Evaluation of Frameworks for Risk Scheduling and Requirement Volatility with Quality Standards in ICT Project

Malaya Kumar Nayak
Research Scholar,
Department of Computer Science & Applications,
Utkal University,
Bhubaneswar, INDIA

Dr Sanghamitra Mohanty
Professor and Head,
Department of Computer Science & Applications,
Utkal University,
Bhubaneswar, INDIA

ABSTRACT
One of the most critical factors of risk scheduling in ICT project has been found to be requirement volatility. The reason of excessive failure rates in ICT project is identified to be imprecise requirement analysis, frequently changing requirements, inability of project leader to schedule the risk with proper effective plan considering cost and resources involved and adaption of inappropriate risk scheduling concepts. Maximum research in this area is to generate a new framework for mitigating such issues from scratch, which is again not at all cost effective and reliable. This research journal focuses mainly on the organization practicing quality standards like TQM, ISO, CMM etc., with diversified risk scheduling frameworks which has been comparatively discussed for find their effectiveness. Based on the discussion, certain suggestion has been made which will be of high value for future researchers for mitigating risk in terms of requirement volatility.

General Terms
Software Engineering, Requirements/Specifications, Project and People Management

Keywords
Risk planning, ICT, Risk Scheduling, Project planning, Requirement volatility

1. INTRODUCTION
Requirements volatility is an estimation of how much program's requirements change once coding being done [1]. Projects for which the requirements changes majorly after coding has been initiated have a high scope of volatility, while projects whose requirements are moderately stable have a minimum scope of volatility [2]. One of the prominent causes of maximum unsuccessful score of risk management in software development is lack of interest of the technical managers for appropriate estimations in order to evaluate and supervise the requirement volatility in software projects [3]. ICT project risk evaluation is one of the important task on which requirement volatility depends, and for accurate risk management it is very significant to execute suitable risk assessment procedures. In case the imprecise requirements are not properly evaluated with prioritization, then mitigating with such risks those are not expected to transpire can completely waste organization resources. Therefore it is very crucial to schedule the risk in ICT project as well as to prioritize it for better cost estimation for an efficient project management. This research journal will discuss

certain existing risk assessment models, analysis and comparison among them and the effect of local organizational factors on the efficacy of models based on the quality management standards practices is following.

2. BACKGROUND
Although every companies follow a standard procedure of Software Development Life Cycle (SDLC), but few follow it strictly without any deviation. Majority of the reason points out in efficient requirement analysis from the clients and thereby giving rise to bring forward an effective risk scheduling practices. In case of inefficient risk scheduling in mitigating requirement volatility, it has a major impact on quality, scope, as well as cost of the ICT project. Risk scheduling in ICT project is considered as the backbone for the successful deployment of project as when it is hit be issues of requirement volatility, in specific, the organizations has to review the probable cost estimation and choose to act on it. In case of failing to understand the requirement from the client side or the offshore partners, the project manager has to invest expensive steps for clarifying the requirements, which is definitely time consuming. In case of ignorance, the consequence of final stage of project is devastating failure of all project objectives. The intention of estimating requirement volatility is to evaluate the intensity of risk and schedule those parameters in more consequential array and institutes a feasibility and impact matrix to evaluate the risk aftermath.

$$\text{Risk Aftermath} = P \times I \qquad (1)$$

Where, P = Probability of risk occurrence, and I = Impact of the risk if it occurred.

3. RISK SCHEDULING MODELS
Certain researches in the field of requirement volatility as well as risk scheduling have also being seen to have some productive output for the ICT project in order to schedule risk. The most recent analysis has shown three effective models for software risk scheduling which is as described below [4] [5].

3.1 First Risk Scheduling Model
This is first model under discussion which is basically a framework for identifying ICT risk involved in project development. The work done in this framework is more prominent on practitioners than any other's work [6]. Primarily is this that projects and environment are very much dissimilar

from the instance when the work is presented. Secondarily, technology as well as organization structure has enormously undergone changed. That will be the reason that an analysis to discover universal requirement volatility record in recent development has been performed with the certain objectives. First is to find the parameter, which project leaders, distinguishes as risk and also identifying that which factors are more vital in view of project leaders. Second is to classify risk parameters in a way that common improvement policy can be used for each classification. Robust association between importance of mitigating imprecise requirement and apparent intensity of managing was observed very crucial; as even with high risk, low apparent intensity of project risk scheduling permits a few for encountering that ICT project risk.

This is a very simplistic and implementable model concentrates on scheduling risk related to a very vague or imprecise requirements only be highly qualified professionals. The work has an importance of scheduling risk and various intensity of managing a project those technical leaders has. Basically, it recommends an analytical framework for scheduling risk with respect to requirement volatility for diversified types of other risk classifications with Strategical solution for each classification of risk.

3.2 Second Risk Scheduling Model

This is the second model under discussion which is about a format risk assessment framework for analyzing software. In case of traditional software development, ICT project requirement frequently changes as development proceeds. In fact the surfacing viewpoint has now turned into the standard in ICT project risk management and as such a query of schedule and expenditure overrun becomes serious where the solution lies in executing appropriate risk scheduling. Recently early requirement volatility is an amorphous problem, which depends on individual human judgments and unreasonable hypothesis such as, not altering necessities and work breakdown structure. To highlight such issues, risk assessment has to be more ordered, efficient, and goal oriented. There is no consideration of requirement volatility in the existing models, which is one of the significant parameter in ICT projects. Some other important parameters were also not considered like complexity of the project, skill gap analysis, human resources, and efficiency of the project team involved. This model was found to analyze and address all these critical issues. This framework is based on requirement volatility, complexity and efficiency [7]. The Requirement volatility (RV) can be estimated by summing up Requirement Birth Rate (BR) and Requirement Death Rate (DR)

$$RV = BR + DR \tag{2}$$

Where;

$$BR = (NR / TR) x100 \tag{3}$$

$$DR = (Del_g / TR) x100 \tag{4}$$

Where,

NR = Number of new Requirements

Del_R = number requirements deleted

TR = Total number of Requirements

Large Granularity Complexity (LGC) metrics can be used for calculating complexity.

$$LGC = O + D + T \tag{5}$$

Where,

O = No. of atomic operators (function or state machine)

D = No. of data stream (data connection between operator)

T = No. of abstract data type required for the system

Efficiency (Productivity) can be calculated as:

$$EF = Direct\,Labor\,Time\,/\,Idle\,Time \tag{6}$$

As per the model, a random variable x is said to have a Weibull distribution [8] with the parameter α, β and γ (with α > 0, β > 0) probability distribution function (PDF) and cumulative distribution function (CDF) are of x are respectively.

$$PDF: f(x;a,b,g) = i \begin{cases} 0, & x < g \end{cases} \tag{7}$$

$$\hat{i}(a/ba)(x-g)a-1 \quad \exp(-((x-g)/b)a), x^3 g$$

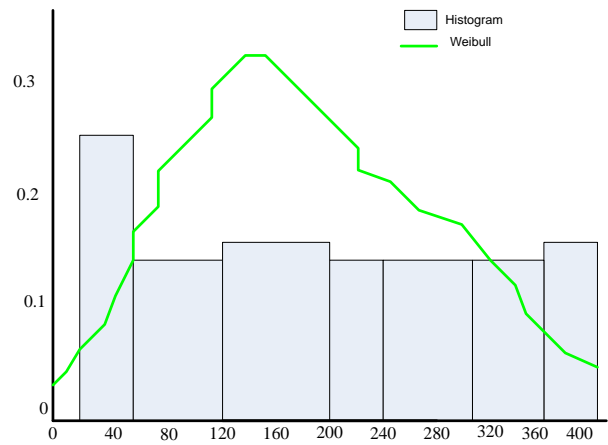$$CDF: F(x;a,b,g) = i \begin{cases} 0, & x < g \end{cases} \tag{8}$$

$$\hat{i}1 - \exp(-((x-g)/b)a)x^3 g$$

Fig. 1: PDF with Three Parameters

According to the model:

If (EF > 2.0) then a = 1.95;

$$g = 22x0.32x(13x\ln(LGC) - 82);$$

$$b = g/(5.71 + (RV - 20)x0.046);$$

else a = 2.5;

$$g = 22x0.85x(13x\ln(LGC) - 82);$$

$$b = g/(5.47 + (RV - 20)x0.114);$$

end if;

This model is perfectly suited for projects, which are evolutionary in nature. The results of the model can be validated by COCOMO [9].

3.3 Third Risk Scheduling Model

This third model is about software risk assessment which highlights an issue of unsuccessful failure for designing product within specified time frame and allocated cost. It also discusses another problem which either the product is in accordance to client's condition along with their satisfaction level. It assumes 9 parameters which give birth of diversified category of requirement volatility in the stage of near completion of the ICT project and will then definitely influence the defined project development timetable, quality along with cost involved. Parameters are product's complexity, human resources involvement in the ICT project, targets for reliability, requirement of product, cost estimation methodology, process monitoring, software development life cycle adopted, software usability and project development technology. It associates the Client's Feedback Index with the project wholesome risk and offers following implications. For example if Client's Feedback Index is less than 5, than we can call that project is not acceptable, if it is between 5 and 10 then the project may be ended with far-reaching schedule and substandard. If it is between 15 and 10 then project completed within due specified dates in pre-allocated cost and as per terms and specifications of the client. For evaluation the project manager answers the questionnaire, recommended by this model, according to the type of the project. Estimate the statistical value connected with the selected choice. Evaluate the normalized figure for each of the nine risk parameters by appending the statistical scores of the question project manager tried and by dividing the total number of query tried. Then estimate the normalized requirement volatility for the project by using following formula:

$$\text{Normalized } RV = R_n = (R - R_{\min})/(R_{\max} - R_{\min}) \qquad (9)$$

It gives an objective numerical figure for nine areas of the project and also recommended an empirical formula to evaluate for entire project using the nine-risk parameter of the project. It can be estimated. It associates the risk value with the customer feedback index, which is an indication of the customer feedback that relates to the standard of the product. It has minimum empirical score when it comes to associate risk value with client's feedback index as results are based on same categories of ICT projects.

4. IMPACT OF RISK FACTORS ON MODEL

It is very important to chose a right model for mitigating requirement volatility in the field of risk scheduling process, as already resources and cost are deployed in the risk scheduling process and if in appropriate model is selected, then, other than slicing down the cost and schedule, it can increment it. Processes are the foremost local organizational parameter which can influence the efficiency of a adopted risk scheduling model and industry quality standards (ISO, TQM, CMMI, KAIZEN, Six Sigma) is one of the principles which shows the height of organization in terms of quality and method maturity. The information which the corporate is implementing what quality standards and their respective practices of quality standards can

assist the activity of selecting appropriate risk assessment model.

4.1 Quality Standards and Risk Scheduling:

It has been seen that the structure, categorization and vocabulary of parameters and metrics applicable to software quality management have been derived or extracted from the ISO 9126-3 and the subsequent 25000:2005 quality model [10]. In the area of Information technology, software process improvement is a major conception and it has actually a potential of mitigating various software project risk scheduling in terms of requirement volatility prominently [11]. Among the entire quality standards, the frequently used standards are ISO standards, Total Quality Management (TQM), Kaizen, and Capability Maturity Model (CMM). Majority of the MNC organization is seen to practice CMM which is designed by Software Engineering Institute (SEI). CMMI (Capability Maturity Model Integration) which facilitates organizations to gauge their "maturity" on a scale of one to five which is represented as initial, repeatable, defined, managed and optimizing in working on software engineering. Enhancement is accomplished by action policy for neglected areas. Risk scheduling is covered at third capability maturity level that facilitates an infrastructure for many IT services, as shown in Figure 2.
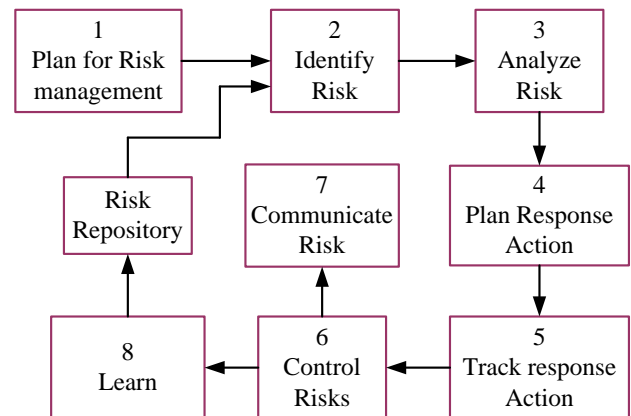

Fig. 2: Risk Scheduling Process

In the first level, the organization defines very few procedures, lack robust project management practices and majorly depends on human resources capabilities. Requirement specification is very frequently is never predictable as it is uniformly changing along with the project development is in progress. Such types of organizations in this level do not have an efficient project management processes including risk scheduling process and will not support appropriate risk scheduling. In this level, if a project manager is thinking for investing resources greatly on risk scheduling, than it can be said that the organization is taking extreme risks which cannot be overcome, as in risk planning procedures are ignored to congregate the project deadline therefore, it may have a blow on impact on the organization [12]. As project leaders are majorly dependable therefore the first model can be used, as it is subjective yet it is consistent and it depends on project leader's efficiency of associating risk and intensity of risk and quality control. Once a project leader discovers this association, the recommended encountering strategies for risk can be planned. The second model cannot be executed in companies not obeying the normal quality standards

and practices for project management. As implementation of this model needs parameters and factors which cannot be accomplished if proper Requirements Management in accordance of standards of quality in software project development is not in practiced.

The fundamental of the project management processes are firmly established in the second level where managing and planning of new requirement is based on previously maintained records and level of accomplishments in past will be repeated. So this level can assure more error free results compared to previous level in risk scheduling. This level lacks the support of organized and documented plan for risk management though first model can be mechanized in this stage. Along a distinction with previous level, this level can let the organization to use second model as they executes requirement management process here. Software Product Engineering of level 2 recommends that the requirement documents be managed through version-control and change control practices, this can help in calculating metrics which are required for second model to avoid the most frequent problem of requirement volatility.

The organization which comes under the third level can be called as consistently standardized as both software engineering and management are established and repeatable. It can be evenly said that a company with a distinct set of quality standards for risk management processes and provides mechanism to support for executing such quality standards can be considered as more mature than a company with only informal standard definitions. For overcoming the risk issues like requirement volatility, any undocumented risk related parameters cannot be accounted to mitigate risks. The development scenario must allows proper documentation of all the steps of the requirement understanding from the client as well as all the formal communication for requirements with client and the development team should be properly analyzed and documented to avoid requirement change in the progress stage of the ICT software project development. Implementing the precise quality standards ensures such practices. This level guarantees the success of the implementing risk scheduling with respect to requirement volatility as this level accounts all the previous project methodology, it success and failures, and improvise in the current project management in analyzing requirements from the client, thereby this level can be considered as cost-effective and efficient risk scheduling level.

5. EVALUATION OF MODELS
The above discussed models are evaluated in diversified companies which follows different quality standards and practices. A survey has been conducted to validate first and third models.

This model is observed to not produce consistent measures of risk schedule for these types of project development. But it does not specify that this framework is not effectual as certain implications found in this framework indicate that the framework can be appropriate on traditional ICT project development. It has been seen that in order to generate Weibull curve the factor β has to be much higher than γ, and in context of this framework, it is only feasible when requirement volatility is much greater which is an intrinsic charecteristics the environment of traditional ICT project development. Based on this fact, it can be stated that this framework would be efficient for traditional ICT project development. This empirical concept

was evaluated and standardized by the author of research paper and the deployment of this framework was also observed in other research work [13]. The evaluation has also identified certain issues with this framework. The framework was found to be complicated, estimating factors required by the model like requirement volatility and risk scheduling will require proper elucidation of various multifaceted factors, which if estimated inappropriately will result in high cost expenditure. The time for execution is found to be high, while the project leaders normally don't have much time. No way, it is appropriate for ICT software project development other than traditional type. In short, this framework can only be used for frequent changes in requirements for ICT project of complex type with some targeted evaluation required.

6. RESULTANT RISK SCHEDULING MODEL
A survey is conducted at companies with different level of quality standards in order to recognize more serious issues of requirement volatility. The findings was intended to conclude which framework is the best suited for companies encountering failures in maintaining risk scheduling schemes. The target of the evaluation by number of project leaders is estimate the risk identification which could possibly have serious impact on ICT project schedules, expenditure etc.

6.1 Strategies for Major Risk Identification
There is a diversified outcome of the survey where the similar risk is evaluated by different participants with varied prioritization. A nontrivial strategy is adopted to understand the seriousness of risk involved. A Pseudorandom weight of 1, 2, 3 is assigned to risk with 3rd, 2nd and 1st priority order. Therefore, if the same risk is found in varied order than it will possess the cumulative risk weightage value as estimated by product of risk weight value with number of human resource selected at 1st priority which is again summed up with risk weight value with number of human resources at the 2nd priority and it continues. There for final risk weight was normalized. See Table 1.

$$\text{Rank of risk} = (RVFxFF) + (RVSxFS) + (RVTxFT) \quad (10)$$

Where,

RVF = Risk value having 1st priority

FF = Frequency of risk at 1st priority order

RVS = Risk value having 2nd priority

FS = Frequency of risk at 2nd priority order

RVT = Risk value having 3rd priority

FT = Frequency of risk at 3rd priority order

Formula: $R \text{ (normalized)} = \{R - R(\min)\} / R(\max) - R(\min)$

Project leaders should first accept those risk factors which have comparatively maximum relative significance and have greater professed score of control. It is then feasible to design risk encountering polices depending on the nature of risk can be developed. Frequency of risks in 1st priority cannot be efficiently controlled by project leaders. Therefore, formal step for executing the framework will be primarily to estimate the

risk ranking according to ICT software, secondarily to classify it in the highlighted model, and finally to relate certain prescribed method to those risks factors which lie in third column.

Table 1. Normalized risk weights (six risk factors)

Risk Items	1st Priority	2nd Priority	3rd Priority	Risk Weight	Normalized Risk Weight
Requirements Volatility	7	3	2	29	1
Incorrect Estimation of project timelines	3	4	2	19	0.6
Lack of Skilled Resources	2	6	5	17	0.5
Improper Requirements Gathering	2	3	0	12	0.4
Customer involvement	2	0	1	7	0.2
Political, social and market instability	1	1	2	7	0.2

6.2 Comparative Analysis of Models

The framework of first model is on evaluating the major risk and then analyzing the ICT project subjected to those risks. It assumes that if the risk is raised by project leaders than they can take up appropriate risk scheduling measures to encounter it. In case the project leaders have maximum score of scheduling the risk and that risk has comparatively greater significance than a project leader should assume those risk first for encounter strategy. As there are very few project leaders can do for those risks factors, which are not in their control. The other two models (second and third) did not assume this factor. All the three models other than second model are based on targeted evaluation. But the problem with second model is it's difficult for estimating the factors which is needed by the framework, besides it is appropriate for ICT software projects. In comparison to derived model with first model, it is observed that this framework has significant risk factors which are not described by first model.

Table 2. Comparative analysis results

Parameter	Complexity	Empirical Value	Validated
1st Model	Low	High	Yes
2nd Model	High	Medium	No
3rd Model	Medium	Low	Yes

7. CONCLUSION

The research journal highlight about the imprecise requirement understanding, which is one of the vital factor behind every ICT project with high cost. This research work is designed with an aim for analyzing various effective risk scheduling frameworks and concluding to a consequences if their robustness and efficiency by conducting their comparative analysis. Some models were evaluated and were found to be effective. These two frameworks are appropriate for the pre-stage of the software development life cycle. It is also helpful to deploy the effective framework which is very expandable and scalable inspite of wasting resources for mechanizing new framework. Based on the analyzation of the frameworks discussed, a framework has been formulated for companies, which comes under the different levels of quality standards (Like ISO, TQM, CMM). Various risk scheduling process are the major local organizational factor that can impact on the effectiveness of the framework, so it is imperative to choose the appropriate evaluation framework by associating the quality standards of an organization with the framework. In order to facilitate project leader for adopting the right framework, the described framework has correlated with different quality standards. The existing research on risk scheduling and requirement volatility has enormous feasibility to rise in multi-directions.

8. REFERENCES

[1] Susan Ferreira, James Collofello, Dan Shunk, and Gerald Mackulak, Understanding the effects of requirements volatility in software engineering by using analytical modeling and software process simulation, Journal of Systems and Software Volume 82, Issue 10, October 2009, Pages 1568-1577

[2] Yashwant K. Malaiya, Jason Denton, Requirements Volatility and Defect Density, Software Reliability Engineering, International Symposium on 1999 , 10th International Symposium on Software Reliability Engineering

[3] Mark Keil, Paul E. Cule, Kalle Lyytinen and Roy C. Schmidt, 2008. A framework for identifying software project risks, 41(11): 76 – 83

[4] Nogueira, J.C., Luqi; S. Bhattacharya, 2009. A Risk Assessment Model for Software Prototyping Projects, published as proceedings of 11th International Workshop on Rapid System Prototyping, pp: 28 - 33.

[5] Say-Wei Foo, 2009. Software Risk Assessment Model, published in Proceedings of IEEE International Conference on Management of Innovation and Technology, 2: 536-544.

[6] Henri Barki, Suzanne Rivard, Jean Talbot, 2007. Toward an assessment of software development risk, published in Journal of Management Information Systems, 10(2): 203 - 225.

[7] Bee Bee Chua, June M. Verner, 2009. Frontiers in Artificial Intelligence and Applications, published as proceeding of the Conference on New Trends in Software Methodologies, Tools and Techniques, pp: 111-125.

[8] J. Nogueira, Luqi, V. Berzins, and N. Nada, "A Formal Risk Assessment Model for Software Evolution", Proceedings of the 22nd International Conference on Software Engineering(ICSE 2000), 2nd International Workshop on Economics-Driven Software Engineering Research (EDSER-2), Limerick, Ireland, June 4-11, 2000. (No page numbers used).

[9] Xishi Huang a, Danny Ho b, Jing Ren a, Luiz F. Capretz, 2007, Improving the COCOMO model using a neuro-fuzzy approach, ScienceDirect, Applied Soft Computing 7: 29–40

[10] Software Quality, http://en.wikipedia.org/wiki/Software_ quality [Accessed on 28th July, 2011]

[11] Chun-guang Pan and Ying-wu Chen, 2008. An optimization Model of CMMI-Based Software Project Risk Response Planning, published in International Journal of Applied Mathematics and Computer Sciences, 3: 105-113.

[12] Don M. Chance, Robert Brooks, Introduction to Derivatives and Risk Management, Cengage Learning, 2009 - Business & Economics - 672 pages

[13] Jiamthubthugsin, W. and D. Sutivong, 2006. Resource Decisions in Software Development Using Risk Assessment Model, published as Proceedings of the 39th Annual Hawaii International Conference on System Sciences, 9(04-07): 229-238.