An Empirical Study of Industry Practice in Software Development Risk Management

Srikrishnan Sundararajan, Bhasi M., Pramod K.V.

Cochin University of Science & Technology, Kochi, India

Abstract - This is an empirical study of industry practice in the management of software development risks. As we know, the revolution in information technology (IT) has ushered in significant changes to software engineering practices, as well as IT business models. These happenings have resulted in changes to software risk perception and risk management. While there are numerous studies on software risk, we need a thorough investigation of contemporary Industry practices to build a software risk management model that is practically useful. Based on an investigation of 145 software projects, we propose a model for risk management applicable to non-inhouse projects. We explain the distinctive features of our model in comparison with the existing literature on software risk. The model identifies the key focus areas of software risk management in offshored or outsourced projects in order to achieve the best outcome.

Index Terms - Software Risk, Project Management, Software Engineering, Information System, Outsourcing, Offshoring

I.   INTRODUCTION

For the best outcome from a software project, a project manager needs to understand and manage the inherent risks. Though there are many definitions of software risk (Boehm 1991), in this paper we define risk as potential events that adversely affect the project outcomes (Schmidt, et al. 2001), such as cost, schedule and quality (Pressman 2010). Risks are associated with a measure of the probability of occurrence and the impact if it occurs (Boehm 1991). Activities associated with risk management are (Project Management Institute 2008) risk assessment & risk control. Risk assessment includes risk identification, analysis and prioritization. Risk control includes risk management planning, risk monitoring and risk resolution.

Today, the global software spending is estimated at a total of USD $1 Trillion (Nasscom 2012). The concept of multiple IT organizations spread across diverse cultures working in collaboration (Mohtashami, et al. 2006) has gained momentum over inhouse projects. As global IT offshoring and outsourcing steadily increases (Gartner Inc 2012), the associated risk exposure also increases considerably (Deloitte 2008).

Chaos Report of 2009 (The Standish Group 2009), finds that only 32% of the software projects are successful (i.e., delivered on time, within budget & with quality) and the trend remains the same throughout the decade from Year 2000 to 2009. However, it may be noted that, there are very few reported studies on actual software risk management practices in IT industry (Wallace and

Keil 2004) and its compliance to the standard risk management process models (Kajko-Mattsson and Nyfjord 2008) and (Benedikt 2009).

Against this background, we undertook an exploratory study of software development risk management in offshored and outsourced software projects.

II.   THE LITERATURE SURVEY

A.   Software Risk Identification and Management

The basis of this study was the literature on software risk research during 1991 to 2012. A number of research studies have been conducted in the area of software risk identification and management. Some of studies are given below.

Boehm (Boehm 1991) identified 10 top software risks, based on his software development experience. Barki (Barki, Rivard and Talbot 1993) collected data from IT practitioners through a survey. The risk factors that emerged from their study included, technological newness, application size, lack of expertise, application complexity, and organizational environment. The risk management factors included Internal Integration, User Participation, and Formal Planning. Nidamulu (Nidumolu 1995) studied the effect of project coordination on project outcome. He developed an integrative model with select risk management dimensions affecting the project outcome, (a) directly, as well as (b) indirectly through an intervening variable called residual performance risk. (Wallace 1999), (Wallace and Keil 2004) conducted a survey of Project Managers who are members of Project Management Institute (PMI) They identified 23 risk items and grouped them under the constructs, User, Development Team, Organizational Environment, Project Complexity, Project Management, and Requirements. They obtained a four factor risk management structure, viz., Customer Mandate, Scope and Requirements, Environment and Execution. Ropponen (Ropponen and Lyytinen 1998) conducted a research on how risk management practices and environmental contingencies help in addressing the software risks. Jiang (Jiang and Klein 2001) conducted a survey of PMI members and identified five risk management factors, viz., User, Institutional, Stake holder Commitment, Simplicity (of system & processes). Schmidt (Schmidt, et al. 2001) conducted a three phase Delphi study. They identified 27 risk items and grouped them under the constructs, Project Size, Application Complexity, Technology acquisition, Insufficient resources, Lack of team expertise, Lack of user support, Lack of user experience, Lack of clear role definition, and Intensity of conflicts. Addison (Addison and Vallabh 2002) investigated the risks identified by (Boehm,

1991), (Schmidt, Lyytinen, Keil, & Cule, 2001) and (Ropponen & Lyytinen, 1998) through a survey of IT project managers. They identified 12 risks and 14 risk management factors. Deephouse (Deephouse, et al. 2005) conducted an exploratory study. They developed a conceptual model relating the effectiveness of software processes to the project outcome. The software process included planning, process training, stable environment, user contact, design reviews, prototyping and cross functional teams. Product quality and time overrun were taken as project performance measures. The study also tested the planning effectiveness as a mediating variable. Project characteristics such as the staffing level, percentage of work outsourced, type of application developed were tested as control variables. (King 2008) conducted a Delphi study of "offshoring issues" by surveying about 100 people who had recently been involved in writing, reviewing or editing papers for a special issue of another journal on the IS offshoring topic. About 200 issues were preliminarily identified. The final "top twelve"-ranked list of offshoring issues are related to, but differs from the top risk factors from literature. Thomas (Thomas and Bhasi 2008) conducted a survey of IT professionals. The risk factors that emerged from their study consisted of Team Risk, Planning & execution, External Risk, User Risk and Complexity Risk. The risk management factors consisted of Project Execution, Human Resource Management, User Coordination, and Project Planning. They proposed multiple models showing the inter-relationship of risk factors, risk management factors and project outcome using structural equation modeling.

A pioneering work in the identification of software risks in *offshore outsourced projects* was done by (Charalambos, Iacovou and Nakatsu 2008), which addresses risks from customer perspective. Our study derives inspiration from the above, but is focused on vendor perspective.

Finally, let's take a quick look at the practitioner's perspective. The Software Engineering Institute (SEI) Technical Report (Carr, et al. 1993), provides a detailed list of software risk variables that affect software development in general. The capability maturity model integration developed by the CMMI Institute (CMMI 2010), and PMBOK (Project Management Institute 2008), provide a good perspective of the software process and project management, from which, one can draw a list of risks.

III. THE RESEARCH OBJECTIVES

The success of a software project is influenced by the inherent risks, risk management, and constraints that are not under the control of the project manager. The main objectives of this study are:-

In the context of the offshored or outsourced software projects,

1. What are the major software development risk factors?
2. What are the major software development risk management factors?
3. How does the Industry practice of software development risk management differ from the existing literature on software risk?

The sub objectives of the study are listed below.

[Hypothesis - H1]

Software risk rankings have changed with passage of time (1991 to 2001), due to the revolution in Information Technology.

[Hypothesis - H2]

Devolution of ownership in software project management in the forms of offshoring & outsourcing has changed the software risk profile, with the passage of time (2001 to 2012).

[Hypothesis - H3]

Software development risks in offshored or outsourced software projects are distinct compared to in-house projects, in particular from vendor perspective.

[Hypothesis - H4]

Software development risk management in offshored or outsourced software projects are distinct compared to in-house projects from vendor perspective.

IV.   THE RESEARCH DESIGN & DATA COLLECTION

A.   The Research Design

The research is exploratory in nature.   The population was software projects having software development phases such as requirements analysis, design, construction, and testing. The unit of observation was software projects. The software risk, software risk management, project constraints and project outcome were measured by collecting data from IT software professionals, who participated in an effort leading to a significant software delivery. Samples were drawn from the projects with the following constraints:-

- The duration of work was 3 or more calendar months
- The software delivery effort was 15 person months or more
- The delivery was done by a team of 5 or more members
- The project consisted of software development phases such as Requirements analysis, Design, Construction and Testing.
- The respondent had an experience of 3 or more years in IT software services
- The respondents were employed with global IT software companies
- The IT companies had matured processes in place

Questionnaire method was used for data collection. The questionnaire was posted online, using the Survey Monkey - a paid service provider that allows one to prepare a questionnaire, create an email address book and collect responses online.

B.   Preparation of the Instrument

A brief description of steps carried out for the development of the survey questionnaire is given below:-

1. A pilot (pre-test) questionnaire was developed to measure the risk factors, risk management factors and project constraints, based on literature survey on software risk during 1991 to 2012, as well as the perspectives of IT practitioners.
2. The pre-test questionnaire was subjected to 3 rounds of review by senior IT practitioners & academicians to improve

its *content validity*, while maintaining brevity in the description and number of questions

3. The pilot questionnaire addressed 127 questions.
4. 22 responses were obtained from the pilot survey.
5. The pilot responses were thoroughly analyzed. The questions that 30% or more respondents did not answer were deleted. We did a factor analysis to determine the variance explained by the items underlying the questions. We identified trivial items (the items with Eigen value less than 1, items that showed heavy cross loading (more than 0.5 loading for two or more factors), items with a loading of less than 0.3. The questions corresponding to these items were deleted. We also analyzed the respondent's comments and discussed the same with senior IT professionals. Based on all the above, the questionnaire was revised - some questions were deleted, merged, split or simply reworded.
6. A seven point scale was used in the pilot (strongly agree, agree, some-what agree, not applicable, some-what agree, disagree, and strongly disagree). But none of the responses carried a value of '1' (strongly disagree), which indicated that each risk item and risk management item was either present in some measure or else, it was not applicable. Based on discussions with senior IT practitioners, the scale was changed to 6 points (strongly agree, agree, some-what agree, not applicable, some-what agree, and disagree).

C. The Final Survey

The final instrument consisted of 78 questions - 17 representing software risk, 44 representing software risk management, 11 on project constraints, and 6 on project outcome. Each respondent was required to rate the software risk as well as risk management strategy, adopted in the project using a six point scale (*See Section IV.B.6*). Project outcome was measured in terms of time overrun (additional effort in person months), schedule slippage (delay of delivery in calendar days) and adherence to quality (the number of defects). Project constraints included the type of project, technology platform, company name, work location, project duration, project effort, team size, onsite/offshore ratio of the team, need for niche skills, the role of the respondent in the project, and the number of years experience in IT.

Non disclosure (of personal information) agreement was stated in the first page of the online survey questionnaire as well as the email communication to the respondents. The questionnaire did not have any questions that revealed the identity of the respondent, project or client.

The questionnaire was posted online, using the Survey Monkey (*See Section IV.A*). We collected the email addresses of 418 IT practitioners. We started with the support of colleagues / former colleagues, leveraging the first author's experience of 22 years in IT industry and requested them in turn to forward the survey request to their colleagues. This is called snow ball method. We received 179 responses over a period of 12 months, out of which 169 were complete (saved and submitted). In spite of best efforts, we found it difficult to locate multiple persons belonging to a project. We initially requested the respondents to forward the survey to two other team members who worked in the same project; but they were reluctant to do so, for fear of being identified. In order to maintain confidentiality, of the response as

well as respondent's identity, we decided to collect only one response per project.

V. DATA ANALYSIS
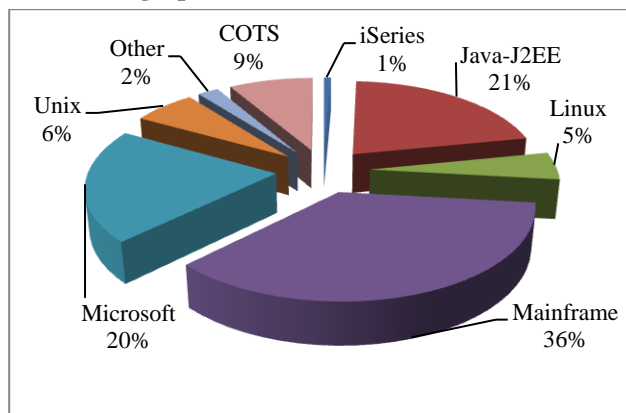
A. Demographics

Fig. 1: Responses by Technology Platform

The respondents belonged to Global IT companies that included Accenture, CSC, CTS, HCL Technologies Ltd., IBM Global Services, Infosys, L&T Infotech, MphasiS, TCS, and Wipro, from among others. These companies had organizational process maturity of CMM Level 4 and above. The average experience of a respondents was 12 years, out of which 53% were Project Managers, 28% Team Leads and 19% Architects. The average team size was 19. The average project duration was 18 months. Distribution of responses, by technology platform and by project category is shown in *Fig.1* and *Fig.2* respectively. The projects ranged from pure onsite (13%) to pure offshore (12%). The ratio of the team at customer site versus offshore is shown diagrammatically in *Fig.3*
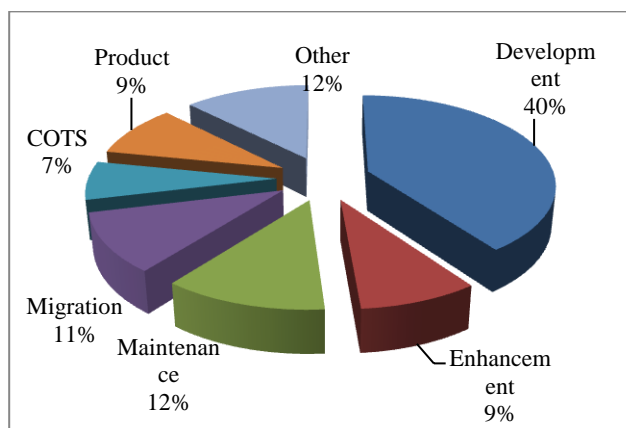

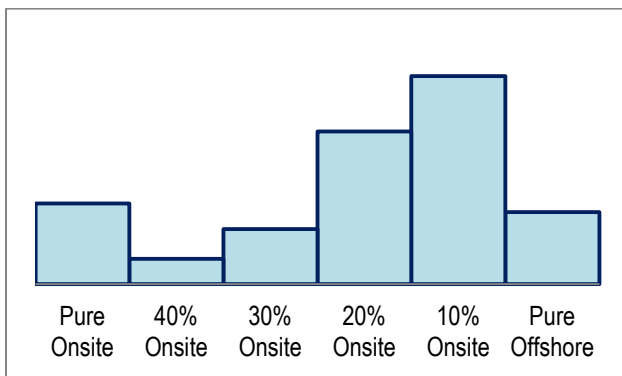**Fig. 2: Responses by project category**

Fig. 3: Onsite versus Offshore Team Ratio

B. Preparation of Data for Analysis

Data Analysis was done according to (Hair, et al. 2006). SPSS software was used for the analysis of responses. To ensure true representation of the responses, imputation of missing values was not carried out.   Cases (responses) with missing data for dependent variables were discarded. Cases with more 30% missing data were discarded. Multivariate analysis was done and outlier cases with Mahalanobis (D^2/df) measure > 2 (with respect to the risk variables) were discarded.

Factor extraction method was Principal Component Analysis with Varimax rotation applied using Kaiser Normalization. Instances of variables with missing values were excluded (Imputation of data was not attempted, since that can bias the observation) Orthogonal rotation was applied. The items with Eigen value less than 1 were deleted. The items that load less than 0.4 were deleted. It may be noted that, one risk item & one risk management item were retained, though they loaded slightly less than 04 – this is discussed in the subsequent sections. The items that showed high cross loading were deleted. The structure was re-specified iteratively with the deletion of each item.

C. Risk Factors

The procedure in the *Section V.B* was used for the preparation of data for factor analysis of risk variables. The risk factor structure that emerged, explained 71.3% of the variance (*See Table-1*). The risk items that constitute these risk factors are listed and discussed under *Hypothesis – H3*.

One of the Risk Item#1 *'Getting SME Time'* loaded only 0.36 (less than 0.4). This is a challenge in projects, where client dependency exists, according to senior IT practitioners. Hence the risk item was retained. The internal consistency estimated using Cronbach's alpha (α) was 0.73 (an alpha value of 0.70 or above indicates good internal consistency). The Kaiser-Meyer-Oaklin measure of sampling adequacy (KMO value) obtained was 0.72 (within the acceptable range of 0.6 to 0.9). Bartlett Test of Sphericity was significant (0.00), indicating that sufficient correlations exist among variables.

Table 1: Software Risk Factors

Description of the Factor	% Variance Explained
Understanding the Requirements	11.59
Knowledge Transfer	14.01
Allocation of Competent Resources	21.94

Resource Availability	10.21
Quality Management	3.99
Cultural Dimensions	9.52

D. Risk Management Factors

The procedure in the *Section V.B* was used for the preparation of data for factor analysis of risk management variables. The risk management factor structure that emerged, explained 66% of the variance (*See Table-2*). The risk management items that constitute these risk management factors are listed and discussed under *Hypothesis – H3*.

One of the risk management items, 'PM Involvement in Estimation' loaded only 0.39 (less than 0.4). According to senior IT practitioners, this is an important risk management item, though in some projects Project Manager was identified only after effort estimation. Hence the risk item was retained. The internal consistency estimated using Cronbach's alpha (α) was 0.93 (an alpha value of 0.70 or above indicates good internal consistency). The Kaiser-Meyer-Oaklin measure of sampling adequacy (KMO value) obtained was 0.83 (within the acceptable range of 0.6 to 0.9). Bartlett Test of Sphericity was significant (0.00), indicating that sufficient correlations exist among the variables. The internal consistency estimated using Cronbach's alpha was 0.93.

Table 2: Software Risk Management Factors

Description of the Factor	% Variance Explained
Understanding the Requirements	4.4
Solution	5.6
Project Planning	10.3
Knowledge Management	10.0
Employee Motivation	13.5
Quality of Build	6.8
Quality of Testing	8.8
Change Management	6.5

VI.   DISCUSSION

A.   Software Risk Factors

[Hypothesis - H1]

> Software risk rankings have changed with passage of time (1991 to 2001), due to the revolution in Information Technology.

A number of research studies have been conducted in the area of software risk identification (*See Section II*). The findings from three major chronologically representative research studies, viz., (Boehm 1991), (Schmidt, et al. 2001), and (Charalambos, Iacovou and Nakatsu 2008) as well as the findings from this study (2012) are provided in *Table-3*. An entry of 'Yes' in a cell indicates that the risk is considered as a top software Risk; 'No' or 'N/A' indicates it is not considered as a top risk, in the respective study.

From the *Table-3*, we see that 4 out of 10 risks (*Item# 12, 14, 15, 16*) in Boehm's list *1991(B)* do not appear in other studies, viz., *2001(B), 2008(C) and 2012(O)*. Risks related to computing

capabilities (*Refer to Item# 15, 16*) are critical in scientific or real time applications. But the revolution in technology (hardware, software, and methodologies) has made the above risks less significant in the context of general business information systems. Moreover, the pervasive application of IT has resulted in process maturity in software development and vendor management (CMMI 2010). Therefore, *Item# 12, 13 and 14* are not top priority risks anymore.

Hence we find that the software risk rankings have changed with passage of time, due to revolution in Information Technology and the consequent maturity of software engineering processes.

[Hypothesis - H2]

Devolution of ownership in software project management in the forms of offshoring & outsourcing has changed the software risk profile, with the passage of time (2001 to 2012).

Table 3: Software Risk Rankings over Time

Ite m#	Risk Description	1991 (B)	2001 (S)	2008 (C)	2012 (O)
1	Lack of Top management Commitment	No	Yes	Yes	N/A
2	Failure to gain user commitment	No	Yes	N/A	N/A
3	Misunderstanding the requirements	Yes	Yes	Yes	Yes
4	Lack of adequate user involvement	Yes	Yes	Yes	Yes
5	Lack of required knowledge / skill	Yes	Yes	Yes	Yes
6	Lack of frozen requirements	Yes	Yes	Yes	Yes
7	Changing Scope / objectives	Yes	Yes	Yes	Yes
8	Introduction of new technology	No	Yes	N/A	No
9	Failure to manage end user expectations	Yes	Yes	Yes	N/A
10	Insufficient / inappropriate staffing	No	Yes	N/A	Yes
11	Conflict between user departments	No	Yes	N/A	N/A
12	Gold Plating	Yes	No	N/A	No
13	Shortfalls in externally-performed tasks	Yes	No	Yes	No
14	Shortfalls in externally-furnished components:	Yes	No	N/A	No
15	Real-time performance shortfalls	Yes	No	N/A	No
16	Straining computer capabilities	Yes	No	N/A	No

From *Table-3,* we find that, *Risk Items 1, 2, 8, 10 & 11* were new additions by Schmidt *2001(S)* as top risks compared to Boehm's list *1991(B)*. *Items 1, 2, 10, and 11* are stake holder related risks in inhouse project management (viz., user, departments, employees, and top management). *Item# 8* is risk related to introduction of new technology. This is a challenge in inhouse

projects. But technology capabilities form the core competency of global IT companies and hence not a major challenge in offshoring & outsourcing. From *Table-3*, it can be seen that the five risks stated above do not appear in the study of offshoring and outsourcing risks by Charalambos *2008(C)* as well as our study *2012(O)*.

Moreover, the risk profile of Charalambos (Charalambos, Iacovou and Nakatsu 2008) consists of many more risk items that are not in the lists of (Boehm 1991) and Schmidt (Schmidt, et al. 2001). They are listed below.

1. Language barriers in project communications
2. Lack of offshore project management know-how by customer
3. Failure to consider all costs
4. Telecommunications and infrastructure issues
5. Vendor viability
6. Difficulties in ongoing support and maintenance
7. Cross-cultural differences
8. High turnover of vendor employees
9. Constraints due to time-zone differences
10. Lack of continuous face to face interaction across team members
11. Threats to the security of information resources
12. Negative impact on employee morale
13. Lack of familiarity with international and foreign contract law
14. Differences in development methodology/processes
15. Political instability in offshore destinations
16. Negative impact on image of customer organization
17. Currency fluctuations

All the above risks are primarily related to Offshoring or Outsourcing and were not considered significant in earlier research studies, where the focus was risks related to in-house projects. Therefore we infer that the devolution of ownership in software project management in the forms of offshoring and outsourcing has changed the software risk profile, with the passage of time (2001 to 2012).

[Hypothesis - H3]

Software development risks in offshored or outsourced software projects are distinct compared to in-house projects, in particular from vendor perspective.

The software risk factors that emerged from this study are shown in *Table-1*. This is diagrammatically depicted in *Fig.4*. Each factor is discussed below. The risk items underlying a factor are listed; followed by explanations obtained through discussions with some of the senior IT professionals who participated in the survey. A brief theoretical explanation is provided where required.

    i.   *Understanding the Requirements*

This factor includes:-

~   Competency of the team in analyzing business requirements
~   Competency of the team in translating business requirements to software implementation

Though, formal as well as informal methods have evolved, requirements engineering remains relatively informal and challenging at the IT industry level (Pressman 2010). Hence the understanding & analysis of business requirements and mapping them to application code is a an area of challenge in all software projects

ii. Knowledge Transfer

This factor includes:-

~ Written requirements specification was available
~ Knowledge transfer from customer/ business to project team was effective
~ Adequate documentation of the system, interfaces, and business functionality was available

Knowledge transfer includes systematic and documented (formal) communication as well as informal communication (Nidumolu 1995).

Usually, the subject matter experts in the customer team provide business requirements dissemination, initial application knowledge transfer along with available documentation. This is further fortified from system study, observation and job experience. In the later phases of the project, subject matter experts from both business and IT side will be available as guides and mentors for the team. In maintenance projects changes are made to existing software application; hence the level of documentation available on application functionality is very critical.
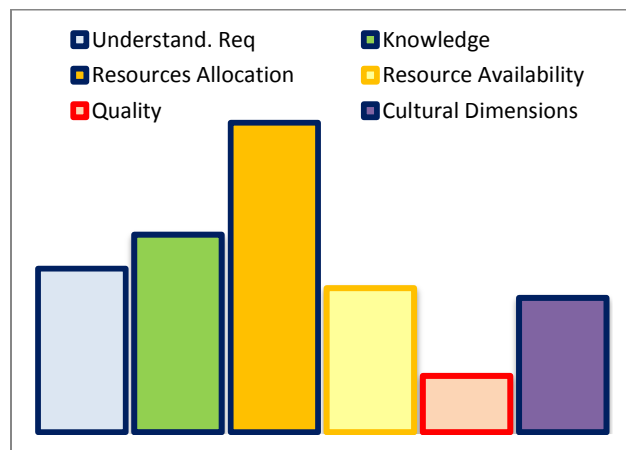

Fig. 4 Software Risk Profile

iii. Allocation of Competent Resources

This factor includes:-

~ The project manager's competency in the resource planning
~ Allocation of key (core) resources at the right time
~ Allocation of all required resources at the right time
~ Competency of the resources against the project requirements
~ Getting user / business / experts time for explanation, clarification, guidance and check point reviews

Project Manager must be competent in understanding the overall requirements; effort estimation, project plan and resource loading

plan. A small team capable of supporting the PM in the above activities needs to be allocated at the onset (project initiation). The number and capabilities of this "core" team varies depending on the complexity of the project. Nevertheless, the core team is expected to have some of the competencies such as, requirements analysis, estimation, devising the solution, selection of resources, and setting up development environment & processes, to mention a few. All the resource requirements of the project must be satisfied as per the resource loading schedule. The team must, together, satisfy the competency requirements of the project. A job description typically involves primary and secondary technical skills. In maintenance projects, prior knowledge of the application is important. Once a working team is in place, each individual's level of competency can be improved through appropriate training program. Finally, plans must be made for making subject matter experts available for knowledge sharing, reviews and guidance.

Getting competent resources allocated in time to a project, emerged as the biggest risk (challenge) in this study.

iv. Availability of Resources

This factor includes:-

~ Employee turnover
~ Employee release
~ Unavailability employees due to vacation or absence
~ Presence of Transient Resources

Turnover or attrition is a major challenge for an IT company – unspecified reports put the figure at a moderate 15%. Planned vacations and unplanned absence also affect employee availability. Projects also happen to be the training grounds for technical skills / competency improvement, by design or by chance. In addition, experts in the application and business functionality, aka, subject matter experts, must be available as required for knowledge transfer and guidance, at least as per a predetermined schedule.

v. Quality Control

Quality Control (QC) includes appraisal activities such as reviews and inspection.

Adequate time & due diligence is needed from the early stages of requirements gathering down to the user acceptance testing, in order to deliver the expected functionality. But the investment in QC activities is always a trade-off between the available budget and cost of quality. Hence the Project Manager must exercise extreme diligence in formulating and adopting appropriate quality processes, procedures, tools/ techniques, standards & guidelines.

vi. Cultural dimensions

This factor includes:-

~ Cultural diversity of the team
~ Cultural dimensions of project leadership

Cultural dimensions play an important role in distributed project execution and in any team where cultural diversity exists. Cultural dimensions (Hofstede 1976) such as Power Distance,

Individualism, Masculinity, Uncertainty Avoidance and Long Term Orientation affect communication and control. Practical implication of cultural dimensions include, challenges in requirements elicitation and knowledge transfer, fostering individual initiatives, individual efficiency, team participation (in decision making, planning and coordination), information sharing, and so on, to mention a few. All these factors influence the efficiency & productivity of the team. Cultural dimensions determine leadership style. The leadership style adopted has a corresponding influence on the risks stated above.

CONCLUSION

The software risk profile that emerged from this study is shown in *Table-1*. The risks that we identified are not altogether new to existing literature. But the priorities are different, with respect to the inhouse risks that emerged from the various research studies mentioned in *Section II*. We also took a closer look at the differences with respect to three research studies, under *Hypothesis - H1 and Hypothesis – H2*. Therefore, the survey findings indicate that risk profile in offshored or outsourced software projects are distinct compared to inhouse projects.

In this study, we profiled software risks from vendor perspective. Refer to the risks identified by Charalambos – part of the risks appear under *Column 2012(C)* in *Table-3*; the rest are listed in the discussion on *Hypothesis – H2*. Charalambos identified the risk profile of offshore outsourced projects from customer perspective. Many of the risks apply to both customer and vendor sides, but the priorities differ *(refer to our detailed discussion on risks identified in our study, earlier in this section)*. Therefore, some of the top risks identified in our study do not appear in Charalambos's list – viz., getting subject matter expert's time, transient resources, knowledge transfer, allocation of resources, application documentation, the time allotted for quality control, and project manager's competency. In fact, allocation of competent resources in-time is considered as the top challenge (risk) by vendor project manager, whereas they do not appear in Charalambos's list. Hence we conclude that the vendor perspective of risk profile that emerged from our study is different from the customer perspective identified by Charalambos.

A theoretical explanation is provided here. Offshoring / outsourcing magnify certain risks and create new threats (Aspray, Mayadas and Vardi 2006) and (Charalambos, Iacovou and Nakatsu 2008). Whereas business knowledge is usually a core competency of the business (Mclvor 2005), the competencies of the IT team becomes more & more prominent as we progress through the technology intensive downstream stages such as design, construction and testing. With IT offshoring, and outsourcing becoming a part of the well formulated business strategy, global IT companies are steadily improving their competitive edge in technology and process capabilities (Peppard and Ward 2004).

Therefore, our findings indicate that risk profile in offshored or outsourced software projects are distinct compared to inhouse projects. Moreover, the vendor perspective of risk profile, as identified in this study, is different from customer perspective as identified by Charalambos (Charalambos, Iacovou and Nakatsu

2008). The following are considered as top risks in industry practice from vendor side - understanding the requirements, knowledge transfer, timely allocation of resources & competency of resources, resource availability, quality control, and cultural dimensions. This is a unique risk profile applicable to offshore and outsourced software projects from vendor perspective.

B. Software Risk Management Factors

[Hypothesis - H4]

Software development risk management in offshored or outsourced software projects are distinct compared to in-house projects from vendor perspective.

The software risk management factors that emerged from this study are shown in *Table-2*. The factors are diagrammatically illustrated *in Fig.5*. Each factor is discussed below. The risk management items underlying a factor are listed; followed by explanations obtained through discussions with some of the senior IT professionals who participated in the survey. A brief theoretical explanation is provided where required.
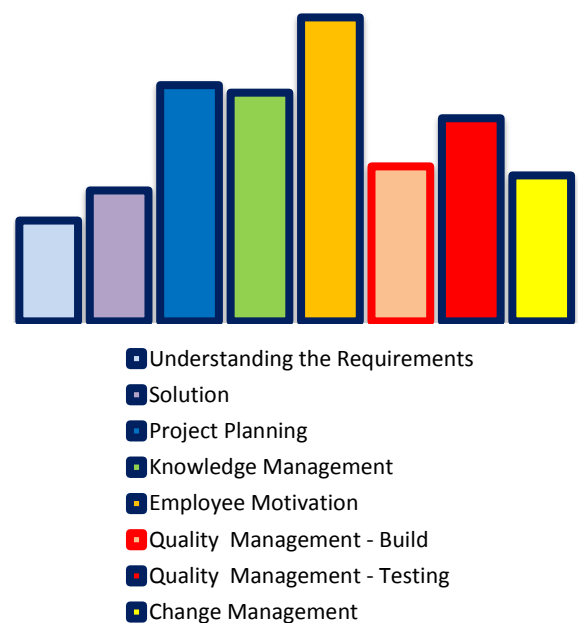

- Understanding the Requirements
- Solution
- Project Planning
- Knowledge Management
- Employee Motivation
- Quality Management - Build
- Quality Management - Testing
- Change Management

Fig. 5: Risk Management Profile

i.   The understanding of requirements

This factor includes:-

~   Adequate involvement of project manager in the estimation process
~   Discussions with users, customer subject matter experts, and business analysts to elicit and understand the requirements
~   Ensuring the availability of users, customer subject matter experts, and business analyst for reviews, clarifications, and guidance, in every phase of the project as required.

Adequate understanding of the business requirements must be obtained, in order to translate them to software specifications that defines functional, domain and technical requirements

ii.   The solution

This factor includes:-

~   The effectiveness of solution (problem solving approach and techniques)
~   The effectiveness of tools used

There are many projects that have run into extra costs and considerable delay due to in-appropriateness in the solution devised (e.g. reengineering versus replacement), tools used (e.g. impact analysis tools) and techniques deployed (design patterns). Hence the above factors need due deliberation & diligence.

iii.   Project planning

This factor includes:-

~   Use of formal estimation methods, estimation tools, organizational level estimation procedure, and historical database of projects
~   Plans to ensure tracking of each user requirement from requirements specification phase, through design, construction, testing, and up to delivery
~   Metrics collection and analysis

Estimation needs to be accurate. The project scope, technology solution, and the estimates are the three major inputs for preparing the project plan. Project plan encompasses all activities directed towards delivering all the requirements within budget and time. Therefore, a Project plan contains a detailed account of deliverables, schedules, phases, activities, work items, measurements, service level agreements and other aspects. Plans must be made for collecting and analyzing the project & process metrics to identify the cause & source of undesirable deviations and devise plans for improvement in productivity & quality.

In order to achieve continuous improvement, a framework called DMAIC (Define, Measure, Analyze, Improve & Control is deployed (Mahanti and Antony 2009), (Peppard and Ward 2004), and (Madhani 2008).

iv.   Knowledge management

This factor includes:-

~   Adequate training to offshore team in the application functionality, before assigning tasks
~   Systematic Induction program
~   Effective on-the-job training program
~   Adequate attention in ramping up trainees allocated to the project
~   Mitigation plan for employee turnover, release, and unavailability

According to knowledge based theory, management of explicit knowledge and tacit knowledge are fundamental to the success of in-sourcing and outsourcing of non core competencies (Mclvor 2005), (Cha, Pingry and Thatcher 2008), (Gefen 2008), (Dibbern, Winkler and Heinzi 2008), (Leonardi and Bailey 2008).

Knowledge management includes the capture & transfer of knowledge that is useful to the project in the short term and the engagement in long term. Each & every individual allocated to a project may not possess the adequate skills and experience; but the project team as a whole must be competent to execute the project. With this objective in mind, the team needs to be ramped up to the required competency level. Plans for this include, on the job training, induction programs, workshops, online or classroom trainings, and relevant certification courses. In the case of trainees, ramp up programs are aimed at imparting technical skills as well as project specific knowledge. Employee turnover and non availability of employees arising from planned or unplanned vacation, creates a knowledge gap. The mitigation strategies for these, include maintaining a buffer (allocating additional resources), cross training (imparting skills required in a different project or module), job work books and so on.

v.   Employee Motivation

This factor includes:-

~   Mapping employee profile and, their career interests
~   Employee goal setting and periodic review
~   Objectiveness of employee appraisal
~   Employee performance versus rewards & recognition
~   Employee performance versus compensation & benefits
~   Onsite assignments versus employee performance & organizational policy
~   Employee work life balance
~   Recognition of individual initiatives and opinion

According to the resource based view of the firm (Mclvor 2005), valuable, rare and non imitable resources form the basis of competitive advantage of a firm. IT industry is human resource centric. IT companies are most concerned about human resource management (Madhani 2008), (Peppard and Ward 2004), since the technical skills and cost competency of resources determine their competitive advantage.

Measuring employee performance and deployment of an organization wide policy are complex tasks. This is since each and every project is unique, thereby rendering objectiveness & standardization inaccurate, though guidelines can be formulated. Some companies have devised 360 degree feedback to supplement the evaluation by supervisors, where the achievements of relatively silent members, gets highlighted from the feedback of colleagues. Measures to improve the employee work life balance include, working from home or nearby locations, company transport service, flexible work time, food court, health & recreation, help desk services for personal services, family healthcare, child care etc. Other risk items listed in this section are self explanatory.

vi.   Quality management – Build

This factor includes:-

~   Requirements analysis
~   Design
~   Construction

The quality processes translate resource skills into organizational competency & capability (Madhani 2008), (Mahanti and Antony 2009).

Quality control includes inspections, reviews, walkthroughs etc. Quality assurance includes training the team in quality processes, quality auditing etc. Global IT companies are certified in quality process frameworks – e.g. CMM (CMMI 2010). They have their

own organization wide quality process framework that is tailored for the specific needs of a project. The PM must ensure optimal allocation time and resources for quality assurance & quality control activities, within the constraints of available budget.

vii.   Quality management – Test

This factor includes:-
~   System test, esp. including Performance test
~   Integration test, esp. including Regression Test and Retrofit test.

System test includes tests for availability, continuity, recovery, security, and stress. Performance test evaluates the processing time and other aspects such as memory, and input – output operations. Regression test is an essential part of integration test. This is done to ensure that the software project delivery under consideration integrates with the existing application without impairing the current functionality. Retrofit test is about incorporating changes that have been made to production code, while the software project in consideration was in progress. User acceptance test was not included in the survey since that was done under the supervision of the customer and information about that was not available with most of the respondents.

viii.   Change management

This factor includes:-
~   Analysis of the impact of changes
~   Negotiation & agreement on scope & schedule
~   Scheduling a group of changes together

Changes in requirements are inevitable due to the tacit nature of knowledge. The user perception, the perception of the team members who received the knowledge, as well as the formal user requirement documentation may all have inherent assumptions, omissions and errors. Hence formalized procedures for change management are one of the usual features of IT project management.   The major concerns in change management include understanding and analyzing the impact of changes (de Souza and Redmiles 2008), (Jones 2007) on the effort, schedule, and quality; plans to mitigate the cascading effect of changes into upstream activities and phases; and scheduling changes together (rather than changing artifacts on ad-hoc basis) etc, from among other aspects. Due diligence in requirements gathering, and customer involvement in the development process (e.g. interim deliveries; periodic reviews & feedback etc.) help to reduce changes and rework

Conclusion

The software risk management profile that emerged from this study (See Table-2) is different from that of in house projects, discussed in Section II.B. Therefore, the survey findings indicate that the top risks in offshored or outsourced software projects are distinct compared to inhouse projects.

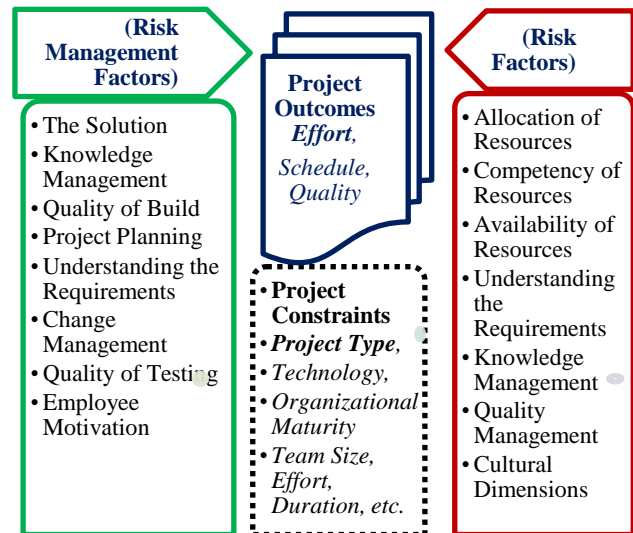### C.   Software Risk Management Model

Fig. 6: Software Risk Management Model for Offshored/ Outsourced Projects

Based on our discussions we propose the model shown in *Fig.6*, for the management of software development risk by vendor team, in the context of offshored or outsourced software projects. We have discussed Risk Factors & Risk Management Factors in this paper. Project constraints; and the relation of risk and risk management factors with the project outcome are not discussed in this paper.

VII.   SUMMARY

D.   Validity and Limitations

The study has limitations that are applicable to all survey based studies.

1.   Samples were drawn only from global IT companies with process maturity of CMMi Level-5 (CMMI, 2010).
2.   The focus of the study was business application projects. Other domains such as, scientific applications, product development, and real time systems, to mention a few, were not considered.
3.   The focus of this study was software projects, post contract signoff.
4.   The study was based on one response each, received from each project. In spite of best efforts, triangulation of observation was not possible, since we were not able to locate more than one team member who worked for a specific project (See Section IV.C).

However, we have strictly followed the guidelines for survey based research. Our questionnaire exhibited content validity (See Section IV.B) and internal consistency / reliability (see Section IV.C and IV.D). Many researchers have used similar methods in software engineering (See Section II). We explained our model using industry best practices compiled from the insights obtained from another work. We also provided a brief theoretical explanation where required, correlating the Industry practice with theory. Therefore, we believe that our findings add value to the body of knowledge on software risk management.

VIII. CONCLUSION & RECOMMENDATION

There are very few published reports on industry practice in software risk management.

This is a study of industry practice in the management of software development risks, in the context of offshored and outsourced projects, from vendor perspective. Hence this study is unique.

Software risks and their rankings have changed over time with (a) the revolution in information technology in terms of hardware and software, (b) evolution of software engineering processes and (c) the devolution of ownership in software project management. The software risks, risk rankings, and risk management, in the context of offshored or outsourced software projects differ from firm owned, or in-sourced software development.

Though generalization of software risk management is a very valuable area of research, research on current IT models, such as offshoring, outsourcing and distributed project management would add value to IT practice in a significant way. Based on this study of 145 software projects executed by global IT companies, we proposed a software risk management model (*See Fig.6*). The constituent software risk factors and software risk management factors care shown in *Table- 1* and *Table-2*. The risk factors and the risk management factors that emerged from this study are clearly distinct, in comparison with the existing literature on software risk.

The model proposes key focus areas for software development risk management in offshored and outsourced projects, to achieve the best outcome.

E. Directions for future research

The risk management model proposed in this study needs further validation, through in depth case studies on Industry practices. Each risk management factor needs further investigation. The variation in software risk management practice across different categories of projects, needs to be explored, e.g., Software Maintenance, Migration, Modernization, COTS etc, to mention a few. The study may be extended to distributed project management models (where collaboration assumes more importance).

REFERENCES

[1] Addison, T., and S. Vallabh. "Controlling software project risks – an empirical study of methods used by experienced project managers." *Proceedings of SAICSIT*. 2002. 128-140.

[2] Aspray, W., F. Mayadas, and MY. Vardi. *Globalization and Offshoring of Software: A Report of the ACM Job Migration Task Force.* NY: ACM, 2006.

[3] Barki, H., S. Rivard, and J. Talbot. "Toward an Assessment of Software Development Risk." *Journal of Management Information Systems* 10, no. 2 (1993): 203-225.

[4] Benedikt, M. "Why Risk Management Matters In It Outsourcing – A Systematic Literature Review And Elements of A Research Agenda." *17th European Conference on Information Systems, ECIS2009-0652.R1.* 2009.

[5] Boehm, B. "Software risk management: Principles and practices." *IEEE Software* 8, no. 1 (1991).

[6] Carr, MJ., SL. Konda, I. Monarch, FC. Ulrich, and CF. Walker. "Taxonomy-based risk identification." Technical Report, Software Engineering Institute, Carnegie Mellon University, 1993.

[7] Cha, HS., DE. Pingry, and ME. Thatcher. "Managing the knowledge supply chain." *MIS Quarterly*, 2008: pp. 281-306.

[8] Charalambos, L., Iacovou, and R., Nakatsu. "A Risk Profile Of Offshore-Outsourced Development Projects." *Communications of the ACM* 51, no. 6 (2008).

[9] CMMI. *Solutions.* 2010. http://cmmiinstitute.com/cmmi-solutions/ (accessed Jan 15, 2013).

[10] de Souza, CRB., and DF. Redmiles. "An empirical study of software developers' management of dependencies and changes." *ICSE International Conference on Software Engineering.* ACM New York, NY, USA, 2008. 241-250.

[11] Deephouse, C., T. Mukhopadhyay, DR. Goldenson, and Marc I. Kellner. "Software Processes and Project Performance." *Journal of Management Information Systems* 12, no. 3 (2005): 185-203.

[12] Deloitte. "The Risk Intelligent Approach to Outsourcing and Off shoring." *Risk Intelligence Series*, no. 8 (2008).

[13] Dibbern, J., J. Winkler, and A. Heinzi. "Explaining variations in client extra costs between software projects offshored to India." *MIS Quarterly* 32, no. 2 (2008): 333-366.

[14] Gartner Inc. *Gartner Worldwide IT Spending Forecast - Overall Enterprise Spending Growth Down Across Verticals.* January 2012. http://www.gartner.com/technology/research/it-spending-forecast/ (accessed May 4, 2012).

[15] Gefen, D. "Business Familiarity As Risk Mitigation In Software Development Outsourcing Contracts." *MIS Quarterly*, 2008: pp. 531-551.

[16] Hair, JF., WC. Black, BJ. Babin, and RE.,Tatham,RL. Anderson. "Mutivariate Data Analysis." Pearson, 2006.

[17] Hofstede, G. 1976. www.geert-hofstede.com (accessed May 05, 2011).

[18] Jiang, JJ., and G. Klein. "Software project risks & development focus." *Project Management Journal* (PMI) 32, no. 1 (2001).

[19] Jones, C. *Estimating Software Costs.* 2. Tata McGrawHill, 2007.

[20] Kajko-Mattsson, M., and J. Nyfjord. "State of Software Risk Management Practice." *IAENG, International Journal of Computer Science* 35:4 (2008).

[21] King, WR. "IT strategy and innovation, Issues in IS offshoring." *Information Systems Management*, November 2008: 287–289.

[22] Leonardi, PM., and DE. Bailey. "Transformational technologies & the creation of new work practices." *MIS Quarterly*, 2008.

[23] Madhani, PM. "Indian Software Success Story: A Resource-Based View of Competitive Advantages." *The Icfaian Journal of Management Research* VII, no. 8 (2008).

[24] Mahanti, R., and J. Antony. "Six Sigma in the Indian software industry: some observations and results from a pilot survey." *The TQM Journal* Volume 21, no. 6 (2009): 549-635.

[25] Mclvor, R. In *The Outsourcing Process*, 188-190. Cambridge University Press, 2005.

[26] Mohtashami, M., T. Marlowe, V. Kirova, and FP. Deek. "Risk management for collaborative software development." *Information Systems Management* 23, no. 4 (2006).

[27] Nasscom. *Resource Center - Global Sourcing.* 2012. http://www.nasscom.in/global-sourcing (accessed 09 23, 2012).

[28] Nidumolu, S. "The Effect of Coordination and Uncertainty on Software Project Performance: Residual Performance Risk as an Intervening Variable." *Information Systems Research*, 1995: 191-219.

[29] Peppard, J., and J. Ward. "Beyond strategic information systems: towards an IS capability." *Journal of Strategic Information Systems* 13 (2004): 167–194.

[30] Pressman, R. "Risk Management." Chap. 28 in *Software Engineering: A Practitioner's Approach*. McGrawHill, 2010.

[31] Project Management Institute. *A Guide to the Project Management Body of Knowledge (PMBoK).* 4. Project Management Institute, 2008.

[32] Ropponen, J., and K. Lyytinen. "Components of Software Development Risk: How to Address Them?" *IEEE Transactions on Software Engineering* 26, no. 2 (1998): 98-111.

[33] Schmidt, R., K. Lyytinen, M. Keil, and P. Cule. "Identifying software project risks: An international Delphi study." *Journal of Management Information Systems* Vol.17, No.4 (2001): pp.5-36.

[34] The Standish Group. *Chaos.* 2009. www.standishgroup.com (accessed January 15, 2012).

[35] Thomas, S., and M. Bhasi. "A Study on Software Development Project Risk, Risk Management, Project Outcomes and their Inter-Relationship." *Dyuthi.* 2008. http://dyuthi.cusat.ac.in/ (accessed March 25, 2013).

[36] Wallace, L. "The development of an instrument to measure software project risk." Doctoral Dissertation, Georgia State University, 1999.

[37] Wallace, L., and M. Keil. "Software project risks and their effect on outcomes." (Communications of the ACM) Vol. 47, no. No.4 (2004).

ABOUT THE AUTHORS

First Author – Srikrishnan Sundararajan, Master of Technology (Computer Science & data Processing), Department of Computer Applications, Cochin University of Science & Technology, Kochi, India, sskrishnan02@yahoo.com

Srikrishnan Sundararajan received Bachelor of Science degree in Electrical Engineering, in 1983; and Master of Technology in Computer Science and Data Processing in 1993 from Indian Institute Technology, Kharagpur. He is currently a Research Scholar in the Cochin University of Science & Technology, Kochi, India. He was employed as Group Project Manager in HCL Technologies; Professor of IT in SCMS-Cochin; Principal Architect in UST-Global; Delivery Manager in CSC; Associate Consultant in TCS. He is certified ITIL2012, PMP2003, and CQA1998. . He is a member of PMI, ACM, and IEEE. He is currently pursuing research in Software Risk. His area of interest includes Software Risk and Project Management.

Second Author – PhD., Bhasi, M., School of Management Studies, Cochin University of Science & Technology, Kochi, India, drbhasi@yahoo.com

M Bhasi received Bachelor of Technology degree in Production in 1986; and Master of Technology in Industrial Engineering in1988 and Ph.D. (Industrial Management) in 1997 from Indian Institute Technology, Kharagpur. He was employed in ICIM as marketing executive for over a year and then has taught at Cochin University for the last 23 years. He has also served as the Managing Director of CONSUMERFED, Kerala for nearly three years. He is currently the Director of the School of Management Studies of Cochin University. He is a member of IEEE and a life member of the IIIE and Kerala Management Association. He has 14 international and 47 National Journal publications. His research interests are in the areas of Operations Management, Information systems and Project Management. He has guided 10 Ph.D. Thesis.

Third Author – PhD., Pramod K.V., Department of Computer Applications, Cochin University of Science & Technology, Kochi, India, pramodkv4@gmail.com

Pramod is presently working as Associate Professor and Head of the Department of Computer Applications, Cochin University of Science and Technology. He received Master of Science in Mathematics from University of Kerala, in 1981 and Ph.D. in Simulation & Modeling from Cochin University of Science and Technology in 1989. He has more than 20 years of teaching and research experience. He is a member of IEEE. He has published about 25 research papers in National and International Journals. Two of his research students have been awarded PhD degree, one in Cryptography and other in Mathematical Morphology. His areas of research interests are Simulations & Modeling, Cryptography, Coding theory, Mathematical Morphology and Data Mining.

Correspondence Author – Srikrishnan Sundararajan, Master of Technology (Computer Science & data Processing), Department of Computer Applications, Cochin University of Science & Technology, Kochi, India, sskrishnan02@yahoo.com.