Influence of Process Models on Requirement Volatility, and Strategies for its Management – An Empirical Investigation

Rahul Thakurta
Xavier Institute of Management, Information Systems
Bhubaneswar, India, 751013
rahul@ximb.ac.in

Abstract

The current research aims to investigate the preparedness among organizations in managing requirement volatility, and how it is influenced by process factors characterizing a software project. A combination of interviews and survey organized in two phases has been used to assess the level of awareness related to requirement volatility, and the current level of organizational competencies in dealing with the problem. Frameworks regarded suitable for handling volatility has been presented. Factors governing process model selection and usage and their applicability under requirement volatility have also been explored.

Concerns were raised over the high percentage of projects found affected because of requirement volatility. Two widely used approaches for managing projects under volatility were found to be involving the business side and resorting to iterative project development. Difference in usage of the approaches based on process models could be observed. Influence of business/customers, project complexity and managerial preferences emerged as the top three factors guiding process model selection. The variation of these factors under requirement volatility has been pointed out. The results also highlight incongruence between perception and practice related to process model usage in projects endangered because of requirement volatility.

Keywords: Requirement Volatility, Managing Requirement Volatility, Process Model, Process Maturity.

1. Introduction

Requirements are the foundation of the software development providing basis for project estimation and project planning [1]. The success of a software project is also influenced by the quality of the requirements. Although the initial sets of requirements are well documented, changes will occur during the software development lifecycle. Requirement volatility which refers to such changes in requirements during the software development life cycle has been reported as one of the main factors that cause a project to be challenged [2]. The problem assumes a still greater significance in the present context of increasing technical and business complexity. Hence the ability to manage requirement volatility in the best possible way has been identified as a necessary prerequisite behind successful project outcomes [3].

This study seeks to investigate the organizational competency regarding the problem of requirement volatility affecting software projects. We want to find out the level of awareness concerning requirement volatility, the management approaches used and how they are influenced by process characteristics. Some of the available organizational frameworks to address requirement volatility have been reported. The study also sheds light on the process models used in projects, and factors that govern their suitability and applicability in projects endangered because of requirement volatility. A process model (synonymously known as systems development life cycle model) referred above related to a conceptual model used in IT project management that describes the stages involved in an information system development project, from an initial feasibility study through maintenance of the completed application.

This research project was carried out in two phases. An exploratory research design was employed in the 1st phase where requirement volatility and its association to project attributes and management techniques were examined in depth. Senior project managers associated with software development were interviewed for gathering insights on the problem. Content analysis of interview data led to development of pertinent hypothesis for subsequent validation. The 2nd phase, in contrast to the 1st, was confirmatory in its approach. It started with the list of hypothesis identified at the end of phase-1. A web based survey was used to test the list of hypothesis in an attempt to generalize some of the findings.

This paper starts with the list of hypothesis identified at the end of phase-1 and then proceeds to validate them based on the available evidences in the survey data. Findings of the interviews have been reported simultaneously to strengthen our claim, and also to report contrasting viewpoints and arguments. Responses obtained during the interviews and the survey revealed interesting insights which led to development of emergent patterns. The paper is organized as follows. A review of relevant literatures is provided next. The subsequent section lists the research hypotheses. It is then followed by sections on methodology, study outcome, and a discussion on the results. The concluding section summarizes the key findings.

2. Related Work

Studies on software project risks have highlighted the growing concern posed by requirement volatility [4, 5, 6, 7, 8], and have stressed the need of proper management techniques in dealing with the situation. Various approaches have been discussed in the literature concerning management of requirements changes. Jones [7] presents a list of methods or techniques which can be utilized to control the rate of requirements change including joint application design (JAD), prototyping and configuration management change control board. Wiegers [9] discusses about baselining requirements with a purpose of controlling the requirements changes. Leffingwell and Widrig [10], Young [11] have both emphasized the need for proper change management planning to reduce the amount of delayed requirements changes. Ebert and Man [12] stressed the importance of contextual change management techniques to obtain success in rapidly changing business environment. The need for emphasis on project workforce management for better administration of project under requirement volatility has been stressed in Thakurta et al. [13].

Factors guiding applicability of some of the process models under requirement volatility have also been highlighted. Sudhakar [1] presents suggestions on the contribution of process models towards management of requirement volatility. His paper summarizes the advantages and disadvantages of various process models in handling volatile requirements. Madachy [14] traced the root of requirements changes to the choice of process models, regarding waterfall model to be quite rigid under changing requirements, similar findings are also reported in References [15, 16]. The characteristics of spiral model and extreme programming for use under requirement volatility have also been listed [17, 18]. Boehm [4] presents his argument in favor of spiral model and prototyping approach as best suited for controlling the risk of requirements changes. The criteria for use of Built-Fix Model have been mentioned in Aggarwal and Singh [19].

Rajlich [16] mentions about adoption of new paradigms like iterative and agile in order to manage the problem of requirement volatility. The papers discussing on process models in the context of requirement volatility have mostly highlighted the benefits and problems of the various models, but the organizational practice related to their use, suitability and applicability under volatility have not been investigated. This assumes a greater significance given the importance ascribed to the use contextual change management approaches [12]. We intend to address this research gap, and investigate the approaches available to the project managers in dealing with requirement volatility, and how they are influenced by process characteristics.

3. Research Hypothesis and Questions

Academic literatures on software project risks have emphasized the growing concern posed by requirement volatility [6]. Schmidt et al. [20] in his study on consolidating project risk factors highlights the time-variant evolutionary nature of software risks. With a range of approaches now available in organizations to manage projects under requirement volatility, we wish to find out the importance of requirement volatility in the present settings. Our first hypothesis can be stated as:

H1: Organizations still perceive requirement volatility as a significant risk affecting software projects

The characterization of process models based on degrees of flexibility could be subjective. Also individuals might have different views regarding the suitability of the available process models in handling requirement volatility. The general acceptance of the limitation of "rigid" waterfall model to incorporate changing requirements leads us to the following hypothesis concerning all process models

H2: There is a relationship between process models considered flexible and process models that are regarded suitable in handling requirement volatility

Omission of process model steps has been cited as one of the factors leading to project failure [21]. Now some projects may not use any of the available process models, or the process model used may not be suitable considering the project characteristics. If this number is greater than 50% then that will indicate that more often than not projects resorting to inappropriate process models or not using process models at all are getting endangered because of requirement volatility. The following hypothesis tests out the significance of the above scenarios on projects that were found to be endangered because of requirement volatility. Since selection of process models is done upfront, a confirmation of the hypothesis would emphasize the need of greater attention on this in order to increase chances of project success under requirement volatility.

H3: Over 50% of the projects that are endangered because of requirement volatility either don't have a process model, or have used an inappropriate process model with regard to the project

Various software organizations employ maturity models to develop and refine its software development process. A model like the CMMI (Capability Maturity Model Integration) defines practices that businesses have implemented on their way to success. CMMI describes a continuum of five stages (i.e. ad-hoc / crises, standard project management, standard project management, managed feedback, optimizing / continuous improvement) based on how well a company or organization follows common and repeatable processes to get work done. Each stage consists of a number of "Key Process Areas" (KPA's) which indicate the essential needs to be satisfied as the maturity level increases. Maturity Level-2 (standard project management) relates to organizations that have achieved standardization in their development process (i.e. having common and repeatable software development processes, deliverables, tools, etc.). At this level, "requirements management" has been identified as a KPA that needs to be satisfied. As managing requirements is an activity closely associated with change management practices [22], we assume the following to be true:

H4: Organizations having a maturity rating of two or above have a plan for managing requirement volatility

Our research also tries to answer the following questions concerning the influence of process maturity and process models on requirement volatility management techniques:

- What approaches are available for managing projects under requirement volatility?
- Whether the approaches can be grouped depending upon the process models?
- Can we categorize the approaches depending upon the process maturity?
- What are the factors that determine selection of process model in a project?
- What essential characteristics should process model posses so as to make it more acceptable in a volatile environment?

4. Methodology

4.1 Research Approach

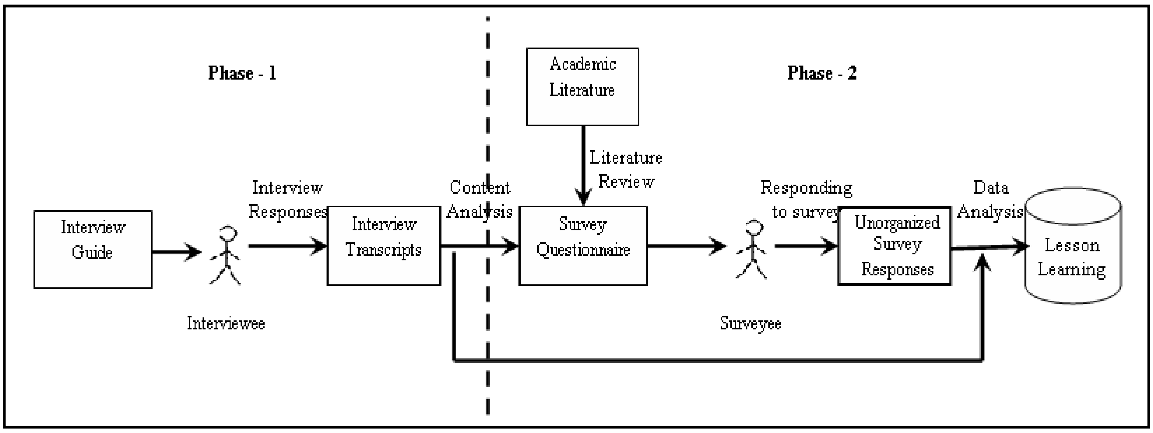

Figure 1: Research Approach

The overall organization of the study is presented in Fig. 1. The work started with a set of premises concerning the problem; interviewee responses and literature evidences were combined to develop the survey instrument for subsequent analysis. Interview responses were developed into hypothesis and research questions. These were then addressed based on a survey questionnaire.

4.1.1 Interviews

In the 1st phase of the study extending from November, 2007 to March, 2008, in-depth qualitative interviews were carried out with eleven senior software project managers belonging to various organizations. The interviews sought to capture individual experiences, opinions, perceptions and knowledge regarding preparedness of organization regarding the problem of requirement volatility affecting software projects. An Interview guide approach was used to carry out the interviews. The guide contained questions on the interviewees' demographics, organizational settings, project background information, systems development life cycle methodologies, software project success, awareness and criticality of requirement volatility, requirement change pattern and degree and effect of requirement volatility on project. Greater flexibility was offered to the respondents to express whatever they wanted in any particular order they wished. All the interviews were carried out over telephone and recorded. Information provided by the respondents was guaranteed to be anonymous, and organizational details were kept in confidence. Interviews normally lasted between one and one-half hours.

Several Ph.D. students tested the interview guide for correctness so that it complies with the guidelines prescribed in Maykut [23]. The test provided feedbacks on the overall layout of the questions and helped to clarify and sharpen some questions. The interview guide was finally pre-tested with an IS project manager. Interview notes were analyzed inductively by the constant comparative method [24]. The responses were codified, with each code representing a theme or idea with which each part of the data was associated. New themes were assigned to data that fell outside the possible alternatives. The codes that had common elements were merged to form categories, which were subsequently clustered for pattern identification.

4.1.2 Survey

A web-based survey was carried out as a part of phase-2 of the study with an objective to test out the phase-1 observations on a much wider sample. The survey contained five sections, appraising the respondents on the purpose of the study, enquiring about demographic information, their association with software projects and take on requirement volatility. One section was devoted to project specific data corresponding to a recently completed software project that had problems with requirement volatility. This was done to ensure the survey results more reliable than the software practitioners' mere opinions and generalizations. The questionnaire was pre-tested by several software developers. The questions were revised to be more unambiguous based on the feedback from these tests. Data analysis was carried out using the SPSS statistical package whenever appropriate. The basic assumptions required for the different tests were checked in advance. Where ever the data did not meet the test criteria, patterns are reported for further analysis.

4.2 Measurement of Constructs

The constructs of this research include process maturity, process model, and requirement volatility.

The process maturity framework of CMMi (Capability Maturity Model Integrated) was chosen for our study. It defines the following five maturity levels:

Table 1: Process maturity level

Maturity Level (L)	Description
L-1 (Initial)	The software process within the organization is ad-hoc and chaotic with ineffective management procedures and project plans
L-2 (Repeatable)	The organization can successfully repeat projects of the same type. However projects success depends more on individual managers and organization folklore acting as a process description.
L-3 (Defined)	The organization has a defined process with formal procedures to ensure the application to all software projects
L-4 (Managed)	In addition to the above activity, the organization has a formal program to collect quantitative process and product metrics and analyze and use these for process improvement activities
L-5 (Optimizing)	In addition to the above activity, the organization demonstrates its commitment to continuous process improvement

The second construct aims to find out the usage frequencies various process models and their ranking on flexibility and suitability dimensions. The process models were classified into the following categories as given in Table 2. Both the flexibility and suitability dimensions were measured on a 1-5 scale with five indicating "very highly" flexible or suitable.

Table 2: Process model

Process Model	Definition
Waterfall Model	It is a software development model (with strictly one Iteration/phase) in which development proceeds sequentially through the phases: requirements analysis, design, coding, testing (validation), integration, and maintenance
V-Shaped Model	This is an extension of the waterfall model which emphasizes parallelism of activities of construction and activities of verification. Here, the process steps instead of moving down in a linear way bend upwards after the coding phase resulting in the typical V shape formation.
Prototyping Model	It is a software development process that begins with requirements collection, followed by prototyping and user evaluation
Incremental-Iterative Model	Here the software project is divided into mini-projects, each of which is an iteration that results in an increment. Each iteration represents a mini-waterfall model.
Spiral Model	This supposes incremental development, using the waterfall model for each step, with more emphasis on managing risk.
Rapid Application Development (RAD)	It is a software development process that allows usable systems to be built in as little as 60-90 days, often with some compromises.
Agile Methodologies	Agile is an evolutionary approach to software development which is performed in a highly collaborative manner by self-organizing teams with the objective of producing high quality software in a cost effective and timely manner. Some of the different Agile Approaches are Extreme Programming (XP), Scrum, Pair Programming, etc.

Requirement volatility has been operationalized via many different constructs based on quantity, timing and propensity of change [25, 26, 27]. Here we intend to measure the degree of risk that is attributed to the problem of requirement volatility affecting software projects. A 1-5 scale with five denoting very high risk has been used for this purpose.

4.3 Sample Description

Experienced software professionals having atleast five years of expertise or worked as project leads, managers or equivalent were targeted in the survey to minimize guess responses. A simple random sampling strategy was adopted and Invitations to participate in the survey were mailed to members of some of the IS mailing lists available online. Follow-up invitations were emailed twice in gaps of two weeks. The survey was made available online for two months. As multiple mailing lists were targeted and these lists are dynamic, it is unknown how many potential respondents actually saw the survey invitation. An access counter on the survey page indicated a total of 176 respondents to visit the survey page, out of which 112 (64%) individuals finally completed it. Of the completed questionnaires, some were out-of-sample responses and problem responses (like multiple response), which had to be discarded. The final usable sample size came out to be 82 (47%). Factors like number of questions (44), depth of information sought, unfamiliarity of the area, amount of time required might have been instrumental behind the low completion rate of the survey. Most of the respondents were male (73%). About 51% of the respondents had over ten years of software project experience. 62% of the respondents were associated with more than ten projects. Among the industry types, information technology services were represented most (35%), followed by software (14%), consulting / engineering service (13%) and banking (12%).

The survey respondents reported being involved in 1470 projects since 2003. More than half of these projects (54.9%) were considered at risk due to requirement volatility. Probing on process related information led to data of 653 projects. Of the 653 projects, 39.9% projects used the waterfall process model and 22.8% were based on iterative-incremental process models. Agile methodologies were used in 9.9% of projects.

Respondents were also asked to report on a recently completed project that was endangered because of requirement volatility. In this case, data on 42 projects could be obtained. 32.6% of these projects represented organizations that did not use any of the maturity ratings. Of the others, level-5 organizations were found to be the most represented (32.6%), followed by level-3 organizations (11.6%). 11.9% of the endangered projects were found not to use any of the available process models. Of the rest, in contrast to the above data, iterative-incremental model was found to be the highest used (33.3%). Waterfall model was utilized in 23.8% of the projects. Other process models like v-shaped, prototyping, rapid application development, and agile methodologies were equally represented in the sample.

4.4 Validation of Research Methods

The survey instrument was validated using Straub's [28] guidelines. Pre-testing was utilized to improve the reliability of our questionnaire. Three of our interviewees also completed the survey questionnaire. Comparison of their data enabled to evaluate the construct validity of the questionnaire. The interview results also helped us to form survey questions and interpret the answers of the respondents.

Our sample represented a broad range of IS project types across a variety of industries and spanning across small to large organizations. The extensive representation of projects and organizations should reduce concerns of bias in the sample. Non response bias which poses a serious threat to the validity of the results was tested by comparing early (those received on first invitation) and late (those received after the second follow up) respondents [29]. Results revealed absence of any significant difference along key sample and project characteristics (α=0.05).

However, the results of the research should be interpreted with some caution. There are chances of observation and information bias during the interviews due to the involvement of a single observer. The results may be biased because of recollection error as some respondents reported association and details of projects executed long back. Project specific responses were also not checked against available data or views of co-project members.

Our sample is relatively small and may not be representative of all development projects and organizations. Thus the survey and interview do not provide sufficient coverage of all situations. However, the focus of our research is more on understanding of the phenomena and how it is influenced by process related factors characterizing a project. Patterns uncovered in the research are early insights, and is expected to provide basis for further work in this area.

5. Results

Here we first address the research questions, and then go on to validate our hypothesis set that we proposed in Section 3.

Research Questions

What approaches are available for managing projects under requirement volatility?

In phase-1 of the study, the interviewees were asked on approaches they think would assist in managing projects under requirement volatility. Ten approaches were identified (1-10, Table 3) to which five more approaches were added based only on available literature (11-15, Table 3) [30, 31] to arrive at the list of approaches given in Table 3. Each of these approaches was arrived at by grouping related activities under a common theme identified here by the name of the respective approach (as given under 'list of approaches'). For example the approach "involving business side into the project" encompass activities like JAD, joint review workshop, etc.

Table 3: Management approaches under requirement volatility

Sr. No.	List of Approaches
1	Involving business side into the project
2	Using iterative/phased project development approaches
3	Reducing project complexity
4	Project scope negotiation
5	Engaging in requirements management activities
6	Documentation of processes, procedures and activities
7	Adjusting project human resource
8	Using expert knowledge
9	Focusing on communications
10	Rescheduling project deadline
11	Readjusting project effort
12	Variable costing of additional requirements
13	Architecting product to withstand change
14	Training workforce
15	Adopting agile processes

The top two entries were found to be most used, the former one being *"utmost important"* irrespective of the project characteristics, with percentage shares of 11.3 and 10.2 respectively based on the survey responses.

Whether the approaches can be grouped depending upon the process models?

Data pertaining to the 42 endangered projects revealed patterns of usage of different approaches depending upon process model categories. The classification was carried out by observing the frequency of usage of the above mentioned approaches for each project model, and identifying differentiating patterns. Statistical technique like multinomial logistic regression which is appropriate for this type of analysis was not used as the sample size requirement was not met [32]. Project management while using waterfall model resorted to "reducing project complexity" and "adjusting project human resource" under requirement volatility. The two recurrent management approaches for projects utilizing the RAD model were "using expert knowledge", and "variable costing of additional requirements". Iterative-Incremental projects employed the "requirements management techniques" more often. Agile projects as expected focused more on "communications". However "documentation of processes, procedures" and activities" were also reported.

The usage of other approaches listed in Table 3 above was invariant across the process models. However among them "involving business side", "rescheduling project", "readjusting project effort", and "project scope negotiation" were used frequently in projects endangered because of requirement volatility.

Can we categorize the approaches depending upon the process maturity?

We also attempted to find out whether the organizations at different maturity levels displayed any specific pattern in usage of requirement volatility management approaches. We restricted our investigation to the approaches found invariant among the process models. Results based on 42 cases however failed to detect noticeable difference in usage pattern of approaches across maturity levels, with the top four approaches same as listed above. However approach of "architecting product to withstand change" was found to be more used in level-1 organizations.

What are the factors that determine selection of process model in a project?

During the interviews we asked our interviewees on the factors that guided selection of a process model in a project. The responses classified inductively revealed the following influential factors as given in Table 4 (the 3rd column indicates the relative influence of these factors in projects endangered because of requirement volatility)

Table 4: Process model selection factors

Sr. No.	Process Model Selection Factors	Influence (%) Under Volatility
1	Influence of business / customers	26.2%
2	Project initial estimates (size, effort/cost, duration)	4.8%
3	Maturity of the overall development process	11.9%
4	Complexity of the project	21.4%
5	Management preferences	19.0%
6	Requirements set of the project	11.9%
7	Project ultimate objectives	4.8%

Importance of the business side was stressed most as one of the interviewees remarked *"the type of our client organization affects our decision on the process models"*. Sometimes *"client's expenditure pattern"* influenced the overall decision. The other two prominent determinants were the initial project estimates and the process maturity as the interviewees felt *"If your process is mature enough, you may then customize based on project priorities and select models that puts you in a win-win situation.* Under some situations, the *"Past history of projects"* also influenced the process model selection. The relative importance of these factors however differed in a volatility environment (3rd column, Table 4). In the later case, influence of business/customers (26.2%), project complexity (21.4%), and managerial preferences (19.0%) were observed to be the top three determinant factors.

We then investigated how these factors influenced the ultimate selection of process model in a project. Results based on 42 responses revealed waterfall model to be mostly driven by management preferences. "Influence of business" and "development process maturity" also influenced its selection. Except factor no 2 ("Project initial estimates", Table 4) and 7 ("Project ultimate objectives", Table 4), all the other factors listed above were found to influence selection of iterative-incremental model, the top two being "complexity of the project" and "requirements set of the project". "Complexity of the project" was found to influence the non-adoption of agile methodologies.

What essential characteristics should process model posses so as to make it more acceptable in a volatile environment?

During interviews, we enquired on the essential characteristics that a process model should possess to be considered "ideal" under requirement volatility. The following attributes were pointed out

- It should prescribe clear and crisp processes
- It should be flexible enough to accommodate midstream requirements changes
- There should be flexibility to switch the phase without hampering traceability
- It should be able to handle unforeseen developments in project environment
- It should provide check of plan and progress so that project objectives are realized
- The process should reduce the administrative overhead

Validation of Hypotheses

H1: Organizations still perceive requirement volatility as a significant risk affecting software projects

Survey respondents were asked if despite all the methodological advancements, requirement volatility is still perceived as a significant threat to software projects. We used the descriptor "significant" to imply a rating of four or five. The number of responses falling in this category was found to be statistically significant (N=82, α=0.05). The corresponding effect size (0.911) was also "large" [33]. Based on the evidence, we could validate the hypothesis.

H2: There is a relationship between process models considered flexible and process models that are regarded suitable in handling requirement volatility

We wanted to find out if the process models that are regarded as flexible by respondents are also considered fitting under volatile development environment. The mean flexibility and suitability ratings on individual five point scales (5: highest rating on either characteristic) assigned by the survey respondents to each of the process models are given in Table 5 (N: 58)

Table 5: Mean flexibility and suitability rating for each process model

	Mean Rank (1-5 Scale)	
Process Model	**Flexibility**	**Suitability**
Waterfall Model	2.10	2.21
V-Shaped Model	2.64	2.64
Prototyping Model	3.46	3.58
Iterative-Incremental Model	3.51	3.02
Spiral Model	3.08	2.46
Rapid Application Development	3.27	3.00
Agile Methodologies	4.10	3.67

Table 6: t-test of the difference between flexibility and suitability ratings

Process Model	**Null Hypothesis**	**P-value**	**Correlation Coefficient**	**Comment**
Waterfall Model	True	0.410	0.622*	The ratings are related
V-Shaped Model	True	1.000	-	The ratings are related
Prototyping Model	True	0.377	0.687	The ratings are related
Iterative-Incremental Model	False	0.015	0.165	The difference between the two ratings is significant
Spiral Model	False	0.014	0.771*	P-value of correlation coefficient = 0.002 (Significant at α=0.05). Hence the ratings are somewhat related
Rapid Application Development	True	0.137	0.536*	The ratings are related
Agile Methodology	True	0.056	0.483*	The ratings are related

* Correlation coefficients are significant

Results of paired t-test to test for the difference among the mean ratings for each of the process models are tabulated in Table 6 (**Null Hypothesis**: There is no significant difference between the two ratings, α=0.05) Except iterative-incremental model, the mean flexibility and suitability ratings of the other process models were found to be related, providing enough evidence to validate the hypothesis.

Table 5 also shows that the respondents found the agile methodologies to be the most flexible in terms of accommodating mid-stream requirements changes, followed by the iterative-incremental and the prototyping model. The waterfall model was considered to be the least flexible as one interviewee found it to be "*extremely imperfect*" under situations of high requirement volatility.

H3: Over 50% of the projects that are endangered because of requirement volatility either don't have a process model, or have used an inappropriate process model with regard to the project

We tested the above hypothesis using the binomial test available in SPSS. 11.9% of the endangered projects (N: 42) were found not using any of the process models, and 40.4% of the remaining were unsure about the appropriateness of the process model considering their project environment. Result of 1-tailed test failed to produce statistical significance of the combined proportion (p value: 0.439, α=0.05).Our hypothesis had to be rejected.

An investigation of the process model selection factors in situations where the respondents were not convinced about the appropriateness of the process models discovered the influence of business to be the most dominant factor (29.4%), followed by management preference (23.5%) and development process maturity (17.6%).

H4: Organizations having a maturity rating of two or above have a plan for managing requirement volatility

Survey disclosed about 65.9% of the respondent organizations (N: 44) to have an organizational plan for handling the changing requirements. Further sub classification identified 25 responses belonging to organizations having a maturity rating of two or higher. 84% of these were found to have frameworks for managing requirement volatility. This number was found to be significant (p value: 0.003, α=0.05) with a corresponding effect size of 1.495 (Large). In the light of the above data we rejected the null hypothesis and accepted H4.

Some of the prominent organizational frameworks for managing requirement volatility included the formation of "Change Control Board (CCB)" and equivalent in finalizing the requirements, usage of a "Central Requirement Management (CRM)" system for tracking requirements and employing tools like "Influx", "Stats", "JIRA" and "Mercury" for change administration.

6. Discussion

The following contributions can be drawn out of the study:

The perception of the threat posed requirement volatility has been captured in the study. Even though it was rated as a significant threat affecting over half of the projects captured in the study, not all viewed it as a problem as one enquired *"Isn't it time we accepted requirement change as a part of our daily life and adapt accordingly?"*

Different approaches for managing projects under requirement volatility could be identified and also classified based on the process model employed. Quite surprisingly the practice of "involving business side" was found frequent even in projects using the waterfall model. This departure provides indication of the tendency of non-adheration to the process model guidelines, and a gradual shift from the traditional process models to more customized models as noted by one of our interviewee *"Depending upon technology and depending upon the application of the technologies we have our own process models which we have created in these years based on experience. For a customer in a particular application area we then take up the model that is fitting".* A couple of examples of such tailored process models included use of PROPS and SDLC+ models which are customized waterfall model. Among the other approaches for managing projects under requirement volatility, the practice of "architecting product to withstand change" in a level-1 organization was also quite unusual.

Another evidence of the departure was the practice of documentation in endangered projects using the agile methodology. Some of the participant organizations had their customized agile frameworks to suit the needs of the involved parties as one commented "we *designed our own agile process together with the customer, because we believe use of the best out of some methods will help more".*

The fact that some of the approaches were used frequently irrespective of the process characteristics could be because some of the project managers were using approaches on a hit and trial basis as one quoted *"We have tried several approaches under crunch situations hoping that some would click and steer our project to safety".*

Placed high on the process maturity ladder did not mean a respite from the problems arising because of requirement volatility. Projects belonging to level-5 organizations' were found to be most represented among the endangered sample.

The influence of business on the selection of process models could be noted. Project complexity was observed as a major hurdle in the adoption of agile methodology [34]. Similar findings were also produced by our study where inherent complex projects were found to avoid the use of agile methodology.

The selection of the waterfall model was found to be driven mostly by management preferences and business influence. It appears that the available expertise in successfully using the waterfall model, or the natural resistance to adoption of new frameworks could be guiding the managers or the business representatives behind continuing with waterfall, which future studies could look to investigate.

A majority of the projects that were endangered because of volatility had their process model decided by the business/customers as one remarked *“It is not us that decide the process model for our project. Our client has a specific process model and we have to follow the model even if it’s not appropriate for the project”*. As the clients may not be aware of the problem posed by requirement volatility, these projects faced difficulties because of the changing requirements.

7. Conclusion

Software projects continue to be troubled because of requirement volatility. This paper presents the organizational preparedness concerning requirement volatility, the management approaches used, and how they are influenced by process characteristics. Even though some of the findings were expected, several interesting patterns emerged which could be explored in future studies.

Fifteen different approaches to managing projects under requirement volatility could be identified, among which “involving business side” and “using iterative project development approaches” emerged to be the most frequent. Some variations in the usage of the approaches based on process models could be noted. However process maturity didn’t produce any noticeable differences. The top three factors guiding selection of process models in endangered projects were influence of business/customers, project complexity and managerial preferences. Although the usage of iterative approaches was stressed, waterfall still emerged as most frequently used process model based on the study sample. Even though not widely used, agile methodology was rated as the most flexible and best suited of all the process models under requirement volatility.

The importance of our study was echoed by one respondent as *“The area of study is quite interesting and have provided means of finalizing the points which might hamper a Project in its execution and implementation”* Future research can look to statistically validate the usage patterns of the different management approaches depending upon process characteristics and investigate the influence of cultural factors on the overall context. The contribution of these management strategies on the various dimensions of project success as identified in the literature can be examined. The behavioral and social factors influencing use of specific process models even under unfavorable situations is also worth exploring.

Acknowledgements

The author would like to express his sincere gratitude to Dr. Frederik Ahlemann of European Business School for his attention to the work and his continuous comments during the entire duration of the project. The author is indebted to the anonymous reviewer(s) whose inputs helped immensely to improve the quality of the submission.

References

[1] M. Sudhakar. *Managing the Impact of Requirements Volatility*. Master Thesis. Department of Computing Science, Umeå University, Umeå, Sweden. 2005.

[2] A. Tiwana, and M. Keil, “The one minute risk assessment tool,” *Communications of the ACM*, 2004.

[3] M. Ben-Menachem, *Software Configuration Management Guidebook*, McGraw-Hill International (UK) Limited, 1994.

[4] B. Boehm, “Software risk management: principles and practices,” *IEEE Software*, 1991.

[5] A.M. Davis, N. Nurmuliani, S. Park, and D. Zowghi, "Requirements change: what’s the alternative?" in *Proceedings of the 32nd Annual IEEE International Computer Software and Applications Conference*, July 2008, pp. 635-638.

[6] A. Tirwana, and M. Keil, “Functionality risk in information systems development: an empirical investigation,” *IEEE Transactions on Engineering Management*, vol. 53, no. 3, pp. 412–425, 2006.

[7] C. Jones, *Software Engineering Best Practices: Lessons from Successful Projects in the Top Companies*, McGraw-Hill Osborne Media, 1st ed., 2009.

[8] S. Ferreira, J. Collofello, D. Shunk, and G. Mackulak, "*Understanding the effects of requirements volatility in software engineering by using analytical modeling and software process simulation*," Journal of Systems and Software, vol. 82, no. 10, pp. 1568-1577, 2009.

[9] K. Wiegers, *Software Requirements*, Microsoft Press. 1999.

[10] D. Leffingwell, and D. Widrig, *Managing Software Requirements: A Unified Approach*, Addison-Wesley Longman. Inc., 2000.

[11] R.R. Young, *Effective Requirements Practices*, Addison-Wesley, 2001.

[12] C. Ebert, and J. Man, “Requirements uncertainty: influencing factors and concrete improvements,” in *Proceedings of the ICSE*, 2005.

[13] R. Thakurta, R. Roy and S. Bhattacharya, “Impact of requirements discovery pattern on software project outcome: preliminary results,” in *Proceedings of Hawaii International Conference on System Sciences (HICSS 42)*, Waikoloa, Hawaii, USA, Jan 5-8, 2009.

[14] R. Madachy, B. Boehm, and J. Lane, “Assessing hybrid incremental processes for SISOS development,” *Software Process: Improvement and Practice*, vol. 12, no. 5, pp. 461–473, 2007.

[15] G. Kontonya, and I. Sommerville, *Requirements Engineering Process and Techniques*, Wiley Publications Reprinted, February 2002.

[16] V. Rajlich, “Changing the paradigm of software engineering,” *Communications of the ACM*, vol. 49, no. 8, August 2006.

[17] K.B. Hass, *Managing Complex Projects: A New Model*, Management Concepts, October 2008.

[18] K. Beck, *Extreme programming Explained: Embrace Change*, Addition-Wesley, 1999.

[19] K.K. Aggarwal, and Y. Singh, *Software Engineering*, New Age International Pvt Ltd Publishers, 1st ed., December 2008.

[20] R. Schmidt, K. Lyytinen, M. Keil, and P. Culle, “Identifying software project risks: an international delphi study,” *Journal of Management Information Systems*, 2001.

[21] G. Stepanek, *Software Project Secrets: Why Software Projects Fail* (Expert's Voice), Apress, 2005.

[22] D. Pfahl, and K. Lebsanft, “Using simulation to analyze the impact of software requirements volatility on project performance,” *Information and Software Technology*, 2000.

[23] P. Maykut, *Beginning Qualitative Research: A Philosophical and Practical Guide* (The Falmer Press Teachers' Library, 6), Routledge, 1994.

[24] J. Hewitt-Taylor, “Use of constant comparative analysis in qualitative research,” *Nursing Standard*, vol. 15, no. 42, 2001.

[25] V. Ambriola, and V. Gervasi, “Process metrics for requirement analysis,” in *Proceedings of the 7th European Workshop on Software Process Technology*, Kaprun, Austria, 2000.

[26] T. Javed, M. Maqsood, and Q. Durrani, “A study to investigate the impact of requirements instability on software defects,” *ACM SIGSOFT Software Engineering Notes Archive*, vol. 29, no. 3, 2004.

[27] R.J. Costello, and D. Liu, “Metrics for requirements engineering,” *Journal of Systems and Software*, 1995.

[28] D.W. Straub, “Validating instruments in MIS research,” *MIS Quarterly*, vol. 13, no. 2, 1989.

[29] J.S. Armstrong, and T.S. Overton, “Estimating non-response bias in mail surveys,” *Journal of Marketing Research*, vol. 14, no. 3, 1977.

[30] S.W. Ambler, *The Change Prevention Process Anti-Pattern*, 2007.

[31] C. Jones, “Strategies for managing requirements creep,” *Software Productivity Research*, 1996.

[32] L.S. Meyers, G.C. Gamst, and A.J. Guarino, *Applied Multivariate Research: Design and Interpretation*, Sage Publications. 2005.

[33] R.J. Grissom, and J.J. Kim, *Effect Sizes for Research: A Broad Practical Approach*, Routledge Academic, 1st ed., March 2005.

[34] A. Khan, and S. Balbo, *Agile versus Heavyweight Web Development: an Australian Survey*, Department of Information Systems, The University of Melbourne, Australia, 2005.