be and how does it fit into the overall organizational program. If those risks are ignored the project is very likely to fail in later stages. The main tool which should be used during this phase is a checklist. Another guideline provided by the article is to think about the following six questions:

1. *Why is the project going to be implemented?*
2. *What do we want to achieve technically?*
3. *When will it be carried out?*
4. *With what resources?*
5. *How much will be spent?*
6. *How will it be executed?*

(p.6)

Planing phase

After potential strategical risks were identified and solved, it is important to create a risk plan which contains a list of all identified operational risks. For each risk at least the following six properties should be identified: "*origin, appearance phase, consequences, evaluation (likelihood and severity), response plan and responsible person*" (p. 7). While doing a risk analysis it is important to find those risks which have the biggest impact on the project outcome. Based on the fact that a risk analysis is done in each phase of the project, it is important to keep the risk plan up to date at any time. For that reason the paper proposes the *Failure Modes and Effects Analysis* (FMEA) (p.7). By using a living document in combination with only two risk properties, the FMEA tool provides a very compressed, all in one risk collection. The two needed properties are the following:

1. Risk probability: *Estimation of how likely the risk will materialize*
2. Risk gravity: *Estimation of how big the impact will be based on the project goals*

By using both properties, a *Risk Priority Index* (RPI) is calculated for each risk which determines a risk priority.

Execution and control

Only identifying and planing for risks is not enough. The best plan doesn't help to avoid problems if it is not possible to detect problematic situations. Therefor the paper suggests to identify 3 to 6 indicators which are used to monitor the risks with the highest RPI. The maximum number of indicators should not exceed 10 because it will increase the management effort and lead to too much extra work (p.8). For each indicator it is required to specify the frequency to monitor the indicator and a person who has to take care that those measurements are done. One thing which needs to kept in mind is that the measurements should have an upper and a lower bound. As long as the indicator value stays within this range no actions have to be taken. This is due to the fact that the amount of work done for a project varies from time to time which can slightly influence the indicators (p.8).

Closure and results management

For the closure phase of the project the paper discusses two important aspects. First of all it is important to identify all risks which can materialize in the final phase. Delivering the project to the customer and finalize a project also comes with various risks. Ignoring those can lead to postponing the project end date over and over again which provides a very negative picture to the customer. The second very important point is to store the lessons learned for the next projects. By doing so a lot of risk strategies can be reused and improved over time. It will increase the likelihood to succeed in future projects (p.9).

Conclusions

In conclusion it is important to mention again that the methodology presented is not only theoretical. The authors used a field study to find general problems in the risk management of SMEs and validated them against the real world. Based on the fact that small sets of tools are which are guided are better than a large range to chose from (p.4), the research paper includes a big amount of checklists. Those checklists are easy to use, easy to extend and easy to adjust to the needs of each single SME. One of the most important achievements of this new methodology is that the time spend on management is limited to around 3.77% of the overall project time. Based on the fact that SMEs have a very limited amount of resources, it is a big improvement (p.10). All in all the new methodology has proven to be effective and it solves the main issues which were found in this research. For that reason this article is a big step towards improving the overall quality of software products produced by SMEs.

Philipp Weitz

Tools used in Global Software Engineering: A systematic mapping review

J. Portillo-Rodríguez , A. Vizcaíno , M. Piattini, S. Beecham, Information and software Technology, volume 54, issue 7, pags 663-685, 2012

Background

The software teams in today's time are expanding i.e. the members of the same software teams which are working on the same project but are separated by large distances. The teams now even work when they are not located under the same roof but in different countries. Teams which are dispersed globally face different challenges and have different style of working. This whole scenario is called Global software engineering. Global software engineering teams also use different software engineering tools which help them to carry out the daily software related operations. This research paper provides a comprehensive review of software tools which provides information about which software tools are currently in use. It also classifies the tools into different types depending upon the features they have, license and other terms.

The first step in getting to know more about the software tools is to gather more knowledge about the tools that are currently being used. For this various sources were searched such as ScienceDirect, IEEE digital library, ACM digital library and Wiley InterSciemce using some special keywords. The people conducting this research had a number of results of which they had to separate out the most relevant ones. After separating results which contained useful information about GSE tools, a data extraction form was created, it contained information about each research paper such as the author, title, tools discovered and their important features etc. Information such as what type of license attributes the tool has, which feature is useful to GSE etc. was also obtained. Obtaining information about tools which can help in facing challenges posed by GSE and tools which supported any features that can help geographically separated teams such as distributed coordination. In tools, the biggest group was of research tools totalling 58 tools with commercial tools numbering 30 tools securing a distant second position.

After conducting the review, the next step included thorough research on each of the tools to obtain more detailed categorization inside them, helping companies focussing on a particular feature make decisions quickly. The research showed that 44% of the total studies done 44% focused on a single tool for a certain area, 46% of them present a number of tools for a certain area and 10% present tools for different areas.

Results

To know more about what features can support GSE, a thorough classification was done depending upon the features. They were divided into seven feature categories. The categories are namely, Subject, License, Communication, Control and Coordination, Awareness, Knowledge Management and Socio-Cultural.

- Tools such as a UML modeller will fall into the subject category.
- License category deals with tools having different license attributes such as open source, commercial etc. Research shows that 33.5% tools belong to the free license category, 43.6% belong to the research category and 22.9% belong to the commercial category.
- Communication is an important part of the team's success video conferencing, voip etc. form part of the communication tools category.
- Email notifications and visual features are in the category of Awareness related tools.
- Issue tracking systems which help the team coordinate fall into the category of Coordination and control systems. Using these managers can keep the track of the progress of the project.
- Document Management systems and blogs come in the category of knowledge management tools.

As we have already seen how tools are divided on the basis of features. The tools can also be classified according to different areas in which they work. The list of area and example of the tools working in those areas are provided below.

- *Requirement tools (SRTs):* examples include Rational Requisite pro, it also includes a document manager which can help to attach important documents.
- *Design tools (SDTs) :* Examples include Artisan studio, CAB, CAMEL etc. These tool also help users to know which user is editing which part of the design and who is currently working on the session , perfect fit for GSE environment.
- *Construction tools(SCTs):* These include a issue tracking system and a version control system on top of the design tools. Example: CollabVS , which also includes services of GitHub and different channels of communication.
- *Testing tools(STTs):* The main feature here is to execute remote execution of tests. Example: OpenSTA, testlink, webtest etc.
- *Maintenance tools(SMTs):* No tools found.
- *Configuration Management tools (SCMTs):* Issue tracking and version control form the major part of these tools. Example: CASI, Darcs,Git etc.
- *Engineering management tools(SEMTs):* Project management and tracking, risk management and measurement tools form the major part of SEMT tools. Example: ActiveCollab, ADAMS, Assembla etc.

- *Engineering process tools (SEPTs):* Features of design tools and awareness combine to form these tools. Example: GENESIS, Hobbes etc.
- *Quality tools (SQTs):*Contains features of a document manager and gives the option of writing comments. Example: AISA, HyperCode etc.
- *Miscellaneous tool issues (MTIs):* These tools are able to integrate ITS, VCS and BMS features to be used as a single tool. Example: MerlinToolChain and RepoGaurd.
- *Knowledge Management tools (KMTs):* Examples include CollabDev, DOCTOR, Google docs etc.
- *Virtual Meeting tools:* These tools help the team to organise virtual meetings with audio and video. Example: TeamSpace, Yahoo Messenger etc.
- *Socio cultural tools:* Tools like twitter fall into this category and this enables the team to share information instantly with each other.

Success of a tool for an organisation largely depends on if that tool has previously been evaluated in the same environment. A complete evaluation of a tool in an environment can only tell if the tool is fit to work in that environment in future. The paper divides the tool evaluation process into internally evaluated tools and externally evaluated tools. Internally evaluated tools are those tools which have been developed by the tool's builder himself such as Ariadne, which was verified by two experiments conducted by the builder team. Externally evaluated tools are not evaluated internally rather in practical field such as 4everedit tool was evaluated by installing it in a large scale industrial project.

Out of the study conducted, 25.8% of the tools have been evaluated in a distributed environment i.e. 33 of 132 tools. The study also suggests that 74.2% of the tools have not undergone any evaluation at all as some are still under development. It is still to be seen if these tools have any impact on the GSE.

Conclusions

The authors felt that there is no comprehensive research reviews on GSE tools. They also state that most of the tools focussed on the subjects of Virtual meeting tools (12.2%), Software engineering management tools (16%) and knowledge management tools (16%). They also observed that 77.1% of the tools discovered were either free tools or lab tools (or research tools), which also means, low percentage of commercial tools were there. Awareness, informal communication support, formal and informal communication are the most common features in the 132 GSE tools. The authors found that although there are plenty of tools which can support each phase of GSE, but there is lack of connection between those tools. Using tools from same company may provide a feature to integrate tools relating to different processes.

Rahul Arora

Project Management Software: Allocation and Scheduling Aspects

C. Chantrapornchai, D. Sawangkokrouk and T. Leadprathom, International Journal of u- and e- Service, Science and Technology, volume 6, number 3, pages 35-42, 2013

Background

In software project management, resource allocation and scheduling are very important parts in project planning phase. These two are interesting features of software project management software. Scheduling is to arrange activities in the project by allocating dates and times for the start and end of each activity. Resource allocation is the process of assigning available resources in the project to each activity in order to complete required tasks.

The Gantt chart and PERT diagram are studied by researchers and the tools are commonly used in many commercial project management project. The start time and the ending time of each task can be presented in Gantt chart. Critical path can be analysed by drawing PERT diagram.

Criteria are factors that affect the scheduling and allocation. Constraints are compulsory requirements in scheduling and allocation. These two components affect scheduling and allocation. Some criteria have to be carefully considered and prioritised for a optimised schedule or allocation.

This article aims to provide an insight of scheduling and allocation aspects of project management software which can be used in designing better software project management software.

Result

In this article, a software project management software prototype is introduced and the theory of conducting scheduling and allocation behind the prototype is discussed. The article provides an insight of scheduling and allocation aspects of project management software by comparing some project management software and tools. The prototype includes modelling in some approaches in resource allocation and scheduling and visual demonstration of results of allocation and scheduling.

To explore problem definition of the prototype, an E-R diagram (p. 38) is demonstrated considering skills, staff, staff's skills, project detail, project tasks and task information. The details of the diagram is not discussed in the article, but it is noted that the main issues are the time allocation of each task and skill that each tasks require in scheduling and allocation.

Based on the problem definition, an approach for resource allocation used in the prototype is discussed. For each task in allocation of the prototype, following process is executed:

1) Find the staff who are available in the duration of the task.

2) Find the required skills of the task and check whether the available staff is capable.

3) Staff with the least number skills are first pick to be assigned to the task because they are less likely to be capable of other tasks.

4) Staff with least pay-rate are considered from the staff selected in 3) in order to reduce project cost.

With other criteria and project's business goal, step 3 and step 4 may be updated and interchanged for better allocation.

In order to conduct project scheduling, ASAP (as soon as possible) and ALAP (as late as possible) are two algorithm considered in the prototype. In scheduling, ASAP is assumed to start the task as earliest as possible. A reminder time can be calculated by the deadline substrating deadline with the earliest task ending time. In contrast, Delaying the task as late as possible is considered in ALAP scheduling. The deadline to start the task can be analysed by ALAP scheduling. With the reminder time and the deadline to start, software engineer may be able to analyse an optimised schedule. In the prototype, both schedules can be generated to view.

With the idea of Gantt chart and PERT diagram used in the prototype, graphical interfaces are shown in Figure 4-7 (pp. 39-42) in order to demonstrate the features of the prototype project management software.

Conclusions

Scheduling and allocation are considered separately in most of the project management software. Resource allocation is more complicated due to the human factors. Many other criteria should be considered in a better approach for allocation. The common goal of scheduling and allocation is to finish the project in time with minimised project cost. It is possible to consider the scheduling and allocation at the same time for an optimised plan, but it is very complicated. When conducting the project plan, the schedule and allocation plan may be updated to keep up with the current condition as the project executes.

Many criteria are not considered in the prototype and improvements should be made for a better project management software. Future work can be focused on answer how to conduct scheduling and allocation at the same.

Ruibin Ye

Risk Management in Video Game Development Projects

M. Schmalz, A. Finn and H. Taylor, in Proceedings of the 47th Hawaii International Conference on System Science, pages 4325-4334, 2014

Background

In article Risk Management in Video Game Development Projects article M. Schmalz, A. Finn and H.Taylor reports their findings in their research about risks in video game projects. They discuss about differences between ICT projects and entertainment software projects and risks related to them. According to writers there is lot of information about ICT-projects as general, but specific research in context of games are not so common. The article itself focuses on risks and research that was made by the authors about it. The article also descripts in general about game projects and project management role in these projects. (p. 4325-4334)

In game projects project manager has usually a title called producer and his or her role in these projects are usually to coordinate activities of other team members. In game projects different kind of roles are usually involved including producer, publisher, game designer, programmers and artists. Producer communicates with external stakeholders and is often responsible to a publisher although in the era of electronic market places it is also possible to publish games without publisher. Producer's role is not well defined and especially in the smaller projects the job description may include several roles in addition to project management. Game industry is known about chaotic projects and as pointed out in the article it has been historically common that persons that had unsuitable skills for the producer's role were chosen to the job from programming or artistic background. (p. 4326)

As about the risks of video game projects they include many of the same risks that concerns ICT-projects, but new risks are also introduced because of creative nature of games. In the article entertainment software projects are compared to research projects where originality and innovation are important aspects of them. (p. 4326)

The goal of the research about risks in the video game projects was exploratory and descriptive. Researchers chose in-depth interviews as methods to full fill these goals. Interviews included eight video game producers with different backgrounds and project sizes. Producers were asked about challenges of recent projects. To analyze this data, researchers used modified Taylor's framework to code different risk source definitions. (p. 4327)

Original Taylor's definition included four risk themes called as project management, relationships, solution ambiguity and environment. Risk sources included vendor, client and third party. In this new research risk sources were named as software studio, user, partner and contractor to suit better in the environment of games. Different risk factors from the Taylor's framework were used without modifications and in the framework there are 43 risk factors in total including for example staffing, business changes and understanding of requirements. However in context of games researchers ended up to add three new risk factors that were audience match, fun factor and extent of originality. In the target audience risk factor the game may not fit for aimed audience and the game may be too hard for them for example. As games are entertainment software they need to be fun and therefore there are risks involving for reaching that goal. And because games are creative work they may bring something original to the table, but creating something entirely new may cause increased risks in that risk factor. (p. 4327-4328)

Results

Research team identified all different risk factors based on the interviews and collected data was compared to Taylor's study and risk factors matched fairly close to them. Nine risk factors were identified at least in half of the projects. These risk factors were development strategy, staffing, schedule and budget management, inadequate specification, fun factor, change management, expectations, trust and top management support. Most of the risk factors had internal source for them from the software studio itself while other sources included users and partners. (p. 4329)

Researchers found two interesting findings about these top risks that are specific to entertainment software projects in comparison to general ICT -projects. First finding was that development strategy was the highest risk occurred while in Taylor's findings it did not ranked to the top risks at all. Second important finding was the new risk called fun factor achieved place five from list of the top risks. (p. 43329-43330)

These top risks were defined as key risk factors in the video game projects. As said development strategy inside the software studio was ranked to the most common risk factor in these projects. In this risk specification for the game or game design may be well done, but technical decisions caused problems during the project. These factors involved for example wrong choices in development platforms or prioritizing the work in the project, issues in testing or failures in prototyping the product before entering to production. (p. 43330)

Second risk factor was staffing. Staffing problems included all problems with project personnel for example there were not enough staff or they possessed wrong skills. Game projects also had problems with schedule and budget. This means that projects may not be able to deliver the product in time and they may exceed the given budget, but projects were very different in terms of schedules and size of the budged. There were clear distinction for projects with publisher as in these projects producers had greater need to control budgets and schedules. Inadequate specification was also among the top risks involved. In game projects game design document is used to define specifications, but size of the documentation varied from non-existing documents to large and formal game design documents. Producers preferred lighter specifications for flexibility, but more precise documents are required dealing with external stakeholders. (p. 43330-43331)

The most unique key risk factor for video game projects was user based fun factor. This was constant subject to worry for producers. Creating fun is not just having good usability that is common to ICT-projects. Producers have to balance between technology and art to create a product that appeals to customers (p. 43331). As researchers describes in the article “game play must be smooth and intuitive and draw the player into the world constructed by the game” (p. 4331).

Risk management strategies for game projects were mostly informal that was surprise to researchers. Producers used indirect practices instead involving prototyping and agile methods that lowered risks in production. With prototypes game studio can test the game idea before entering into production and stop development before investments become too big. Distributing games through Interned also allowed making small-budget games and developing them further if the game becomes popular. (p. 4332-4333)

Conclusions

Video game projects share much in common with traditional ICT-projects, but they include certain context specific risk factor that project managers or producers should take into account. Because of entertainment and artistic nature of games it has a requirement of being fun and there is a risk that project may not achieve that goal. Entertainment software projects have also had problems with development strategies. Risk management or having proper specifications have also seen as artistic restriction among producers. However prototyping and agile methods have been natural and good solutions for avoiding risks in video game projects. (p. 4325-4334)

Sami Pulli

Different stakeholders groups and their perceptions of project success

K. Davis, International Journal of Project Management, volume 32, issue 2, pages 189-201, 2014

Background

Davis cites different authors to convey that there is a lack of research in project management and that it is considered as an immature research field. She states that there has been previous literature reviews to define project success but that project failures indicate that there is a need for further investigation. In addition, the researcher mentions that current literature recognises that stakeholders can have different views on the factors contributing to project success.

Study

Kate Davis conducted a systematic literature review to perform her study. The aim of the study was answering the three following research questions:

1. *What is the nature of project success as it is described in the literature?*
2. *Which stakeholder groups have been identified by the literature as having an interest in project success, taking a view on how to judge project success (criteria) and which factors will contribute to project success*
3. *What are the different perceptions of project success factors between different stakeholders which have been identified in the literature?*

The author used the keyword analysis provided by Web of Science to identify relevant articles and analysed their data using Bibexcel, based on the analysis provided by these tools and the author's criteria, a total of twenty nine articles were selected for the study. Finally, after reviewing the articles, a coding framework and thematic charts were created to answer questions two and three.

Research Question 1: What is the nature of project success as it is described in the literature?

In order to answer research question one, Kate Davis uses citations to multiple articles from her literature review to indicate the key characteristics of project success analysis during the decades since 1970 until present time. By highlighting these characteristics, the author gives an overview of the evolution of project success analysis during time. Some of the key aspects highlighted by the author are:

1970s: Davis marks how literature during this period focused on time, cost and quality, also known as the 'iron triangle'. Additionally, the author indicates that the project success was only evaluated during the implementation stage and how it was only based on the point of view of the project manager. She also highlights how project managers were more interested in the technical parts of the project instead of the communication with customers.

1980s - 1990s: The author introduces this decade as a change in the point of view, now concentrating also in the aspects of the project relation to the client organisation. Davis also points out that the planning phase was not taken into account when evaluating success and indirectly involved stakeholders were also excluded from analysis. In addition, she indicates how critical success factors were produced during this decade but they were crafted intuitively. Furthermore, Davis marks that success studies where only performed once during the project and that the definitions of stakeholders during this era were vague.

1990s - 2000s: The author indicates that during this time CSF (Critical Success Factors) frameworks were developed and that the importance of internal and external stakeholders in project success was recognised. Finally, the author provides an insight, asserting that 'success factors were being reproduced and that there was a lack of new factors being created' and identifies the need for creating updated CSFs instead of just testing the existing ones.

21st century: The author marks that this period relates project success on the short-term goals occurring in the project lifecycle instead of those long-term goals of the organisation. Additionally, she refers to studies were it is stated that successful projects have more communication between the owner and project management than unsuccessful projects.

Davis also indicates that the evaluation of project success between different stakeholders is not commonly performed and the importance of a longitudinal evaluation of all stakeholders during different project phases. The author describes a gap in these studies, she states that the stakeholder group identification is not explicit enough since it fails to identify certain groups. Therefore, Davis suggests that 'four groups (board, programme director, portfolio director and other organisational involvement) need to be defined as being included in another group or additional groups created as they are involved in the project process'.

Finally, the author notes that during this century 'there is a focus towards stakeholder satisfaction and a move towards examining the project owner's perception of success' but she also points out that the majority of studies focus on the project manager's view of success and not on other internal/external stakeholders.

Research Question 2: Which stakeholder groups have been identified by the literature as having an interest in project success, taking a view on how to judge project success (criteria) and which factors will contribute to project success.

In the interest of this question analysis, Davis elaborated a table (i.e. Table 2 in the original paper) in which she counted the frequency with which each stakeholder was mentioned in the twenty nine articles she evaluated in her study.

In the analysis of the table, Davis finds notes that the project manager is the most referenced stakeholder and she asserts that ‘the perception of project success of certain stakeholder groups, but in the majority this was not tested empirically’ . The author determined that it was common to empirically study the project manager, client and user; she also identified that there were more references of stakeholders involved in the project (e.g. project manager, project team, client, and such other people involved) and less references to those indirectly involved (e.g. director, engineer, executive, owner, project executive, project leader among other stakeholders).

The author mentions that there is a limited amount of studies focused on the senior management stakeholder group (e.g. top management, owners and company director) , even though it is stated in many studies that top management is very important to project success. Davis asserts that ‘the more senior the role in an organisation, the less research has been undertaken’ and describes it as ‘a gap in the literature’.

Research Question 3: What are the different perceptions of project success factors between different stakeholders which have been identified in the literature?

In pursuance of answering this question, the author divides the stakeholders into different groups, and once again uses citations from several studies to summarise the factors that different stakeholders consider relevant to project success. The different stakeholder perceptions are the following:

The project manager perception of success: The author identifies that time, cost, quality (the ‘iron triangle’) and stakeholder satisfaction were the most important factors for project managers.

The client/end user/customer/consumer perception of success: Davis points out that stakeholder satisfaction and communication were the two factors that clients mostly mentioned as relevant to project success. However, end users perceived quality (described as meeting customers’ needs) as the most important factor in the studies reviewed.

Project team perception of success: Level of collaboration within the project was considered the most important factor for project success among team members. This coincides with the user/customer perception whereas the owners only recognise the need for collaboration. The author states that ‘this highlights the lack of collaboration between stakeholder groups when defining project success and could account for different perceptions of what constitutes success between groups’.

Senior management-sponsor, owner and executive perception of success: In this group only ‘identification of objectives’ was a recurring factor during the review. In addition, there were no coincident factors in the sponsor or owner stakeholder groups. Davis observes the need to conduct an empirical study to evaluate senior management perception of success.

Davis also produced a table (Table 3 in the original paper) linking the different success factors and their importance to different stakeholder groups. She identifies communication as the main success factor according to five different stakeholder groups (i.e. project manager, client, owner, user and project team). The second factor in common was setting a meeting and schedule (mentioned by four stakeholder groups) and some other factors related to satisfaction and cost are the in the third place as the most frequent among stakeholders.

In the comparison of different perceptions, Davis finds that the client and user groups where the ones with more success factors in common and she concludes that this is due to the overlap in the user and client definitions in the literature. She asserts that there are four factors in common between project manager and user/client (communication, time, stakeholder satisfaction and cost/budget) but there were less factors in common between project manager and sponsor/owner, which the author describes as a possible explanation for the project manager need for 'top management support'.

The results also showed that the project manager and team as well as project team and user/client have fewer factors in common. The author describes these findings as surprising since 'it could be assumed that these would be the closest groups, as the project manager would inform the project team of the success factors and these would be filtered to the user/client'.

Davis finds concerning that some groups have no success factors in common, all those groups are related to the senior management level, she suggests that there is need for an evaluation of the three different levels (i.e. senior management, project core team and project recipient), why their perception differs and if it affects the project success.

Conclusions

Davis found that in spite of the current focus on stakeholders and the inclusion of the owner/sponsor and senior management in project success, there are rarely evaluations across all stakeholder groups because studies focus on the project managers perception. In addition, she points out that the studies indicate that the current success factors are viewed as adequate and there is no concern of conducting research on new success factors.

The author observed a consensus in literature about time, cost and quality as being important factors when evaluating a project success. Additionally, she identified the need for defining the stakeholders roles and responsibilities and the inclusion of some other stakeholders into the picture (e.g board, programme director, business departments, and such others).

Davis found support in the literature about the importance of different stakeholders satisfaction, however, the perception of project success for many stakeholder groups was not found. In addition, she noted that 'the more senior a role in an organisation, the less research has been undertaken' and suggested the conduction of her own empirical work into understanding the different stakeholder groups and their perception on project success (question three in her research).

During the work conducted by the researcher in this study, the main issue found was that there were no success factors in common between some stakeholder groups and that all those groups were related to the top management level. The author concludes that this reveals the discontinuity between the senior management, project core team and project recipient.

Finally, the author proposes future research to replicate her study and confirm the validity of its findings. She also suggests that a study for clarification of the stakeholder groups , and the success factors perceived as most important by these groups, in order to convey a deeper understanding of the different perceptions of project success

Silvia Rubio

Estimating Software Testing Complexity

J. Ferrer, F. Chicano and E. Alba, Information and Software Technology, volume 55, issue 12, 2013

Background:

This paper introduces a new measure of the difficulty of a program, and then analysis the relationship between the complexity measure and the code coverage.

The motivation of predicting the difficulty of a program is that we can know how difficult to test this program or how difficult the computer can generate test cases automatically. For example, if a program is very complex, containing a lot of code block, we need to have more test cases to give us confidence that no error is there. The benefit of having this measure is that we can test our programs better then solve the errors in early stage. The paper also mentions "In addition to this correlation between complexity and errors, a connection has been found between complexity and difficulty to understand software." (p. 2127)

These measures can be classified into two groups, one is dynamic and the other is static. The dynamic measures require the execution of the program, and static measures do not need. The paper introduces some main measures, for example "Lines of Code (LOC)", "Halstead's Complexity (HD)" and "McCabe's Cyclomatic Complexity (MC)". They are static measures. However, the weakness of HD and MC is that they put no notice to nesting degree, which is very important because it can increase the program's computation massively. So the paper proposes a dynamic measure called Branch Coverage Expectation (BCE).

Results:

Branch Coverage is the percentage of branches of the program that are traversed. Branch Coverage Expectation is based on a Markov model. The Markov model used here is built from the Control Flow Graph (CFG) of the program. Firstly, we define a basic block (BB) is a portion of the code that is executed sequentially with no interruption. (p. 2128) Then we define E[BBi] as the frequency of appearance of basic block *i* in one time of execution of the program (a value from 0 – 1).

E[BBi] = Pi *i* / Pi *1* , where Pi i is the stationary probability of BBi.

If we have an entrance of the program which is BB1, then E[BB1] is always 1 because BB1 will appear definitely in the execution of the program. Then we define E[BBi, BBj], the expectation of traversing a branch via i to j.

E[BBi, BBj] = E[BBi] * Pij, where the Pij is the probability to traverse to BBj from previous block BBi.

Finally the Branch Coverage Expectation (BCE) is defined as the average of the values E[BBi, BBj]. This value should be lower than 1/2. The paper explains the meaning of

BCE -- “If a program has a low value of BCE then a random test case generator is supposed to require a large number of test cases to obtain full branch coverage.” (p. 2128) So what we can tell from BCE is that we can provide the number of test cases that should be generated to obtain a good coverage. And from BCE, “we can create a theoretical prediction of the evolution of the coverage depending on the number of generated test cases.” (p. 2129)

Then they did some experiments to see the effect of BCE and for validation purpose. Validation is needed for new measures because we need to check if the measures represent the attributes accurately. The software measurement validation can be two kinds, one is theoretical validation and the other is empirical validation. Empirical validation of BCE is required so the authors use a tool to generate evolutionary test cases. In this paper they use two Evolutionary Algorithms (EAs) as the optimization algorithm of the test case generator: evolutionary strategy (ES) and genetic algorithm (GA). They also developed a generator to generate programs automatically. These programs can be used as real analysis. Then they also involve some real programs for experiments. Then the paper analyses the relationship between the static measures and coverage, and which of them is better to estimate the difficulty and complexity for a computer to generate test cases. The paper also mentions another use of BCE “given a number of test cases x, we can compute the number of branches that would be theoretically traversed if the tester execute x random test cases” (p. 2135)

Conclusions:

The Branch Coverage Expectation’s theoretical background and foundation is a Markov model. The results in their experiments prove that it is more correlated with branch coverage compared to those static measures (in our background section). The results also show that the prediction of the BCE measure is similar to the real execution of the test case generator. So BCE is a more accurate way to predict the complexity of code. With BCE, we can estimate the difficulty of testing a block of code more accurately, and know better about how many tests we need for this code to cover all situations. After analyzing the static features and the complexity measures, Branch Coverage Expectation is most correlated with the branch coverage of code.

Chenlu Wang

Evidence-Based Decision Making in Lean Software Project Management

B. Fitzgerald, M. Musiał and K.-J. Stol, in proceedings of the ICSE'14, 2014

Background

Software development is undergoing a change from so called handwork, small companies, to a major industry in which big companies do the software. The decisions done are based on gut feel indstead of investigating the results and data that can be seen.

To illustrate the change, think of a carpenter. He does all his work probably at home in a shed. All of the work is handmade, each an individual. However these days carpenters work in big companies and the products are done as a conveyer belt work to ensure that the quality stays the same. Similar kind of change can be seen in software development and it is essential that the changes are noticed in software project management. Evidence-based decision making is a way to ensure that the quality of the software stays good.

Results

The article introduces a case study that has been made using the Erlang-C model. The model together with the evidence-based decision making helped the project overcome the bottleneck situation. The bottleneck situation being a situation when all of the workload gets stuck into one place because of a delay/problem/etc. that affects all of the components in the project. The evidence-based decision making together with Erlang-C model helped the project managers to overcome this kind of bottleneck. The two also prevented the project managers not to get too overloaded with the workload of the project.

The evidence-based decision making together with the Erlang-C model helped the project team meet up with the deadlines and the time estimate of the project. Using these two together the project avoided delays and other time related problems. A way to meet up with the time related issues is to either increase the staff quality (increase the amount of staff members/educate the members better/etc.), to improve the time estimates or use the two together.

Conclusions

Evidence-based decision making is proved to be an effective way to manage a project. When investigating the current data, more accurate results and decisions can be made. Evidence-based decisions conlude to better results when comparing decisions made based for example to an expert opinion. When the data that is available is used as a base to make decisions, the projects succeed better. Every project is different and therefor it is important for the managers to think what kind of an approach should be used to project management.

Eventhough only a little research and literature has been done about evidence-based decision making and lean project management, it has been proved that it is a higly successful way of project management. The Kanban way of project management has been proved to be a good way of management. In this case study it has been very successful and other results indicate that way too. It has been an issue to use mathemathical parameters in project management but the Kanban way is a sort of safe way to use them. A way to introduce the evidence-based decision making in software project management to the public is to observe and document projects that have been using it. Both good and bad results should be introduced to give the pros and cons of this way of managing a project.

Senja Ampuja

Barriers to effective configuration management application in a project context: An empirical investigation

U. Ali and C. Kidd, International Journal of Project Management, volume 32, issue 3, pages 508-518, 2014

Background

Configuration Management (CM) is an integral activity of project management. It assists professionals in identifying issues, tracking progress, controlling and managing the changes in products or systems throughout the life cycle. However the authors of this publication believe that many organizations fail to grasp the true significance of CM, thereby they end up either ignoring it altogether or implementing it in a disorganized manner. Unlike some project management activities like quality and knowledge management, there is very limited literature available in the field of configuration management which is particularly evident in issues such as challenges and problems faced in achieving effective configuration management system. This journal publication highlights the barriers to the successful implementation of configuration management, with focus to the aerospace and defense industries.

Results

The study identifies the obstacles in effective planning and execution of configuration management system in organizations. The authors have used interviews and questionnaires to collect the necessary data. This method was divided in to three pahses; the first phase, which was an open-ended survey, was conducted to identify the most prominent barriers in implementing CM. During the second phase, seven semi-structured interviews were held with various CM professionals. The data collected was further analyzed in detail and nineteen barriers were short listed for the final review. Another questionnaire survey was used in the third phase to verify and validate the final results.

The shortlisted nineteen barriers were then rated on the basis of their mean values, using the descriptive statistical approach. Furthermore, the barriers were categorized in to three groups on the basis on their apparent association, using the principal component analysis technique, and then verified by Varimax rotation. Based on participants' gender, qualification, CM experience and organization structure, inferential statistics was used to analyze the participant's opinion on the three groups of CM barriers. The results revealed that, employees in the aerospace sector consider these obstacles as the main reason for ineffective CM process.

Group 1: Managerial and Organizational barriers

1.1. B1.Lack of top management support

The top management does not realize the importance of the CM process and often ignores it .This is the most prominent obstacle to the implementation of the CM system, and the reason for many other obstacles.

1.2. B2.Lack of centralized body for the governance of CM

Organizations do not follow the standard CM policies & procedures, and create their own methods of CM practices. This leads to the inconsistent behavior of the CM processes across organizations.

1.3. B3.Lack of CM training across organizations

Organizations do not include Configuration Management in their key training programs for the employees.

1.4. B4.Lack of authority to implement CM principles/policies

CM managers are not fully independent in making decisions, and have to comply with the project manager judgments. This results in the failure to implement CM process effectively, and hence affects the quality of the end product.

1.5. B5.Implementation costs outweigh CM benefits

The lack of knowledge and training in the CM field, leads to organizations underestimating the benefits of CM processes. They consider it to be an over budgeted process, and therefore do not put their efforts and money in this field.

1.6. B6.Lack of recognition and underestimating the importance of CM at every level of the organization

CM is often neglected at all the levels within an organization. It is not acknowledged and accepted as an essential activity for a project.

1.7. B7.Lack of career progression for CM professionals

CM is perceived as a profession with limited opportunity to progress, therefore it fails to attract highly skilled professionals.

2. Group 2:Planning and Process barriers

2.1. B8.Poorly defined CM requirements and process

The available CM standards and procedures are not well defined. Organizations find it difficult to understand and implement the process completely.

2.2. B9.Lack of maintaining consistency in CM practices across projects

Organizations do not follow the standard CM processes, and customize them according to the project in question, leading to ambiguities and inconsistencies in CM practices across the projects.

2.3. B10.Lack of flexibility in CM process

CM process is considered to be quiet rigid, offering less chances of diversion. It is difficult to implement the process successfully, especially in case of complicated projects.

2.4. B11.Outdated CM process

The CM process has not been restructured for a long time, therefore it may not necessarily conform to the latest technologies; thus it could be rather difficult to adhere it to certain project management activities.

2.5. B12.Lack of current CM plans

CM plans are managed poorly; they are not updated and followed regularly, resulting in the mismanagement of CM activities at different levels of a project.

2.6. B13.Lack of CM process across the lifecycle

CM process is not incorporated in all the stages of the project, making it difficult to implement it effectively to the whole project.

3. Group 3:Implementation barriers

3.1. B14.Lack of CM awareness in customer world

The customers do not recognize the importance of CM processes in managing their product quality, therefore without explicit guidance; customer may ignore crucial elements of the project, eroding quality of their products.

3.2. B15.Lack of effective communication

Another major hindrance is limited or less communication, resulting in many unrequired changes throughout the life cycle of a project.

3.3. B16.Lack of effective CM tools

The available CM tools are designed and developed by the people, who are not well aware of CM process. As a result, most of these tools are either complicated or do not fulfill CM requirements.

3.4. B17.Lack of resources

Organizations do not invest in the CM field as compared to other project activities, leading to the limited availability of trained staff.

3.5. B18.Lack of support from stakeholders

The CM process does not get enough support from stakeholders, making it harder to implement successfully.

3.6. B19.Extreme project pressures

The pressure to meet project deadlines causes the project managers to tailor the CM process, ignoring some necessary activities of effective implementation.

Conclusions

This research has identified, prioritized and categorized the most common barriers to the implementation of the Configuration Management process. It provides a comprehensive overview of the general mistakes and misconceptions within the top management in developing configuration management practices, stating it the main cause of many other obstacles. Organizations can refer to this study as a guideline (i.e as a Do's & Don't of CM process management) to plan and manage the CM process more efficiently by acknowledging the problem areas in early phases of a project. However, this study has been conducted for the department of defense and aerospace and therefore may not explicitly apply to the commercial sector.

Faiza Ahmed

.

Standards and Excellence in Project Management – In Who Do We Trust?

Nino Grau, Procedia - Social and Behavioral Sciences, volume 74, pages 10-20, 2013

Johdanto

Projektin laadun varmistaminen on tärkeä asia projektin johtamisen näkökulmasta. Laadun varmistamiseen liittyen on tärkeää, että yrityksellä on käytössä jokin standardi, jonka mukaan projektit johdetaan. Projektinhallintaan liittyviä standardeja on useita ja standardeja on tehty eri näkökulmista. Tästä syystä onkin tärkeää ymmärtää, mitä standardeja on olemassa ja mihin käyttötarkoitukseen eri standardit on tehty. Kuusi vuotta kehitetty ISO 21500-standardi julkaistiin vuoden 2012 lopulla ja sen tavoitteena on olla projektinhallintaan ja projektisalkunhallintaan liittyvä pohjastandardi. Pohjastandardi tarkoittaa sitä, että standardi on tehty yleiseksi, jolloin se ei itsessään anna selkeitä ohjeistuksia projektien johtamiseen, vaan se kuvaa yleisiä määritelmiä ja vaatimuksia projektien hallintaan.

Laadunvarmistuksen lisäksi usein pohditaan, miten projektien onnistumista voidaan mitata. Projekteihin liittyvän onnistumisen mittaamiseen tueksi on 2000-luvulla kehitetty IPMA Project Excellence Model (PEM), jonka avulla voidaan helposti arvioida eri kriteerien perusteella projektin onnistuminen. Tämän mallin avulla voidaan yhdessä asiakkaan kanssa todeta onnistuiko projekti lopulta, vai ei.

Tulokset

Artikkelin on kirjoittanut IPMA:n, eli kansainvälisen projektinhallinnan järjestön (International Project Management Association), standardeista ja palkinnoista vastaava johtaja Nino Grau. Hän esittelee artikkelissaan ISO 21500 standardin ja IPMA:n projektinhallinnan mittaamiseen tehdyn mallin (PEM) yleiset periaatteet. Hän myös pohtii eri standardien lähtökohtia ja miten ISO 21500 standardi liittyy jo olemassa oleviin standardeihin.

Projektinhallinnan standardit voidaan jakaa neljään eri ryhmään. Ryhmät ovat de facto, eli käytännöiksi muodostuneet standardit, erityiseen tarkoitukseen kehityt standardit, de jure, eli viralliset standardit ja elinkaarensa lopussa olevat standardit. Yritykset joutuvat miettimään, mitkä standardit he ottavat yrityksen toiminnan tueksi ja yrityksen projektinhallinnan mallit ohjaavat yrityksiä valitsemaan tietyt standardit. Esimerkiksi scrum-mallille on olemassa oma erityinen standardi ja sen valitsevat yritykset, jotka toteuttavat scrum-mallisia projekteja.

Yritykset voivat tehdä myös erinäisistä syistä omia yrityspohjaisia standardeja, eivätkä käytä yleisiä ja tunnettuja standardeja. On kuitenkin yrityksen tulevaisuuden kannalta tärkeää, että yritys investoi tunnetun standardin sisäänajoon ja koulutukseen, jotta projekteja hoidetaan ammattitaitoisesti, laadukkaasti ja yleisesti hyväksyttyjen normien mukaisesti.

Vuoden 2012 lopulla julkaistu ISO 21500-standardin tavoitteena on olla jatkossa projektinhallinnan ja projektisalkunhallinnan pohjastandardi. Standardia on kehitetty kuusi vuotta ja sen kehittämisessä on ollut mukana useita kansainvälisiä yrityksiä ja organisaatioita. Standardi kuuluu de jure, eli virallisten standardien ryhmään ja siinä kuvataan laajasti projektinhallinnan terminologiaa, projektinhallinnan ja projektisalkunhallinnan yksityiskohtia ja miten nämä ovat suhteessa keskenään. Standardissa käydään läpi myös projektin eri sidosryhmiä ja sidosryhmien välisiä suhteita toisiinsa.

Projektin onnistumisen mittaamiseen tehty PEM-malli on 2000-luvulla kehitetty ja se on todettu helpoksi sekä arvioijan että hakijan näkökulmasta. PEM:ssä mitataan sekä projektin hallinnan onnistumista, että projektin lopputuloksia. Projekti voi saada kaiken kaikkiaan 1000 pistettä ja pisteet annetaan yhdeksän eri kriteerin perusteella. PEM:n avulla toimittaja voi siis yhdessä asiakkaan kanssa todeta, miten projekti lopulta onnistui. PEM:iä käytetään myös yrityksen projektinhallinnan kehittämiseen, koska PEM:n kriteerit ovat julkisesti saatavilla ja kriteerien avulla saadaan selville projektinhallinnan parhaita käytäntöjä. PEM:n kriiteerit on lueteltu tarkemmin julkaisun sivulla 18.

Johtopäätökset

Projektinhallinnan standardointi on hajautunut laajasti. Standardeja on tehty useista eri näkökulmista ja yritysten on ollut vaikea valita heille sopivaa standardia. ISO 21500-standardi tuo yrityksille hyvän vaihtoehdon projektinhallinnan ja projektisalkunhallinnan pohjastandardiksi, mutta tämäkään ei tuo vielä merkittävää helpotusta yrityksille. Yritysten pitää kuitenkin valita myös jokin toinen standardi ISO 21500-standardin päälle, sillä ISO 21500 antaa vain yleisen näkökulman projektien johtamiseen. Yrityksen pitää tunnistaa omat tarpeensa ja yrityksen pitää myös tietää, mitä eri standardivaihtoehtoja on ja mitä standardin jalkauttaminen yrityksen toimintamalleihin tarkoittaa. Projektin onnistumisen mittaamiseen tehty PEM-malli antaa puolestaan yrityksille selkeän mittaamistyökalun, jonka avulla toimittaja ja asiakas pääsevät yhteiseen näkemykseen projektin todellisesta onnistumisasteesta. PEM-mallin avulla yritykset pystyvät myös kehittämään omia projektinhallinnan mallejaan, koska PEM:n avulla saadaan selville myös mitä odotusarvoja onnistuneille projekteille annetaan.

Ari Varpenius

Perceived Causes of Software Project Failures - An Analysis of their Relationship

T. O. A. Lehtinen, M. Mäntylä, J. Vanhanen, J. Itkonen, and C. Lassenius, Information and Software Technology, volume 56, number 6, pages 623-643, 2014

Background

The commonness in project failure has led to the development in Software engineering discipline and Software Process Improvement (SPI) concept; analysis of the causes of failures is important in developing effective and feasible software process improvement ideas. Prior studies indicate that software project failures are commonly caused by People, Methods, Tasks and Environment; and the causes of failure are spread over the software project process areas, and they are interconnected. Thus understanding the causal relationships between the causes become important. However, the prior studies have been mainly focus on identifying the causes for failures, and lack of the analysis of their relationships.

In this study, the authors collect data from four software product companies, focus on the perceived causes of software project failure and analysis the relationships between the causes. The research is aimed to take a step over prior studies towards building a causal model of software project failure.

Results

In this research, the causes of software failures are classified by two dimensions: process areas and cause types. The research objectives are:

- identify process areas and cause types of common causes in software project failures;

- reveal causal relationships and interconnections between process areas;

- evaluate the importance and feasibility of process of the causes for failure prevention and process improvement.

The overall research approach is a multiple case study in four software product companies. The research includes data collection and data analysis phases.

In the data collection phase, ARCA root cause analysis methods (p.626) were used, which aimed to identify the perceived causes of software project failures. The method includes four steps: problem detection, root cause detection, corruptive action innovation, and documentation of the results.

In problem detection step, focus group sessions were carried out with companies' key representatives to identify target problems; then a preliminary cause collection and a causal analysis workshop were carried out in root cause detection step with selected participates to identify the causes of the selected failure. In addition, before and after

ARCA method, interviews and questionnaires were used to validate the findings. The validation shows that the causes underlined by the key representatives were able to be detected and extended using ARCA method, furthermore, the participants and key representatives perceived the causes detected with the method were mainly correct and accurate.

In the data analysis phase, a detailed classification system was utilized, which was developed by the authors based on prior works and literatures. The classification system includes four dimensions for each cause: process area, type, interconnectedness, and feasibility for process improvement (p. 627).

The process areas describe where the cause occurs, and are categorized as: Management (MA), Sales & requirements (S&R), Implementation (IM), Software Testing (ST), Release & Development (PD), and Unknown (UN) for those causes cannot be categorized into above process areas.

The cause types describe what the cause of the failure is. They are characterized on a general level of People (P), Tasks (T), Methods (M) and Environment (E) and the extensions of sub-types for each level with more details.

The causes that interconnected the process areas are defined as the bridge causes in this research. Qualitative methods were used in analysis of the dimension of interconnectedness.

During the causes classification process, the causes were also quantitatively studied during the causal workshop for the detected causes, after the workshop for attendees to propose causes for improvement and for key representatives to select the causes feasible for process improvement.

The results for each company's case were presented separately and followed by a cross case analysis in the end. The pseudonyms indicating the main cause of failures selected for analysis have been used in replacing the companies' names in the results. The four company cases analyzed are: Case defects, Case quality, Case complicated and Case isolated (p.p 629-637).

The results indicate that the distributions of causes in process areas are case dependent, however, each studied cases were commonly influenced by insufficient management and by problems of software testing and implementation; the type of the causes of failures in the study were equally distributed.

The study of interconnectedness of causes shows that the common causes of failure are related and interconnected through the bridge causes, which suggest that more attention should be given to analyzing the causal relationships between the causes, and feasible targets for the causes in the interconnected process areas should also be considered in the process improvement plan.

Conclusions

The results confirm the claim in prior studies that a software project failure can not be concluded into a single cause, instead it is a result of several causes.

The studies with four industrial companies cases indicated the common software project failures causes distributed equally in cause types as people, tasks, methods and project environment and they are interconnected. Through the case study, some of the common bridge causes for the failures which interconnected different process areas of management, sales & requirements, implementation, software testing, and release & deployment were identified. Furthermore, the studies indicate that software testing may play a central role in the software project failure. The paper highlights the importance in analyzing the causal relationships between the causes of failures and consider feasible improvement for the causes in the interconnected process areas as well as fixing in the weakest process areas.

The study also suggests a case specific analysis of a project failure, as the causes of failures and their causal relationships varies with the case context.

Chen Jie

Implementation critical success factors (CSFs) for ERP: Do they contribute to implementation success and post-implementation performance?

J. Ram, D. Corkindale and M-L. Wu, International Journal of Production Economics, volume 144, issue 1, pages 157-174, 2013

Background

ERP implementation projects keep failing at a high rate despite the fact they have been implemented for decades. Several ERP implementation critical success factors (CSFs) have been identified in research. However, there is a gap in the knowledge about how these CSFs affect the implementation and the business performance after the actual implementation. Additionally, there are so many CFSs identified that the researchers doubt whether they actually qualify as such. The aim of this research was to gain knowledge on the relationship between some of the CSFs and the ERP implementation success as well as post-implementation organizational performance improvements. In this study a CSF is defined as an area (e.g. in an organization, project or implementation) that has to be successful for the organization to succeed. (pp. 157-158.) In the following paragraphs I will summarize the research questions, methods and main findings of the study.

Results

The researchers identified two separate but connected variables in the topic: Firstly, the immediate ERP implementation success, and secondly, an overall organizational performance improvement. These variables should be looked at independently but considering that the first could affect the latter. The research question is twofold as well: The first question is whether the investigated CSFs for ERP implementation are also critical for achieving improved organizational performance. The second question examines if the relationship between them and the organizational performance is mediated by a successful implementation. (p. 158.)

The researchers conducted a literature review that focused on the implementation phase, the post-implementation organizational performance improvements, and the influence of the proposed CSFs on the organizational performance. Firstly, it was found that a successful ERP implementation yields many operational performance improvements but the gains are usually only achieved after two or three years of the implementation. The literature review also revealed that many so called CSFs were identified in previous studies. However, the previous study results were fragmented and based on somewhat incommensurate success and performance indicators. Therefore it was important to study the relationship of CSFs to the implementation success and the possible organizational performance improvements. In addition, the study takes into account whether a successful implementation mediates the effects of some of the CSFs on organizational performance. As a result of the literature review, the effect of four central CSFs, the project manager (PM), training and education (of the staff) (TED), business process re-engineering (BPR) and system integration (SI), on ERP implementation success (IMP) and organizational performance (OP) were chose to be evaluated. (pp. 158-161.)

The researchers built a conceptual model showing the potential relationships of the chosen CSFs on IMP and OP. In total nine research hypotheses were formed, as follows: Preliminary study had revealed that PM and OP could be related. Therefore, the researchers decided that this should be more thoroughly investigated in an empirical manner. As a result, the hypothesis 1 (H1) was formulated to be: "The use of PM for ERP projects is directly and positively associated with OP". Consequently hypothesis 1a (H1a) was that: "The influence of the use of PM on OP is mediated by the achievement of IMP". (p. 161.)

The second hypothesis pair has to do with TED. The researchers saw that TED is a very important success factor both in an ERP implementation stage and post-implementation organizational performance, as it helps transfer knowledge and help users both understand the system better and gain confidence in using it. Therefore, the H2 was: "TED is directly and positively associated with OP" and H2a: "The influence of TED on OP is mediated by achieving IMP". (pp. 161-162.)

BPR is seen as a crucial factor of ERP project success in many studies: redesigning business processes (BPs) helps eliminate inefficiency and non-value-adding functions as well as implement industry best practices. Also, the previous research suggests that BPR helps improve the chances of a success in IMP and OP, as it aligns BPs with the ERP beforehand. Thus, H3 is: "Undertaking BPR is directly and positively associated with OP" and H3a: "The influence of BPR on OP is mediated by achieving IMP". (p. 162.)

The final set of these double-hypotheses has to do with system integration, SI: It is expected that the better integrated systems the better the organization is able to handle its information needs and control its processes. System integration alongside ERP implementation is therefore important, since many organizations keep utilizing their old legacy and other systems even after implementing the ERP. In an ideal situation the old extra-ERP-systems would work seamlessly together with the ERP supporting the business processes and organizational functions. Therefore, H4 is: ”SI is directly and positively associated with OP" and H4a: ”The influence of SI on OP is mediated by achieving IMP". (pp. 162-163.)

In addition to these four double-hypotheses, the researchers added one hypothesis that sort of founds all the others: The hypothesis is based on findings that an ERP implementation success precedes organizational performance improvements after the ERP project has gone live. In short, the researchers base their view on the stage thinking, which includes four stages in an ERP project: Adoption, Implementation, Use and Effects. The H5 is: ”The IMP of ERP is positively and significantly associated with OP". (p. 163.)

The research was conducted using a questionnaire utilizing several central measurement items identified in past empirical studies. The data was collected from senior managers with considerable involvement in ERP projects in Australian organizations. Prior to sending out the survey, it was evaluated by academics and ERP practitioners, which resulted in some modifications to it. In all, the revised survey yielded 209 usable responses for the analysis. The data received was validated against non-response and common method biases. It was concluded that there was no high risk of such, and therefore the actual analysis of the data took place. Also several different measurements were used to ensure that the research model met the quality criteria and that it was significantly reliable and reflective of the research questions. Suitable statistical analysis methods were used to 1) ensure that the data collected was sound, 2) ensure that the research model was valid and 3) examine the actual data. (pp. 163-164.)

The main results of the analysis are briefly summarized in the following paragraphs: TED and SI had a positive and significant effect on OP. At the same time BPR and PM were not found to have a significant direct effect on OP. It was revealed that IMP has a significant and positive effect on OP. IMP's mediating effect of CSFs on OP was studied with the following results: IMP mediates the effect of PM on OP. TED's effect on OP was only partially mediated. BPR did not have a direct or mediated effect on OP. And finally, IMP did not mediate the effect of SI on OP. (p. 166.)

According to the researchers, it is understandable that the impacts of the PM on OP are mediated by IMP, since the PM directly influences the success of the implementation rather than a long run organizational performance. In practice it is very important that formal PM methods and techniques are used in ERP projects. The organization should also pay attention to things such as monitoring the implementation schedule and costs, carefully defining the scope of the project and holding regular project status meetings. (p. 167.)

The finding that TED and OP have a direct relationship was in line with previous study. The effects are partially mediated by IMP, though. This would tell us that the benefits of TED carry further off than just the implementation phase, since it helps users to understand the system and its effect on the processes in an operational environment. Especially training programs that help users build confidence and are of adequate length and content can help build OP after IMP. (pp. 167-168.)

The result that BPR would not have a either direct or mediated effect on OP is somewhat in line with previous studies. However, there could be some other variables mediating the effect of BPR on OP than IMP. This is difficult to prove by this study, though, so it remains a speculation. Also there is some mixed information in the research field on how much BPR affects IMP, as in some context BPR has been considered even a pre-requisite for ERP implementation. It is possible that businesses are already so developed in following industry best practices that their processes are already so well in line with the ERP requirements that they do not require extensive BPR. On the other hand, it is possible that ERP products are so mature that they fit better the existing processes of organizations, which, again, leads to reduced need of BPR. In conclusion, this hypothesis would require further examination to yield any unambiguous results. (p. 168.)

The research shows that SI has a direct and significant relationship with OP but that it is not mediated by IMP nor has it significant relationship to it. Effective system integration can result in many organizational benefits, such as reduced production time and maintenance cost. Also, if integrated with vendor systems, for example, information is more readily available for the organization to utilize. This could result in competitive advantage e.g. in reducing the time to bring a new product to the market. In order to achieve effectively integrated systems and efficient operation, organizations would need to establish implementation strategies and systematic guidelines. The fact that SI and IMP do not share a significant relationship can be due to SI being an ongoing process rather than a once-off project. This is to say that some SI activities could take place only after the ERP implementation and not before. (p. 168.)

Conclusions

To conclude the findings in short, PM and TED are CSFs for IMP. SI and BPR on the other hand are not. In addition, TED and SI affect directly and significantly OP. The purpose of the study was to expose the relationship of some major CSFs on not only the ERP implementation but also the organizational performance improvements at the post-implementation stage. The study provides empirical knowledge on how these CFSs affect the implementation success and the organization in a longer run. The study pointed out that the implementation success (IMP) and organizational performance (OP) should be looked at as two separate variables, and that IMP can act as a mediator of the CSFs on OP. This knowledge should be taken into account, when planning ERP projects. Despite some limitations to the study (such as, not including the user satisfaction aspect, the respondents' subjective perception of OP, conducting the survey at a single point in time, limited number of factors included in the study as well as using a well-developed country of Australia as a source of data), the researches judge the study as valid and results reliable. Further research opportunities include the investigation of potential two-way relationships between the perceived CSFs and IMP. Furthermore, the user satisfaction aspect and additional ERP stages, such as Adoption and Use, should be analyzed. This would give important information for organizations and practitioners on how to succeed with ERP. (p. 169-170.)

Katriina Löytty

Factors that motivate software engineering teams: A four country empirical study

J.M. Vernera, M.A. Babarc, N. Cerpad, T. Halle, S. Beechamf. Journal of Systems and Software, volume 92, pages 115–127, 2014

Taustaa

Ohjelmistoprojektit epäonnistuvat usein: tulos ei vastaa vaatimuksia tai budjetti tai aikataulu ylittyvät. Ohjelmistoprojektin onnistumisen mittaaminen on vaikeaa, eikä sitä voi määritellä tarkasti. Jonkun osapuolen mielestä projekti voi olla onnistunut, kun taas toinen arvioi sen epäonnistuneeksi. Epäonnistumisen syitä on monia. Ne liittyvät usein huonoon projektin johtamiseen, suunnitteluun, vaatimusmäärittelyyn ja budjetointiin, asiakkaan epärealistisiin odotuksiin ja työntekijöiden huonoon motivoitumiseen. Eräs tapa tarkastella projektin onnistumista on ottaa huomioon sekä projektin hallinnan että projektin lopputuloksen onnistuminen. Ohjelmiston tuottajan näkökulmasta projekti on onnistunut, jos se pystytään toteuttamaan sovitussa aikataulussa, sovituin kustannuksin ja lopputulos on laadukas. Asiakkaan tai käyttäjän näkökulmasta lopputulos on onnistunut, jos se vastaa heidän tarpeisiinsa. Ohjelmistosuunnittelijan henkilökohtaisesta näkökulmasta muilla mittareilla epäonnistuneeksi määritelty projekti voi olla hyödyllinen. Projekteissa voi kokea onnistumisen tunnetta ja oppia uusia taitoja, joita voi hyödyntää tulevissa projekteissa.

Ohjelmistotuotannon historiassa motivaatio on aina ollut suurin yksittäinen tekijä, joka vaikuttaa työntekijän tuottavuuteen ja ohjelmistoprojektin onnistumiseen. On kuitenkin vaikea määritellä, mitkä ovat ne tekijät, jotka motivoivat juuri ohjelmistosuunnittelijoita pysymään työpaikassaan, tuottamaan laadukkaampia ohjelmistoja, kohentamaan työn tuottavuutta ja jakamaan osaamistaan muiden kanssa. Ihmiset ovat kaikkien ohjelmistoprojektien kriittisin osatekijä. Kuitenkin monet johtamiskäytännöt ovat hyvin rahalähtöisiä. Projekteja viedään läpi tuottavuuden nimissä siten että motivaatio ja moraali kärsivät. Aiemmissa tutkimuksessa on huomattu, että kulttuuri on eräs motivaatioon vaikuttava tekijä. Ohjelmistotuotanto on globaali ala, jolla ihmiset työskentelevät usein oman kotimaansa ulkopuolella, joten kulttuurierot ovat alalla suuressa roolissa. Tämän tutkimuksen kohteena olivat tiimin motivaatio, kulttuurin vaikutus tiimin motivaatioon sekä motivaation vaikutus työn tulokseen. Tutkimus toteutettiin neljässä maassa: Australiassa, Chilessä, USA:ssa ja Vietnamissa. Tutkimuksessa vertailtiin motivaation merkitystä eri maissa ja sen vaikutusta projektin lopputulokseen sekä vertailtiin tuloksia maiden välillä. Lisäksi tutkittiin, onko eri maiden ohjelmistosuunnittelijoiden motivaatiotekijöissä eroja.

Tulokset

Tutkimus toteutettiin eksploratiivisena tutkimuksena (exploratory research), jonka tarkoitus on kartoittaa aihealuetta, jota on aiemmin tutkittu vain vähän, tai jos ongelmaa tai aihetta on vaikea määritellä etukäteen. Tutkimustapaa käytetään, jos halutaan saada syvempi ymmärrys jostain aiheesta. Tässä tutkimuksessa aineistonkeruu toteutettiin käyttämällä kysymyslomaketta, jonka avulla kerättiin vastauksia ohjelmistokehittäjiltä neljässä eri maassa. Kysymyksiä oli alkuperäisessä kyselyssä 88. Kysymykset liittyivät seitsemään projektin onnistumiseen vaikuttavaan tekijään, joita olivat 1. projektin johtaminen (management) 2. asiakkaat ja käyttäjät (customers and users), 3. vaatimukset (requirements), 4. arviointi, aikataulut ja henkilöstö (estimation, schedule and staffing), 5. projektipäällikkö (project manager), 6. ohjelmiston kehitysprosessi (the software development process) ja 7. kehittämiseen osallistuva henkilöstö (evelopement personnel). Lisäksi vastaajilta kysyttiin henkilökohtainen mielipide projektin onnistumisesta. Taustatiedoissa kysyttiin missä maassa kyseinen kehitystyö tehtiin sekä oliko kyseessä uuden ohjelmiston kehitys (development) vai olemassa olevan ylläpito (maintenace).

Osallistujien valinnassa käytettiin harkinnanvaraista otantaa (convenience sample). Kyseisessä tavassa osallistujiksi valitaan sellaisia vastaajia, jotka on helpoin saavuttaa. Osallistujat olivat kaikki ohjelmistokehittäjiä, mutta työskentelivät eri toimialoilla, erikokoisissa organisaatioissa, erikokoisissa projektiryhmissä ja sekä organisaation sisäisissä että ulkoistetuissa projekteissa. Yhteensä vastaajia oli 312, ja dataa saatiin 333 eri projektista, joista 290 oli uuden ohjelmiston kehitysprojekteja ja loput 43 ylläpitoprojekteja. USA:n, Australian ja Chilen osalta tutkimus tehtiin kahden vuoden aikajaksolla. Kysymykset lähetettiin sähköpostitse vastaajien äidinkielelle käännettynä. Vietnamin osalta tutkimus tehtiin vuotta myöhemmin osana isompaa tutkimusta. Tällöin kysymyksiä karsittiin, ja jäljelle jätettiin ne 25 kysymystä, jotka oli aiemmassa tutkimuksessa havaittu kaikkein relevanteimmiksi. Tutkimus keskittyy näihin 25 kysymykseen.

Kerätty data oli luonteeltaan suurimmaksi osaksi ei-parametrista: järjestysasteikollista tai kategorista (kyllä/ei-vastaukset). Kerättyä dataa analysoitiin muun muassa tekemällä frekvenssianalyysia, ristiintaulukointia sekä khiin neliö-testiä, jota käytetään kun verrataan kategorisen asteikon muuttujia ennalta asetettuun hypoteesiin. Testin perusteella voidaan päätellä, onko riippuvuus tai ero tilastollisesti merkitsevä. Joidenkin muuttujien arviointiin käytettiin Mann-Whitneyn U-testiä sekä Kruskall-Wallacen testiä. Edellä mainittua käytetään kahden ja toiseksi mainittua useamman kuin kahden ryhmän vertailuun erityisesti silloin kun otoskoko on pieni ja muuttujat eivät ole normaalijakautuneita.

Ensimmäinen tutkimuskysymys liittyi projektitiimin motivaatioon ja sen vertailuun eri maiden välillä. Aineiston analyysin perusteella havaittiin, että eri maiden ohjelmistokeittäjien motivaation tasoissa oli eroa, eikä ero todennäköisesti perustu sattumaan. USA:n, Chilen ja Australian välillä ei havaittu selvää eroa, mutta vietnamilaiset ohjelmistokehittäjät ovat selvästi motivoituneempia kuin muut. 70 % vietnamilaisista oli joko hyvin tai erittäin hyvin motivoituneita.

Toinen tutkimuskysymys liittyi motivaation ja projektin lopputuloksen väliseen yhteyteen. Tuloksista tehtiin päätelmä, että USA:n, Chilen ja Australian vastausten osalta tiimin motivaatio ja projektin onnistuminen ovat merkitsevästi yhteydessä toisiinsa. Tämä kävi ilmi siten, että mitä motivoituneempi tiimi on, sitä todennäköisemmin projekti onnistuu ja mitä onnistuneempi projektin lopputulos on, sitä motivoituneemmaksi tiimin jäsenet tuntevat itsensä. Kaikissa maissa onnistuneissa projekteissa motivaation taso oli korkea.

Epäonnistuneiden projektien osalta USA:n, Chilen ja Australian tulokset olivat keskenään hyvin samankaltaisia, mutta erosivat selvästi Vietnamin tuloksista. Vietnamissa oli selvästi enemmän erittäin hyvin motivoituneita tiimejä. Muista maista poiketen Vietnamissa myös epäonnistuneeksi koettujen projektien tiimin motivaatio oli keskitasoa, kun muissa maissa epäonnistuneissa projekteissa motivaatiotaso oli keskitason alapuolella. Tulos saattaa johtua myös siitä, että Vietnamin osalta epäonnistuneita projekteja oli mukana otoksessa vain vähän, joten tilastollista merkitsevyyttä ei näin syntynyt.

Kolmas tutkimuskysymys liittyi motivaatiotekijöihin. Analyysin perusteella löydettiin viisi tekijää, jotka vaikuttavat motivaatioon kaikissa neljässä maassa: 1. projektipäällikön kommunikaatio tiimin kanssa, 2. riskien kohtaaminen ja hallinta 3. asiakkaan luottamus, 4. työskentely-ympäristö ja 5. tiimityö. Aineistosta käy ilmi että projektin tyypillä (developement vs maintenance) ei ole vaikutusta motivaatioon missään maassa.

Maiden välillä on yhtäläisyyksien lisäksi myös eroja, jotka johtuvat kulttuurisista tekijöistä. USA:ssa ohjelmistokehittäjiä motivoi eniten hyvä projektipäällikkö, jolla on selkeä visio. Amerikkalaiset arvostivat myös selkeää päämäärän asettamista ja riskienhallintaa ja avainhenkilöstön pysyvyyttä. Australialaisten tiimien tärkein motivaatiotekijä liittyi myös projektipäällikköön, mutta heille projektin ulkoiset tekijät olivat vähiten tärkeitä. Chileläisten ohjelmistokehittäjien mielestä hyvän projektipäällikön lisäksi motivaatioon vaikuttavat ohjelmistokehitykseen liittyvät tekijät, kuten kehitysmenetelmän määrittely, riskien hallinta ja henkilöstön pätevyys. Chileläiset eivät muista poiketen edellyttäneet, että hyvä projektipäällikkö olisi arvostanut alaisiaan. Vietnamilaiset motivoituivat ainoina eniten ulkoisista tekijöistä, kuten asiakastyytyväisyydestä. He olivat huomanneet että asiakkaiden tarpeiden tyydyttäminen on oleellista. Vietnamilaiset projektipäälliköt olivat vähiten kokeneita, mutta se ei vaikuttanut tiimien motivaatioon. Tämä voi johtua siitä, että vietnamilaiset projektipäälliköt ovat lähinnä teknisiä johtajia, ja monet johtamispäätökset tehdään ylemmillä tasoilla, koska useimmat projektit ovat toimeksiantoja ulkomaalaisilta yrityksiltä.

Yhteenveto

Tutkimuksen tarkoitus oli selvittää onko projektin lopputulos riippuvainen motivaatiosta, onko suhde sama kaikissa neljässä maassa ja motivoituvatko kaikkien maiden tiimit samalla tavalla. Tutkimus osoittaa, että mitä motivoituneempi projektitiimi on, sitä varmemmin projektin lopputulos arvioidaan onnistuneeksi. Toisaalta voidaan ajatella niin päin, että onnistunut lopputulos saa tiimin motivoitumaan. Tutkimuksen mukaan suhde motivaation ja lopputuloksen välillä pätee USA:ssa, Chilessä ja Australiassa, mutta Vietnamin tulos poikkeaa muista. Tämä saattaa johtua siitä, että vietnamilaiset ohjelmistokehittäjät olivat kaikkein motivoituneimpia ja epäonnistuneeksi tulkittuja projekteja oli mukana otoksessa vain vähän.

Tutkimuksessa havaittiin myös, että motivaatiotekijät ovat osittain kulttuurisidonnaisia. Kaikissa maissa pätevät tietyt yhteiset tekijät, mutta niiden lisäksi on myös paljon tekijöitä, jotka vaihtelivat maittain. Amerikkalaiset ohjelmistokehittäjät työskentelevät kilpailuhenkisessä, yksilöllisyyttä korostavassa kulttuurissa, joten ammatilliset tekijät ovat heille tärkeitä. Australialaiset tiimit motivoituvat parhaiten hyvän projektipäällikön johdolla. Chileläiset arvostavat eniten ohjelmistokehitykseen liittyviä tekijöitä. Vietnamilaiset motivoituvat ulkoisista tekijöistä, kuten siitä, että asiakkaalla on realistiset odotukset. Amerikkalaiset ja chileläiset arvostivat yksilöön liittyviä ominaisuuksia, jotka eivät olleet tärkeitä australialaisille ja vietnamilaisille.

Tutkimuksen perusteella voidaan todeta, että huolimatta siitä missä maassa projekti toteutetaan, projektipäällikön olisi tärkeää huolehtia siitä että kommunikaatio osapuolten välillä on toimivaa, riskien hallintaa pidetään tärkeänä ja tiimityön sujuvuuteen kiinnitetään huomiota. Lisäksi tärkeitä ovat ulkoiset tekijät, kuten asiakkaiden luottamus ja työympäristö, jotka eivät ole suoraan projektipäällikön hallinnassa, vaan ne täytyy huomioida ylemmillä johtamistasoilla. Ohjelmistoprojektin johtajien on hyvä ymmärtää, mitkä ovat ne tekijät, joita ohjelmistokehittäjät pitävät tärkeänä ja mitkä vaikuttavat heidän motivaatioonsa. Ymmärryksen avulla voidaan vastata ohjelmistokehittäjien tarpeisiin, mikä parhaimmillaan johtaa myös asiakkaan, käyttäjän ja heidän organisaationsa tarpeiden parempaan tyydyttämiseen.

Suvi Aho

A decision-making tool to maximize chances of meeting project commitments

T.-H. Nguyen, F. Marmiern, D. Gourc, International Journal of Production Economics, volume 142, 2013

Background

The article addresses a problem that more and more companies are facing: how to ensure better product quality, better deadlines, and lower costs in the context of market globalisation? Various methods for project management and risk management have been developed to help managers choose the most suitable strategy to maintain project profitability. T.-H. Nguyen et al. observe that few methods are able to link project planning, project management, and risk management together. Three closely related methods are discussed at length: RISKMAN, PRAM, and ARAMIS.

The RISKMAN method introduces the concept of risk reduction strategy that managers can use to reduce, eliminate, or avoid the impact of risks. Having identified and qualified the risks a manager must apply the reduction strategy at each project life cycle.

The PRAM method is an iterative process that comprises several steps including defining, focusing, identifying, structuring, clarifying, estimating etc. During identification phase each risk is associated with a corresponding reduction action creating a scenario where secondary risks might arise from these reduction actions.

The ARAMIS method was developed as an alternative to probabilistic or deterministic approaches to risk management. Building on existing study results, the method defines accident scenarios and protection barriers to stop the evolution of these scenarios. A bow-tie shape is an integral part of the ARAMIS and is used to model different steps in risk management.

While these methods provide ways to model the impact of risks on project planning, they are unsuitable for adequate analysis of risk interdependencies and repercussions on a project as a whole. Therefore, T.-H. Nguyen et al. propose a new decision-making tool, ProRisk, for better project planning, project management, and risk management.

Results

The ProRisk method takes into account process interrelations between project and risk management whereas traditional methods usually model these processes as independent. T.-H. Nguyen et al. base their work on the synchronised process of project schedule and risk management from Pingaud and Gourc (T.-H. Nguyen et al., Int. J. Production Economics 142, 2013, p. 216). Two situations are covered where the tool can be used to its full potential: in the event of a known risk a manager can use the tool to determine the best treatment strategy and the manager can notify the sales department if the contractual constraints allow for the identified risks to be integrated into project profitability when responding to an invitation to tender. The main objectives of the tool are to determine the impacts of the identified risks and treatment actions on the schedule and to help choose the best treatment strategy.

The ProRisk method relies on two main hypothesis. Firstly, the risks are integrated to the project management in respect to the deadline and cost. Secondly, the project tasks and risks must be identified at the beginning of the project. T.-H. Nguyen et al. recommend the Delphi method for eliciting task and risk related data. The hypothesis do not include the aspect of resources such as availability or skill. Key concepts of the method include the definitions of risk scenario and treatment scenario. A risk scenario corresponds to a combination of reoccurring risks while a treatment scenario corresponds to a set of treatment strategies. In turn, a treatment strategy corresponds to a set of treatment actions that can be taken to reduce or avoid a risk.

The proposed tool was tested on a case study data that was carried out by T.-H. Nguyen et al. The case study was organised to first gather data about the project scenarios without any treatment strategies applied, and then with the risk scenarios and treatment scenarios taken into consideration. In order to choose the best treatment strategy, cost and duration metrics were used for each project scenario. The data shows each project scenario including their respective level of criticity, associated risks, probability, level of impact, duration, and cost. By scrutinizing the data, the treatment strategies can be evaluated by their impact on cost and duration and ultimately a treatment strategy that stays within the acceptability zone can be selected.

The ProRisk can also provide the manager information about chances of success when defining target costs and deadlines. Analysing data with the tool provides probabilities for different risks and impacts on the project duration and costs which helps the manager to estimate the chances of staying within the budget and meeting the contractual commitments.

Conclusions

First, the existing methods were analysed for background information. Then a new method was proposed which uses the synchronised processes principle. Finally, the reliability and functionality of new method was tested on the data from a case study results.

The new method models the interdependencies between risks and thus produces more accurate results than previous related methods. The ProRisk method can be used throughout the whole project life-cycle to evaluate various project scenarios and their risks and to choose the most appropriate treatment strategy.

Juha Kaura

Software Process Models and Analysis on Failure of Software Development Projects

R. Kaur and J. Sengupta, arXiv.org, submitted on 5 Jun 2013

Background

The paper discusses various aspects of software developmental models. Software developmental models are considered as backbone of any software, because the actual software is built based on the infrastructure set by the models. Despite efforts and various techniques, some projects fail. The failure is devastating for an organization. However, it is unclear as to what defines a project as failed, many organizations have different standards and they might have a different criteria of a failed software. Even so, the paper discusses various processes models and analysis of the factors behind failure of projects and software.

The article discusses various reasons of failure and the models that are widely used in the norm of software development. The statistics shown regarding the failure percentage of software is staggering, however, there still is no means of detecting at the beginning if a project will fail or succeed.

Results

The paper mainly focuses on the software process models their importance and the disadvantages they impose on any process. Developmental models are essential part of the process, they define specific set of step to follow through which a software will take shape. Models helps us give a clear picture of what the actual software might look like at the end. Other advantages include time saving, cost effectiveness of the project and quality of the final product. Remember, chosen process model will have a drastic impact on the final result. A poorly chosen process model might see the project going over budget, over the limited time and not meeting the quality criteria set by the customer.

Technically, if a project goes beyond cost, fails to meet the quality criteria and development goes beyond target time, it fails to become profitable to the organization and hence it can be dubbed as a failed project. Some other factors that might influence project failure are listed below:

- Project team compromised
- Inability to handle varying demands from clients.
- Estimation misjudgment.
- Unclear goals.
- Change of management during development.

To avoid such mishaps, it is advised to follow a well devised plan. Such as, following a software process model. Listed below are a few process models and their impact on the software development:

Model	Advantages/Disadvantages
Waterfall model	Undoubtedly, waterfall method is one of the most popular methods, however, the inability to handle ***new requirements*** in the middle of development sets it back.
Spiral model	Spiral model is like waterfall method, however, it ads further depth of information. The requirements are checked after one complete iteration. However, they method can prove to be complicated with large software applications.
Prototyping	We are all aware of prototyping, in fact, it is the best way to present a final representation of a project. One of the issues with prototyping however is ***false expectations***. You might not be able to deliver what you prototype.
Rapid application development	RAD in a sense is an advance form of prototyping. It is best used when there is a noticeable lack of ideas. The model sadly can sometimes go ***over budget***.
Agile development	Less stress on analysis and design. Implementation begins early, very cost effective and very adaptive to changes.

These methods weren't proposed all at the same time. They evolved with the needs of day to day software development needs. This evolution shows that the evolutionary needs of the field of software engineering. The models are becoming more risk effective and customer oriented with every iteration. However, no matter which model one might chose, there is a certainty that the project might fail. Running over budget, premature termination and the reasons mentioned previously, still remain a huge concern. An overview from the CHAOS report might give a clearer picture. According to the survey conducted in 2004, quoted from the article (page 2):

- 29% of projects are successful.
- 53% projects are challenged.
- 18% projects fail to meet the criteria.

According to TATA consultancy service 2007 survey:

- 62% of IT projects fail to meet their schedule.
- 49% projects suffer from budget issues.
- 47% had miscalculated maintenance cost.
- 41% failed to generate expected revenue.
- 33% had performance issues.

It is quite bizarre that almost 80% of projects fail to reach their optimal goal. Even if you chose the right process model, do everything according to plan, uncertainty factor remains the same. There are some uncontrollable factors that effects the outcome as well:

The project might over run budget or get delayed if there requirements extracted are not what the client wanted. This is a common problem, obviously there might be some technical difference between the organization and the client. During development, it is essential to test the software and having iterative meetings with the client. This would ensure that the development is on the right track. If software does not get tested by the client, unforeseen requirements might pop up at the deployment stage, which can prove to be very costly. Last but not least, the team size. Team size matters a lot: for a small project, a team size of 10 should be the maximum limit, team size of 11-25 would fit a medium scale project and team greater than 26 should fit a large scale project. Smaller teams are more flexible and there are less chances of mishaps.

Last but not least, quality control and testing. There are the final stages a software goes through before the pipeline ends. Spending too much time and effort on non-essential requirements can lead the project to a disaster. Another factor that must always be kept in mind is uncontrolled requirements: requirements that are led by previous requirements can create a havoc, so it is essential that a room for error is measured in the beginning of the project.

Conclusions

The world of software is evolving day by day. The risk of failure remain no matter which model an organization implies. However, if an organization is aware of the factors which may or may not affect the failure rate, it becomes significantly easier to keep a project on track towards success. The failure rate of all IT projects is quite high according to recent reports, however, all of them can be minimized to great extent. Various process models were discussed in the paper, along with the flaws they have, the best way to handle all these is to plan and learn lessons from other failed projects.

Muhammad Farrukh Anwar

Performance on agile teams: Relating iteration objectives and critical decisions to project management success factors

M. L. Drury-Grogan, Information and Software Technology, pages 506-515, volume 56, issue 5, 2014

Background

Failure statistics of information technology projects indicate that projects fail often and are considered failed due to budget, schedule or quality issues (p. 506). All of these factors – time, budget and quality – are also defined as success factors in the golden triangle of project management. In her study Performance on agile teams: Relating iteration objectives and critical decisions to project management success factors, Meghann L. Drury Grogan discusses how iteration objectives and critical decisions in agile software development teams relate to the golden triangle of project management.

The motivation for the study was to find out how practitioners in software projects take success factors into account and how the success factors are represented in project objectives and decisions. Additionally, the results could explain the high project failure rates.

Results

The study covered three separate case studies, results of which were then prepared for cross-case analysis. Three agile software development teams, all of which represented the same organization and used a hybrid development model of XP and Scrum, were examined in terms of using success factors in their iterations. Team members were interviewed on iteration objectives and critical decisions related to their projects. The answers were categorized and mapped to success factors. In addition to the interviews, two iteration planning meetings and two retrospective meetings held by the teams were directly observed for notes of iteration objectives. Misinterpretations were avoided by clarifying answers and observations afterwards, when needed. All the factors were coded in two rounds to add more perspectives to the study; the first round was coded by the author and the second round by two research assistants. (pp. 509-510)

Iteration objectives discussed by agile teams were functionality, schedule, quality and team satisfaction. Functionality consisted of the sub-categories "*Develop Iteration Functionality*", "*Test Developed Functionality*" and "*Document What You Did"*, schedule consisted of "*Plan Work*" and "*Finish Work on Time*" and quality consisted of "*Ensure Product Works Pre-Release*", "*Fix Bugs*", "*Address Client Issue*", "*Review Others' Code*" and "*Ensure Client Satisfaction*". Team satisfaction was an objective only ensuring teams' overall satisfaction on the project. Budget is one of the primary success factors in project management but it was not referred to in iteration objectives by any member of the teams. The categories were prioritized in the order they are mentioned above. (p. 510)

The identified categories of critical decisions were quality, dividing work, iteration amendments and team satisfaction. Decisions on quality were related to working on functionality, e.g. deciding to improve functionality whether or not the client had demanded it. Decisions related to work division were considered critical as well, because work intelligently divided in components which the team members understand and can work on results in success. Software development teams should also be careful when making decisions on iteration amendments. Accepting amendments might cause problems in development, such as unnecessary or overlapping work or excessive pressure from the client. Team satisfaction is built up with the decisions made by the team, making the whole team participate in decision-making and feel responsible for the project. (p. 511-512)

When comparing the discussed iteration objectives and the success factors presented in the golden triangle, quality-related objectives – including all of its sub-categories – were mentioned most frequently. Schedule was the second most frequently mentioned objective and budget was given hardly any attention at all. As a result, it was considered that agile teams prioritize quality over the other factors because they want to focus on getting a project and a product properly finished rather than finishing it in a hurry and with a smaller budget and finishing with an unsatisfactory product. The other iteration objectives – functionality and team satisfaction – cannot be directly mapped to the success factors of the golden triangle. (p. 512)

Quality was also considered as the most important factor of critical decision categories. It is also highly related to the iteration objectives, as making critical decisions between the success factors often leads to poor scheduling while quality is the priority. The other categories – dividing work, iteration amendments and team satisfaction – were related to iteration work improvements and they cannot be directly mapped to the golden triangle. (p. 513)

Conclusions

The success factors discussed by agile teams in this research were quality and schedule. The agile teams interviewed and observed in this study ignored budget as a success factor but, depending on the project, it could be taken into account as well when creating iteration objectives and making critical decisions. According to the results, agile teams tend to choose quality over schedule when it comes to success factors of project management. (p. 514)

The paper introduces readers to real world agile software development cases in which objectives are created and decisions are made under certain circumstances, in certain environments and in a certain organization culture. The findings are also related to the methods used in development. Similar studies examining projects in different types of organizations, across project teams using agile methods, would be an interesting addition, as it would present new perspectives in which the balance of success factors could be different.

Juho Toivonen

Review the benefits of using Value Engineering in Information Technology Project Management

H. Tohidi, Procedia Computer Science, volume 3, number 917-924, pages 3-8, 2011

Background

- Background information

In order to satisfy the increasing IT operational projects in organizational and national levels, Value egineneering has been introduce to this area. The main function of value engineering is to eliminate or modiy anything that causes unnecessary costs, without damage to essential funtions. And the purpose of applying value engineering is to complish the projects, increase performance and reduce costs in all stages of projects operation and research.

- Terms

Value Engineering (VE):it It is an orgnized effort aimed at studying and analyzing all activities of a plan since the formation of the initial thinking to the design and implementation stages with full implementation plan to realize the lowest cost and time.

Performance Ratio : output/input,using value engineering to increase both output and input by project management, project analysis, value analysis and value management. Finally the performance ratio will increase.

Performance Index value: price performance of cost function, with the performance increase, the productivity will increase.

- Motivation

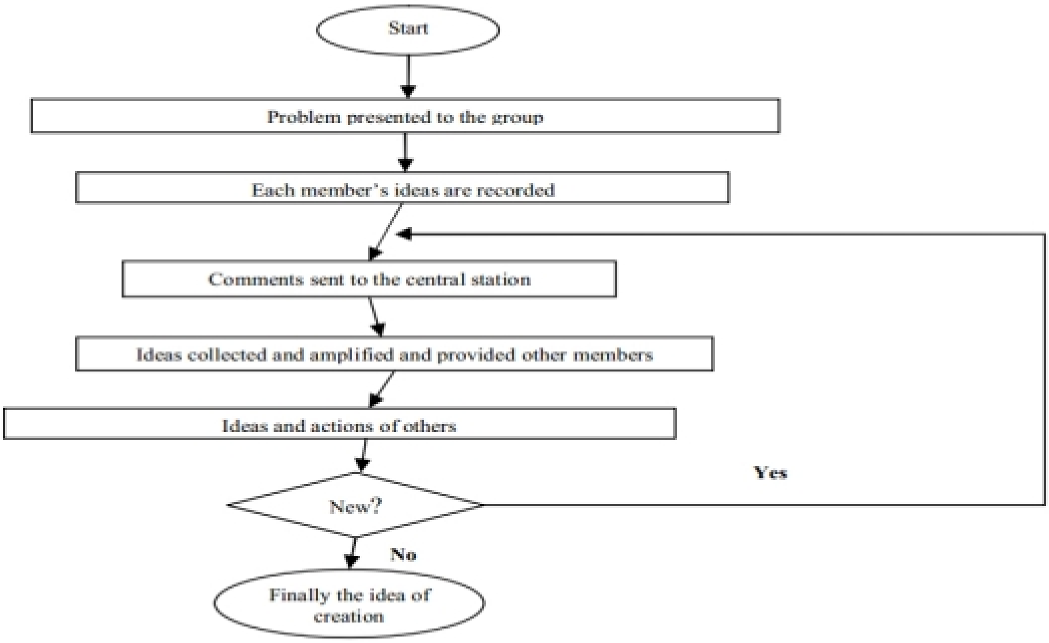
Long time ago, because of the project managemet spent less time on value engineering, the method was not regular, such as depending on past experience. Instead of the oast, now information,identification of problem areas, developed methods has been proposed.

Results

- Method

There are 11 VE methods mentioned in this article. The specification as follow:

1. Mental movement: There are three demisions in this article,they are demand(z-axis),Value(x-axis),Quantity(y-axis).And it is like a circle iteration. People could submit their ideas.-Compound and modify ideas: combine more ideas.
2. Delphitecnique: It have five stages, the flow chart below could explain this:


Start
Problem presented to the group
Each member's ideas are recorded
Comments sent to the central station
Ideas collected and amplified and provided other members
Ideas and actions of others
New?
Yes
No
Finally the idea of creation

3. Collapsed sociological analysis Technique: The main feature of this technique is it can be formed and conponents compared with the main dimensions. Every component on concerntric circles.
4. Nomnal Group Technique:it also include five stage: Group member writte thier owen decision and keep decision secretly(They can communication before decision making time).-Each member show their idea to the group until the idea is accepted.-Evaluate proposed idea-The highest score idea become the best idea.
5. A Scamper technique or a conversial ideas and questions:Substitute -> Combine -> Adapt ->Magnify -> Modify -> Rearrage -> Put to other use -> Eliminate -> Substitute.Finally the best choice will appear.
6. FAST Technique:It is a systematic guide map for the task.FAST charts(releated to each other) and work are breifly introduced and presented in Graphics.It is better for larege system.
7. Model nature: Selection of a powerful form of flows of items on laws of nature with choice of images.
8. Qulity Home:There is a house western part (quality demands and customer needs), second floor (quality characteristics), main floor (dependency matrix).gable room(relationship of designs characteristics), East (comparint result of comoetitors) .
9. DO IT technique: It is formed four English words. Define topic precisely.-Divide it into smaller problem.-Questions can be seen from different angles. The ideas should become practical solutions.

10. The illusion of creative techniques: People always think it is reality,but it is not.Something the eyes seeing is not the object theirself but the brain experience review.
11. Mandatory communication techniques: The technique of placing two different objects together and try to communication between the will help to creat direction cause theories and ultimately cause new products or adding.It has five process group:

    Early processes (Initiating Processes): This process gives the project recognition and a license is issued to it.

    Planning Process (Planning Processes): to define and refine fials ,aswell as best practices among other methods of operation in order to achiece desired project goals,committed to reaching their deals.

    Process Executive (Executive Processes):to coordinate people and other resources for profram data.

    Process Control (Controlling Processes): The prohect goals through regular monitioring and evaluation of the processes.

    Process closing (closing Processes): Accept the project of phase,implemented and give it to recongize the end of a regular point and provides specific guidance.

- Case

Combining methons propsed in a project.

Starting Phase:

1. Market review.

   make sure stakeholders,target community,catagory of market,other market

   The technique using here is socialogical anaylisis,Delphi technique,Model from nature.
2. Technique review.

   make sure thecnique means,tools,functions and operations.
3. Financial review.

   make sure operating expenses.total investment plann,financed resources

Planning Phase: Define project scope,project activities,sequencing activities.The technique using herer is FAST technique.

The Executive Phase: Administrative phase of resources needed by the committee research project is anticipated to begin and the project is required to be collected and taken according to plan.

Phase Control: Using continious issues contral the carious issues in different phases.The technique using here is Delphi technique.

Closing Phase: Review the final product production and project ecalyation.Here also use Delphi techique.

Conclusions

VE could apply to different fields. With the development of the IT field,the Ve is needed. Moreover project leader want to achieve maximum efficiency and through reduce administrative cost and increase profit. The advantage of the VE is it could analysis project cost qualitive and quantitive. And there are lots of methods support this technique.

But sometimes lots of method ofen cause technology selection problem. Even through the VE is also necessary in many project.

Yang Mengyuan

Blazing a New IT Project Management Career

K. Ziehl and D. Pecora, in Proceedings of the 36th annual ACM SIGUCCS fall conference: moving mountains, blazing trails, pages 145-148, 2008

Background

Managing an IT project can be problematic, and in an University environment it can be especially challenging. In ademic culture people do not respond to IT project management methodology well. Projects are one of, if not the most important thing to an IT organisation. They are important because of they are they are the main method for implementing large changes in systems, and for making services.

In 1995 a report submitted by Standish Group stated that only 16% of all software projects are completed within budget and time constraints, and 31% are cancelled before they are implemented (p. 145). IT project failures are often rationalized or covered up in a way that nothing is learned from the mistakes made in said failed projects.

The report aims to address reasons IT projects fail, and how the major pitfalls of IT projects can be avoided.

Results

From their own experiences, the authors recognize the ten most common reasons for the failure of IT projects. In addition to recognizing the reason for failure, the authors have suggestions for the avoidance of said issues.

"Inexperienced project managers" (p. 146). A trained and experienced manager is not necessarily a good one, but having no experience and training makes project management more difficult. Although there is no substitute for proper work experience, there is a plethora of recorded tips and knowledge available for study. Also finding a mentor who can give case-to-case advice is advisable.

"Lack of project funding" (p. 146). Lack of project funding affects the outcome of a project, and is the reason for failure in many projects. Managers usually know which tool or application they need to fulfill the project objectives, and denying funding for them is a big problem for the project. Knowing when to stop work on a project and inquire on the availability of funding is a very important ability,

A similar issue is the *"Lack of understanding of the true cost of a solution" (p. 146).* The cost of researching and implementing software tools is not acknowledged. E.g. Finding and modifying an open source tool. The solution for this issue is as simple as acknowledging the costs that accumulate (e.g. salary) when a employee uses on something.

"The same people are needed for nearly all the projects" (p. 146). If a organisation has too many projects and too little staff, the projects are bound to suffer. A solution for helping to realize you own project instead of the others would be to help prioritize it.

”Not understanding that project participation involves work” (p. 146). There is a illusion that the project manager should do all the work and be the subject matter expert, which is not the case. This illusion can be shattered early on in the project by indicating a role and set of responsibilities to each team member. Giving a person a role and set of resposibilities that fit their specialized set of skills is important to ensure the best outcome for the project. The issue is similar to the issue of *”Communication” (p. 146).* The lack of communication results in duplicate efforts, eating up recources with redundant work. Relevant people should also be kept in the loop about project steps, as not to cause confusion or bad blood.

Creeping featurism or *”scope creep” (p. 146).* Scope creep is a continual hazard of a project. In short the continuous growth of the project scope leads to uncontrollable changes. Remembering that scope creep is an expected property of a normal project progression helps to handle the issue. A project can be broken down into phases or sprints (e.g. SCRUM), and the functionality to be delivered during each phase can then be decided. This helps handle new feature requests, because in this kind of a model postponing feature implementation is natural.

”Doing things because its cool, not because it should be done” (p. 146). There is a risk that a new service is implemented because it is new and cool. Sometimes the technological infastructure or personnel support is not adressed before implementation, leading to technical issues or the lack of use of the new service. Finding and communicating to the right stakeholders reduces the risk of unwanted or technologically bad services.

”Requiring buy-in from everyone or no one” (p. 146). Requiring buy-in from everyone is a slow and arduous process. Then again, not consulting anyone when deciding on key solutions and processes is equally harmful to a project. Getting everyone involved to agree on an issue is nearly impossible, and not consulting anyone can result in a solution that is only usable to some end users.

”Making project management process too complex” (p. 146). The implementation of a new and untested project management processes, making them too complex and bureaucratic is a recipe for disaster. If the management processes are not embraced, they will become a hindrance. A simple and transparent management process is a much better path to take.

Conclusions

Although IT project management is filled with issues and various pitfalls, there is a world of solutions to be applied to said issues. Knowing why projects fail is the first step to successful project management. The anticipation of problems and preemptive solutions are a large part of project management.

Juho Ylipoti

Analysis on Enterprise's Software Project Management Based on Game Theory

NIE Gao-Hui, School of Information Management, Jiangxi University of Finance & Economics

Background

The paper introduces the concept of game theory to the software project management process. Basically, it notices that the management process of a software project could be presented as a "game" between two "players", the enterprise presenting a need and the software house, which provides the solution.

First of all, what exactly is game theory? Whenever "game" theory is mentioned, one would often think about its relevance to parlor games such as poker or bridge. This association is not too far off, as game theory is essentially an observation of social situations. Most research in game theory focuses on how groups of people interact. Therefore, in this paper, the author observes the interaction between the enterprise and the software house, and documents the "players" and "rules" of this "game".

Results

The paper starts off with a quantitative analysis on the "game" of a software project management process.

Here, we have two "players" in this game:

1. The "enterprise", which is the organization who needs the software to be built.
2. The "software house", which is the provider of said needed software.

Here are the "rules" of the game:

1. The "enterprise" player have the following goals:
    a. Start a software project based on the needs of the organization
    b. Collect requirements and define goals
2. The "software house" player have the following goals:
    a. Review requirements and goals
    b. Determine feasibility and propose cost estimates

Here are the "win/lose" factors of the game:

1. Win – the game is won when the software project is concluded successfully, with the software outcome deployed and utilized.
2. Lose – the game is lost when the project is abandoned, or if the project does not meet its goals.

Here's how to "play" the game:

1. Start the project and provide input for requirements and for defining project goals
2. Determine "income" for the software house
3. Determine "profit" for the enterprise
4. Proceed with the game until an outcome is determined.

The concept of "Nash equilibrium" also applies in this game. Nash equilibrium is the balance achieved in a game due to the fact that no player can "win" or "lose" by changing their strategy (or behavior) in the game, even after knowing the strategies of the other players. In this game, due to the need for income and profit, all players benefit from full disclosure of information and maximum contribution to the project.

Here is the equilibrium analysis of the game:

1. The software house has to consider their income in the game, and the enterprise has to consider their profits. This could result in an unequal struggle for success if:
    a. The enterprise chooses to delay the software project, or to halt it altogether, resulting in a reduced income for the software house.
    b. The software house chooses to extend the timeline of the project, resulting in an increased income for themselves, but a reduced profit for the enterprise.
2. If all players are being rational, they will choose to complete the project successfully. The successful completion of the project will reap maximum benefits for all players.

The qualitative analysis of the game, which is what is required of the players to achieve a win in the game, is for the enterprise player to have a clear goal of the project and to fully reveal any and all project requirements to all players of the game. The software house player's winning strategy would be to provide the correct solution for the project, and deliver a proper product at the end of the project.

Following the theme of game theory, the author concludes by presenting suggestions for the players to achieve "double win" for this game:

1. The enterprise player should do a rational analysis of the project requirements provided to them, as well as determine objectively if the project solution is feasible. This should be done before the start of the project implementation.
2. The software house player should do an analysis of the solution provided by the enterprise player before signing any contracts with them.
3. All players should establish and maintain formal and stable communication at all times.
4. Each player will want to conceal certain fallacies of their contribution to the project based on their respective problems of investment and return. They will want to maximize their income and profits. In order to avoid associated risks, both sides should exercise due diligence in ensuring that the information required to successfully complete the project is communicated as needed.

Conclusions

The paper makes an interesting comparison to a software project management process to the concept of playing a game. It also notes the fallacies of such a comparison by stating that it is an oversimplified presentation of a software project management process. Further work may be needed to fine-tune the intricacies of the game play to resemble more realistically the process.

Tsailing Wong

Book Review: Advanced Multi-Project Management

Gerald I. Kendall, Kathleen M. Austin, Advanced multi-project management: achieving outstanding speed and results with predictability, J. Ross Publishing, 2012

Background

In their book Advanced Multi-Project Management, authors Gerald I. Kendall and Kathleen M. Austin write about how incorrect management behavior is causing companies to waste resources and time on managing projects in a multi-project environment. Especially multi-tasking and running too many projects simultaneously is causing projects to take much longer than necessary.

Previous management styles of maximizing the use of resources has led to the state where project resources are pushed to the maximum, instead of trying to maximize the flow of the projects. This is mostly due to the fact that too many projects are being done simultaneously, due to the target of maximizing resources.

What this means is moving resources between projects and important tasks is causing the already strained resources to perform even poorer, generally resulting in more than 25% efficiency loss, up to 50% on certain tasks.

By implementing the elements listed in this book, the authors claim a 25% increase in project completion, without sacrificing budget or scope.

Results

The authors found in their studies, six elements of proper multi-project management that provide a solution to a bad situation for any type of company, running any type of projects.

The six elements are listed as:

- Project networks
- Strategic buffering
- Controlled project WIP
- Fast execution management
- Single priority system
- Consistent recovery approach

Also the as a drastic measure to cut the amount of active project work by 50% (e.g. by freezing projects) then stagger project accordingly. This has generally made it possible for even the frozen projects to be finished in time, due to the increase in project flows.

Project networks

As described in Part III (pages 103 – 175), the idea of project networks (or project plans), is to properly create a model of the project. The actual technique doesn't matter (pert, lists, gantt). Research suggest that 75% of a project value is created during the execution phase. So a project plan needs to be simple, and void of waste.

The project network should include a model of the major work needed to meet the stakeholder needs and drive some part of the organizations goals. With only a very limited detail of the planned work. Since most project live and change, too detailed planning is futile work.

The most common problems in project networks are:

- Lack of process and lack of scrutiny, leading to poorly and ineffectively made plans.
- The use of named resources, leading to impossible expectations
- Incorrect level of detail, leading to too much work, or too little plan to work with
- Not checking tasks against stakeholder needs, thus doing unnecessary or wrong things
- Not checking for dependencies, thus missing important work or relations

Because the amount of scrutiny, the authors suggest using a ready-made 10-step method to systematically create a project network, using people in your organisation most suited for this.

The selected people are important, because this type of task can be irritating to some people, thus leading to quality issues on the plans.

It is also noted that a well-made project network and project plan is an elemental piece of a projects success, and one of the most common issues on failing projects.

Strategic buffering

Described in part IV (pages 179 – 201) is the idea behind buffering; to protect the projects from variability. Variability is inevitable, and thus trying to avoid it is futile, therefore it is wiser to build measures to battle the variability. This is where buffers come into play. The actual strategic buffering is divided into planning and execution. The planning phase creates the buffers, and the execution phase acts on the amount of buffers being utilized. To be able to do project buffering, each task needs to be given two estimates for completion, the aggressive (best case) and standard estimates. Also to calculate the estimates, the maximum amount of resources should be used.

Then, on the planning phase, buffers are created at three different places.

- Project buffer, which is placed at the end of the project, to provide protection for the whole project. This buffer is sized by calculating the difference between aggressive estimates and standard estimates of all the work on the critical chain, and then dividing that amount by two.

- Feeding buffer, which buffers the critical chain tasks from lesser tasks, so that even if those non-critical-chain-tasks somehow get delayed, they will not push the critical-chain-tasks forward. The buffer size for the feeding buffer is calculated by summing the differences between the aggressive and standard estimates of every task before the critical task, and dividing that sum by 2. These buffers need to be then placed before each critical task, which is being fed by a non-critical task.

- Critical milestone buffer, this is an optional buffer that is placed before any critical project milestone (like a gate). The size of the buffer is dependent on the milestone, but should be calculated using all the critical and non-critical tasks before the milestone.

On execution phase, it is important to track the buffers on a daily basis. Buffers should be divided to 3 areas, green / yellow / red. The tracking should focus on following how much of the buffer has been used by the project (buffer is used when a task takes longer than estimated, and the buffer is regained when a task takes less time than estimated).

The project manager should act when his project reaches yellow buffer zones, and start creating a recovery plan, to be executed if the project goes onto the red zone. This plan is meant to return the buffer to the yellow zone.

Controlled project WIP (work in progress)

The part II, chapter 7 (pages 55-63) has information about Controlled project WIP, the excessive amount of work projects are causing in an organization that is usually the result of too much work being done, or multi-tasking too many tasks. This effects all levels of the organization, resources waste time switching between tasks, manager attention is harder to get, upper management has less time to track project progress and in general managers have too little time to solve project / organizational problems.

The solution is to cut the active amount of projects to a level the organization can handle, without over utilizing the critical resources. How this is done, is by staggering the projects on a common criteria. This criteria is often a project phase (like execution), or a single group of people (like architects).

By staggering projects on a common criteria, people can more easily focus on a single project at a time, thus increasing the flow or projects and lowering the amount of multitasking. Also the amount of projects in progress is lower due to this, so managers will have time to focus on the existing projects.

The important thing is to find where the WIP is collecting, and freeing that capacity to work on what actually needs to be done. This is why selecting the criteria, and following and acting on the WIP is so important.

Once the projects start flowing, and the resources have been freed from their multitasking problems, the organization can re-evaluate its capacity and act on that to further increase the flow of projects.

Fast execution management

Part VI (pages 229 – 275) presents the idea of fast execution management; a concept where any project that has a problem that would take more than 24h for the management to solve, should cause a project to be frozen and the resources to be freed to other tasks.

This requires high management dedication to projects and enough time for management to take part in projects and to solve project problems.

The idea behind the 24h timer is that any problem bigger than that, would in any case cause the progress on the project to halt on some level, so in order to avoid further damage (and the idling or multitasking of resources) the project should be halted until the problem can be solved and the resources freed to other project.

The project manager can then create a recovery plan for the project, to be utilized once the problem has been solved and the project unfrozen. This might cause a delay to the project, due to project staggering, but will make sure that a single project cannot block the flow of the whole portfolio.

Single priority system

In chapter 8 of part II (pages 63-73) is the idea behind the single priority system. What it means is that upper management needs to define a clear criteria on which project are prioritized, so that portfolio and resource managers can always have a clear criteria on how to prioritize work.

This also related to the criteria on how to stagger projects, everything needs to be static, in order for clear processes to form and the system to properly function.

The priority system on projects on only works until the project has been approved to start. After that the priority system will always be based on the project buffers. The projects that have used most of the buffer, have priority on resources, as the idea is to have all projects complete on time, not for projects to complete as soon as possible based on initial priority.

Consistent recovery approach

In chapter VI (pages 229 – 285) it is stated that a consistent recovery approach is important for every project (and the whole organization) so that project resources and project managers have the freedom and peace to do their job according to the criteria and indicators presented to them.

The recovery approach dictates that no “corrective” actions should be done when they are not needed (e.g. the project is still on the green or yellow buffer zone), and that corrective actions need to be done when they are required by the indicators (e.g. project is on the red buffer zone).

For these actions, the project managers need to have plans ready to implement once necessary, but they should not be implemented before it is necessary by the buffer rules. Also the plans should be reviewed so that only necessary actions are taken, and on the correct issues.

For example, a project manager might want to push a completion of a project task when the project is still on a green buffer zone, thus only adding unnecessary pressure on the resources. Or the project manager might want to rush a task that is not on the critical chain, when the project buffer is on the red.

So every action should be done only when absolutely needed, and be focused only on the actual issues, not anything else.

Conclusions

The authors state that by following the instructions given, organizations were able to complete projects 25% faster, without sacrificing budget or scope. Also they were able to increase the number of projects executed with the resources by 70%, and in the process generated over 50% more throughput for the organizations.

These experiences came from some of the biggest, well-known names in the world - Boeing, Rio Tinto, ABB and Chrysler, but the list also included some very small organizations that were struggling to get a few projects completed per year.

Ville Murtonen