

# Software Risk Management- An Integrated Approach

GJCST Classification  
B.7, K.6.1

Mr. N. Sathish Kumar<sup>1</sup>, Mr. A. Vinay Sagar<sup>2</sup>, Mr. Y. Sudheer<sup>3</sup>

**Abstract-**The risk management is an integrated approach, it is essential for every project i.e. from small size to large size projects. In this paper, we present a new approach for managing the risk in easy and effective manner. This approach is for minimizing the confusion and loss associated with a project. It is performed continually over the life of a program and it can be achieved in several phases which include Risk Identification, Risk Priority, Risk Management Plan, Risk Monitoring and Internal Audit. The risk management plan includes product size, business impact, customer, process, technology, development environment, staffing (size and experience), schedule, and cost.

## I. INTRODUCTION

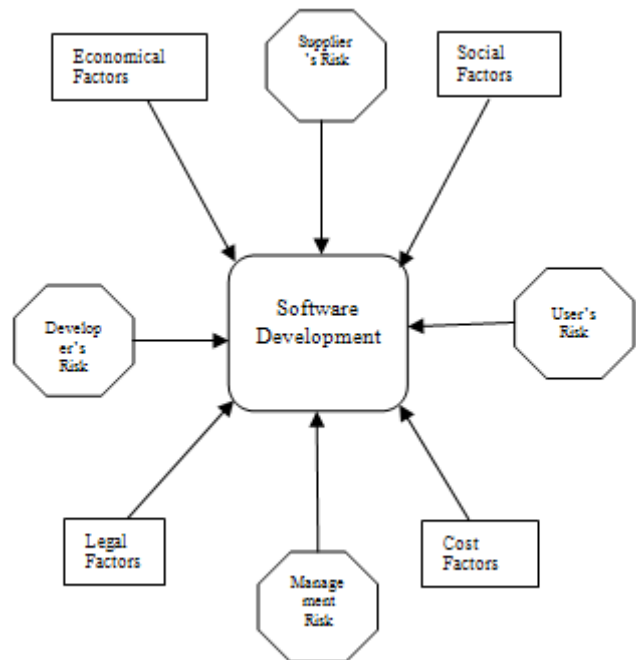
Software Risk Management is an integrated approach for minimizing the uncertainty and potential loss associated with a project by providing insights to support informed decision making. It is performed continually over the life of a program, from initiation to retirement. Some categories of risk include product size, business impact, customer-related, process, technology, development environment, staffing (size and experience), schedule, and cost. A risk is an event or condition that, if it occurs, has a positive or negative effect on a project's objectives. The three common characteristics of risk are (1) it represents a future event, (2) it has a probability of occurring of greater than 0%, but less than 100%, and (3) the consequence of the risk must be unexpected or unplanned for. Future events can be categorized as positive risk if their consequences are favorable or as negative risk if their consequences are unfavorable. Risk Management concentrates on performing bottom-up, detailed, continuous assessment of risk and opportunity. It focuses on addressing the day-to-day operational risks that a program faces. Risk Management follows a two-stage, repeatable process of assessment i.e., the identification, estimation and evaluation of the risks confronting a program and management i.e., the planning for, monitoring of, and controlling of the means to eliminate or reduce the likelihood or consequences of the risks discovered. It is performed continually over the life of a program. There are a variety of risks that confront the global software industry.

About<sup>1</sup> - Associate Professor, Department of CSE, SVS Institute of Technology, Bheemaram, Hanamkonda, Waranga-506001, A.P., INDIA <sup>2</sup> sathish4info@gmail.com

About<sup>2</sup> - Assistant Professor, Department of CSE, SVS Institute of Technology Bheemaram, Hanamkonda, Waranga-506001, A.P., INDIA anchurvinay@gmail.com

About<sup>3</sup> - Assistant Professor, Department of CSE, SVS Institute of Technology Bheemaram, Hanamkonda, Warangal-506001, A.P., INDIA sudheer.yethipathi@gmail.com

The characteristics of the legal, social, economic and competitive environments impose constraints and opportunities that help to define the nature of the risks for suppliers, buyers, and other stakeholders in the software acquisition and development process. Enterprise Risk Management (Corporate boards, CEOs, CFOs and other members) and Operational risk management (Focus on integrated governance, risk and compliance, regulatory measures) are the existing systems, in which there are many problems are occurred.



## II. THE MOST COMMON & SERIOUS SOFTWARE RISKS

There are various reasons as to why earlier risk management is difficult to implement effectively. We observed that few projects have more than 20 active risk factors at any one time. Another reason for the relatively low implementation of earlier risk management methods in practice that the fact that risk is a fuzzy concept for which users lack the necessary tools to more accurately define risk for a deeper analysis. Users may not have the ability to provide accurate estimates for probability and loss projections required for a reliable risk analysis. Existing risk management methods may not provide support for dealing with these differences.

Risks may also affect a project in more than one way and most risk management approaches focus on cost, schedule or quality risks, but there may be combinations of risks or other characteristics such as future required maintenance, company reputation, or potential liability that should be considered important in decision-making process. Finally, many current risk management techniques may be perceived as too costly or too complex to use. Simple, straightforward risk management techniques that require an acceptable amount of time to produce results might be the answer. A key software risk area, and one that has an immediate and potentially catastrophic impact on software projects, is related to the development of software. Software security policies do not fully address the risk of using foreign suppliers to develop weapon system software. Current guidance allows program managers discretion in managing foreign involvement in software development, without requiring them to identify such risks.

### III. RISK MANAGEMENT PROCESS

An endless list of all possible risks that need to be considered in a software development process. There are any numbers of internal/external scenarios that may influence the possible risks that our software on is susceptible. The key to successful risk management lies in the ability to incorporate a new risk management process that addresses the complementary needs of the business and its customers. Risks can be identified at the very earliest stages of the software life-cycle. The ability to identify risks earlier translates into earlier risk removal, at less cost, which promotes higher project success probability. Software project risk defines the operational, organizational and contractual software development parameters. Software process risk includes both management and technical work procedures. In management procedures, process risk may be found in activities such as planning, staffing, tracking, quality assurance and configuration management. In technical processes, it may be found in activities such as requirements analysis, design, code and test. Planning is the management process risk that is most often reported. Software product risk contains intermediate and final work product characteristics. Primarily a technical responsibility, product risk may be found in requirements stability, design performance, code complexity and test specifications. Product risk is difficult to manage because software requirements are often perceived as flexible. Requirements are the most significant product risks in risk assessments. A software risk can be classified in the following figure. In general, a risk management process should consist of the following activities, acquisition, supply, development, operation and maintenance of software products and services

- Early planning for risk management
- Implement the Risk Management
- Managing the project risk profile
- Advanced risk analysis technique
- Risk monitoring

- Risk treatment
- Effective Evaluation

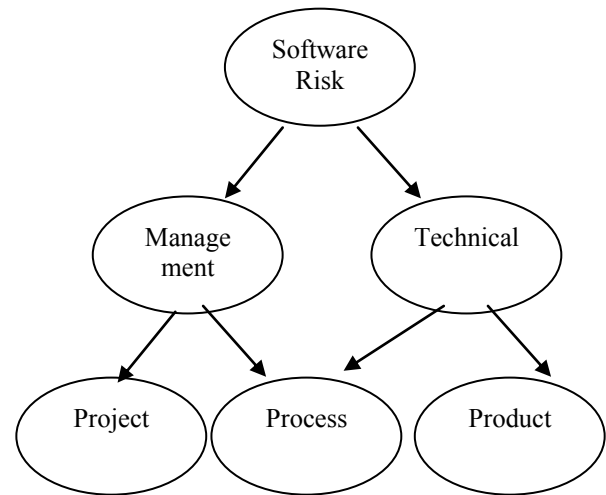


Figure2 : Software Risk classifications

### IV. RISK MANAGEMENT TOOLS

#### 1) Risk Identification

The Universal Risk Project identifies two basic types of risk statements that can be used to identify whether a set of circumstances represents a risk to the project: "IF technology is not available, Then we will not meet the requirement" and "IF we cannot hire sufficient qualified software engineers, Then We cannot meet the planned development schedule". A tool that can be used to aid in the identification of risks on a software project are strategically defined software measures. In this we propose the use of a software measurement process as an integrated approach for identifying risks before they become problems.

- Software Component Status
- Staff Experience
- Reviews Completed
- Problem Report Status
- Resource Utilization

#### 2) Risk Analysis

After risks are identified, they should be partitioned into categories such as technical, cost, schedule, management, etc. Note that some risks may fall into multiple categories. Why do risks need to be partitioned? First of all, some risks are more important than others. Also, different stakeholders may be concerned about different risks, or different personnel may bear responsibility for tracking/monitoring different risks. Finally, different risk types may require different mitigation strategies.

The initial activity in risk analysis is to identify contributing factors, then establish a hierarchy of those contributing factors. It illustrates a hierarchy of how a project might fail, given the contributing factors of Staffing, Funding, Performance Failures, and so on. The Staffing factor is further broken down to show, first, how staffing may

become a contributing factor to project failure and second, what the contributing factors might be that result in insufficient staffing (subsequently leading to project failure). All contributing factors defined within the hierarchy would be broken down to a correspondingly meaningful level of detail. Similarly, for positive risk, a hierarchy of contributing factors could also be created, this time highlighting those elements for which risk is being undertaken in order to leverage a perceived opportunity for the project, such as "Schedule Completion Will Be Early". There are any number of ways that risk can be partitioned, analyzed and quantified. The approach taken and method(s) used should always be tailored to meet the needs of the business, the customer and the project. If you reference the Tools and Methods section of this document, you can review several different MS Excel-based tools developed by the DACS from information available in the literature, each having its own partitioning scheme and offering a different approach to quantifying project risk. Once you have successfully partitioned and quantified software risk, prioritization of risk becomes the next logical activity.

### 3) Risk Priority

Risk priority is a critical characteristic of the formal risk management process, as it provides the opportunity to apply what are typically limited project resources to those risks having the largest potential impact on the project. For many risk priority approaches, risks are ranked and priority based on some combination of probability. This can be done qualitatively in a risk priority matrix using some type of composite probability-impact score. Note that the qualitative matrix has some basis in quantitative values for frequency and impact, although these can be subjectively defined and tailored to suit specific sets of circumstances. The highest priority risks would be those falling in the red region. The quantitative example of risk priority highlights the combined probability-impact scores for specific elements of the project. Of course, these are very high-level risks to try to deal with effectively. The highest priority risk is schedule failure, followed by technical failure, cost failure, etc. The MS Excel-based tools provided in the Tools and Methods section show how high-level risk areas can be further broken down to quantify and prioritize very specific elements of risk. The combined risk exposure number associated with performing regression testing indicates that it should be considered a much lower priority than if regression testing is not going to be performed at all based on the quantitative probability and loss associated with each potential undesired outcome.

### 4) Risk Management Plan

The risk management plan includes product size, business impact, customer, process, technology, development environment, staffing (size and experience), schedule, and cost. Risk management planning provides the basis for the identification of the monitoring procedures that should be put in place for each risk, including how to tell if a risk is

going to manifest as a real problem, and how frequently each identified risk should be monitored. Risk planning also takes into account risk aversion planning i.e., what actions will be taken to mitigate risk before it occurs and contingency planning i.e., how to react if a risk actually manifests. The document "Risk Management Survey of Department of the Navy Programs" contains samples of Risk Management SOW Contract Clauses, a Risk Status Report format, a product/process combination risk management process, a technical risk assessment form, a technical risk status form, a risk management process and analysis methodology, an event-driven risk mitigation report, and risk management survey references.

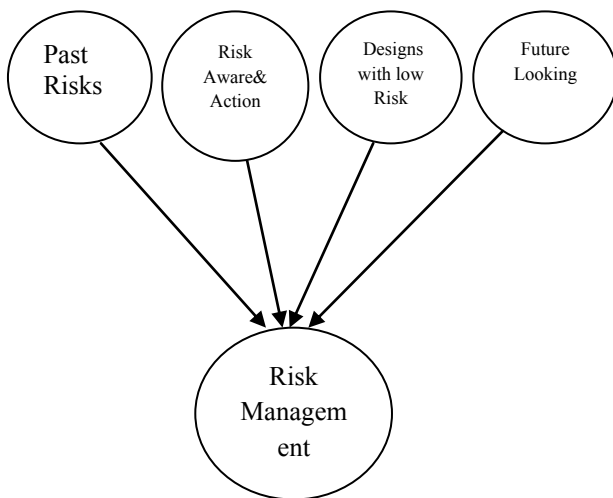
### 5) Risk Monitoring

There are a number of measurements/metrics and tools that can be used to monitor and report status of project risks. Following are the some of the basic measurements and metrics that can be considered to meet these needs.

- Number of Identified Risks
- Number of Active Risks
- Number of Risks Assessed "High", "Medium" and "Low"
- Expected Value (Probability x Cost)
- Overall Project Criticality Index

The number of identified and active risks provides a picture of how many risks have been mitigated, and how many remain as open. Prioritized risks can be categorized as the number considered "high", "medium" and "low". Combining risk probability and impact into a total composite score is another way to identify, track and monitor high priority risks. Identifying an "average" probability impact (PI) score subdivides the risk population into "greater than" and "less than" average risks, providing a very general guideline on which risks to address first. Another way to look at impact is risk probability times cost impact, rather than consequence or severity impact, in order to calculate the expected value of each potential risk. A useful metric for assessing project risk is the percent likelihood of meeting the target schedule and/or budget. Any significant negative deviation from this metric would not well for the success of the project. There are also a number of risk management audit measurements/metrics that are appropriate to monitor and track progress on controlling project risk. These include factors such as (1) the number and ratio of scheduled and actual risk management audits, (2) the effort required to support risk management audits, (3) the total number of risk-related problem reports (and the number of open and closed reports) that are measured in each risk management audit, and (4) the number and ratio of actual and deferred corrections that are reported in each risk management audit. This number of corrections can be further broken down into categories of major and minor corrections and, as with the number of actual corrections, can be assessed against the amount of effort required to implement them. One of the better tools to use in the risk management process is Risk Status Reports.

They represent a best practice in that they allow the communication of appropriate risk metrics to all stakeholders on the project. Risk Status Reports can include such things as a listing of the top ten risk items, the number of risks resolved as of the report date, the number of new risk items introduced since the last report, the total number of current unresolved risk items, the presence of unresolved risk items that lie on the project critical path, and assessment of the probable cost of unresolved risk in comparison to the amount of risk reserve. A second useful tool for monitoring risk and reporting risk status is the Event-Driven Management Report. This tool can be represented by a generic waterfall chart, where risks are identified and documented on an Integrated Product Team watch list. One general caution or observation regarding the Event-Driven Management Report, Be aware that risks may not always follow the waterfall, i.e., high- or medium-level risks could conceivably pop up anywhere in the life cycle, depending on how effective the overall risk management process is. Event-driven risks will not necessarily conveniently decrease in risk level as the life cycle evolves.



#### 6) *Internal Audit*

A software solution that includes an audit component is a paperless audit management system proven to reduce the time and cost associated with internal audit processes. It is better for mid-market and large enterprise organizations to improve audit efficiency and productivity of the entire audit process including risk assessment, planning, scheduling, preparation, review, report generation, global issue tracking and administration.

- Analyst Reports
- Events Calendar
- Early Updates
- Success Stories
- Surveys

Effective implementation of risk management is based on the following set of benefits resulting from the process:

- The risks that could impact project success are identified
- A priority order in which risks should be addressed is established
- Mitigation alternatives appropriate for each potential problem are carefully considered based on project circumstances
- Optimized mitigation techniques for all risks above their thresholds are selected
- Contingency plans in case the risk mitigates are developed proactively, rather than as a result of fire-fighting
- Information to improve risk management policies is captured, analyzed and acted upon
- Risk management processes / procedures are systematically and periodically reviewed and improved to further reduce risk

#### 7) *Safe Tracking And Implementation*

Results of the risk reduction implementation must be tracked. The tracking step involves gathering data, compiling that data into information and then reporting and analyzing that information. This includes measuring known risks and monitoring triggers as well as measuring the impacts of risk reduction activities. This includes

- Identify the new risks and add them to existing risk list.
- Information needed for additional requirements planning
- Implementing the plan

The implementation starts with the requirement specification and continues till release of software project. The major objective of this phase is to maximize the utilization and effectiveness of resources in time and within project budget in order to carry out the various activities during development.

## V. CONCLUSION

Software risk management is the integrated approach in which risk factors are systematically identified, assessed, and implemented. The determination of the risk in a software either due to external or internal causes in a major part of software management. The process of risk analysis is continuous and applies to many different levels, at once identifying system level, assigning probability impact. For a software manager it is responsibility to manage the software risk in effective manner. The risk management plan is implemented as a part of initial project planning and utilized throughout all of the phases of the software development process. Risk management requires fear free environment where risks can be identified and discussed by conducting meetings and periodical reviews. We can reduce confusion,

time, development and management cost by using an integrated approach.

## VI. REFERENCES

- 1) *Software Risk Management* by Barry W. Boehm, Les Alamos, CA, IEEE Computer Society 1989,
- 2) *Risk management for Software Projects*, Alex Down, Michael Coleman, Peter Absolon, Tata McGraw-Hill Book Company, 1994
- 3) *Practical Risk Assessment for Project Management*, by Stephen Grey, John Wiley & Sons, 1995.
- 4) *Software Engineering Risk Management: Finding Your Path Through the Jungle*, Version 1.0, Dale Karolak, IEEE Computer Society.
- 5) *Risk Management: Concepts and Guidance, 2nd edition*, by Carl Pritchard, ESI International, 2001
- 6) *Project and Program Risk Management: A Guide to Managing Project Risks and Opportunities*, Max Wideman, 1998.
- 7) *Making Hard Decisions with Decision Tools*, by Robert T. Clemen and Terence Reilly, Duxbury Press, 2001.
- 8) *Project Risk Management: Processes, Techniques and Insights*, by Chris Chapman and Stephen Ward, John Wiley & Sons, 1997.
- 9) *Stop IT project failures through risk management*, by Dan Remenyi, Butterworth Heinemann, 1999.
- 10) *Managing Risk: Critical issue for survival and success into the 21st century*, Alan Waring and A. Ian Glendon, Thompson Learning, 1998.
- 11) *Managing Risk: Methods for Software Systems Development*, Elaine Hall, Addison Wesley.
- 12) [http://acc.dau.mil/simplify/ev.php?ID=1203\\_201&ID2=DO\\_TOPIC](http://acc.dau.mil/simplify/ev.php?ID=1203_201&ID2=DO_TOPIC)