

Student Work

---

12-1-2005

# A case study of effective practices for the management of global software development projects

Mudita Agarwal  
*University of Nebraska at Omaha*

Follow this and additional works at: <https://digitalcommons.unomaha.edu/studentwork>

---

## Recommended Citation

Agarwal, Mudita, "A case study of effective practices for the management of global software development projects" (2005). *Student Work*. 1361.

<https://digitalcommons.unomaha.edu/studentwork/1361>

This Thesis is brought to you for free and open access by DigitalCommons@UNO. It has been accepted for inclusion in Student Work by an authorized administrator of DigitalCommons@UNO. For more information, please contact [unodigitalcommons@unomaha.edu](mailto:unodigitalcommons@unomaha.edu).



**A CASE STUDY OF EFFECTIVE PRACTICES  
FOR THE MANAGEMENT OF GLOBAL SOFTWARE  
DEVELOPMENT PROJECTS**

**A Thesis**

Presented to the  
Department of Information Systems and Quantitative Analysis  
and the  
Faculty of the Graduate College  
University of Nebraska

In Partial Fulfillment of the Requirements for the Degree  
Master of Science in Management Information Systems  
University of Nebraska at Omaha

**By**

Mudita Agarwal  
December, 2005

UMI Number: EP73401

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI EP73401

Published by ProQuest LLC (2015). Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against unauthorized copying under Title 17, United States Code



ProQuest LLC.  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 - 1346

**University of Nebraska, Omaha**  
**Graduate College**


A Case Study of Effective Practices for the Management of  
Global Software Development Projects

Mudita Agarwal, ISQA Department

**THESIS ACCEPTANCE**

Acceptance for the faculty of the Graduate College,  
University of Nebraska, Omaha, in partial fulfillment of the requirements  
for Master of Science in Management Information Systems

**Committee**



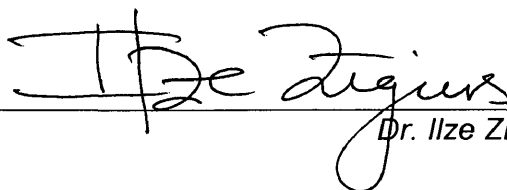
---

Dr. Gert-Jan de Vreede



---

Dr. Parvathi Chundi



---

Dr. Ilze Zigurs, Chairperson

11/28/2005  
Date

**A CASE STUDY OF EFFECTIVE PRACTICES FOR THE MANAGEMENT  
OF  
GLOBAL SOFTWARE DEVELOPMENT PROJECTS**

MUDITA AGARWAL, MS IN MIS

University of Nebraska, 2005

Advisor: DR. ILZE ZIGURS

Global software development has proliferated in recent years because of rapid globalization, development of telecommunication and information technologies, and maturing of the software development processes.

This thesis synthesizes available research on the global software development paradigm into an integrated model. The theoretical study analyzes different aspects of dispersion, their effect on traditional group processes of communication, coordination and control, and the recommendations in the literature for addressing some of these issues. The model developed in the theoretical study was then used to perform a detailed case study of a CMM Level 5 software company that specializes in global software development.

A comparison of findings from the literature survey with these insights from a practitioner organization was used to draw inferences about how closely the theoretical model follows the real issues faced by industry, the practices and methodologies actually being used, and some areas of concern that available research does not address adequately.

This case study revealed overlaps as well as differences between academic research and practice. Recommendations are made to managers of global software projects and areas of future research are identified.

# CONTENTS

<b>1. INTRODUCTION</b> .....	<b>1</b>
1.1. BACKGROUND .....	1
1.2. MECHANICS OF GLOBAL SOFTWARE DEVELOPMENT .....	4
<b>2. PURPOSE OF THE RESEARCH</b> .....	<b>5</b>
2.1. SCOPE .....	5
<b>3. THEORETICAL FRAMEWORK</b> .....	<b>7</b>
3.1. DISPERSION .....	8
3.2. GROUP PROCESSES .....	9
3.2.1. <i>Communication</i> .....	9
3.2.2. <i>Coordination</i> .....	10
3.2.3. <i>Control</i> .....	11
3.3. PRACTICES THAT SUPPORT GROUP PROCESSES .....	11
3.3.1. <i>Methodological Practices</i> .....	11
3.3.1.1 Management Practices.....	11
3.3.1.2 Object-Oriented Project Management Model .....	12
3.3.1.3 The Capability Maturity Model .....	13
3.3.1.4 Software Development Processes .....	15
3.3.1.5 Team Organization .....	16
3.3.2. <i>Team Practices</i> .....	17
3.3.2.1 Dynamics of Virtual Teams.....	17
3.3.2.2 Leadership .....	20
3.3.3. <i>Technological Environment</i> .....	22
3.3.3.1 Group Communication Support Systems (GCSS) .....	24
3.3.3.2 Group Information Support Systems (GISS).....	24
3.3.3.3 Group's External Support System (GXSS).....	25
3.3.3.4 Group's Performance Support Systems (GPSS).....	25
3.3.3.5 Review of Tools .....	26
3.4. SUMMARY OF LITERATURE SURVEY.....	28
3.5. EXPECTED FINDINGS FROM THE CASE STUDY .....	29
<b>4. RESEARCH METHOD</b> .....	<b>31</b>
4.1. METHOD OF STUDY .....	31
4.2. PROFILE OF THE ORGANIZATION .....	31
4.3. DATA COLLECTION METHODOLOGY .....	32
4.4. STUDY PARTICIPANTS .....	33
4.5. DATA ANALYSIS METHODOLOGY .....	34
<b>5. FINDINGS AND DISCUSSION</b> .....	<b>38</b>
5.1. FINDINGS FROM THE QUESTIONNAIRE .....	38
5.2. FINDINGS FROM INTERVIEWS.....	42
5.2.1. <i>Issues Faced</i> .....	42
5.2.1.1. <i>Communication</i> .....	44
5.2.1.2. <i>Coordination</i> .....	47
5.2.1.3. <i>Control</i> .....	48
5.2.2. <i>Strategies Used</i> .....	50
5.2.2.1. <i>Methodological Practices</i> .....	51
5.2.2.2. <i>Team Practices</i> .....	54
5.2.2.3. <i>Use of Technology</i> .....	54
5.2.2.4. <i>Practices not found useful</i> .....	55
<b>6. CONCLUSION AND IMPLICATIONS</b> .....	<b>56</b>
6.1. CONCLUSIONS .....	56

6.2.	RECOMMENDATIONS TO MANAGERS.....	57
6.3.	LIMITATIONS OF THIS STUDY.....	58
6.4.	FUTURE AREA OF RESEARCH .....	59
	<b>REFERENCES .....</b>	<b>61</b>
	<b>APPENDIX A: COLLABORATION SOFTWARE PRODUCTS.....</b>	<b>65</b>
	<b>APPENDIX B: COVER LETTER FOR IRB .....</b>	<b>68</b>
	<b>APPENDIX C: QUESTIONNAIRE .....</b>	<b>69</b>
	<b>APPENDIX D: INTERVIEW QUESTIONS.....</b>	<b>76</b>
	<b>APPENDIX E: RESPONSES OF THE INTERVIEWEES.....</b>	<b>77</b>
	<b>APPENDIX F: TABULATED QUESTIONNAIRE RESULTS.....</b>	<b>84</b>
	<b>APPENDIX G: CALCULATION OF PROJECT CHARACTERISTICS .....</b>	<b>86</b>
	PROJECT SCOPE .....	86
	PROJECT COMPLEXITY.....	86
	PROJECT RISK .....	87
	TECHNOLOGY USE.....	88
	DEGREE OF SUCCESS .....	88

## Figures

Figure 1: Theoretical Framework of the Study .....	7
Figure 2: Use of Collaboration Tools and Technology .....	41
Figure 3: Communication issues found through the Interviews .....	45
Figure 4: Coordination issues found through Interviews .....	47
Figure 5: Control issues found through Interviews .....	49

## Tables

Table 1: Summary of CMM .....	14
Table 2: Summary of literature survey .....	29
Table 3: Types of projects worked on by the participants .....	34
Table 4: Items used to measure complexity, scope, risk, technology use, and success .....	36
Table 5: Actual scores of complexity, scope, risk and overall project type .....	38
Table 6: Overall success ratings of each project .....	39
Table 7: Project type and use of technology .....	39
Table 8: Project type, use of technology and project success .....	40
Table 9: Technology use by role in project .....	41
Table 10: Focus of interview questions .....	42
Table 11: Problems discussed in literature and their occurrence in the sample projects .....	43
Table 12: Additional problems reported in the sample projects .....	44
Table 13: Strategies recommended in literature that are actually in use in the company .....	51
Table 14: Strategies that had not been found in the literature survey .....	51



# 1. Introduction

## 1.1. *Background*

The last decade has been characterized by rapid acceleration in the pace of globalization of industry and commerce. The software engineering industry took a lead in this globalization phenomenon by rapidly embracing the collaborative work paradigm where distributed teams or individuals were working on the same projects either simultaneously or sequentially.

The primary driver for this trend is the cost of software development which is comparatively high in the U.S. and in Europe. According to McKinsey Global Institute, labor can account for three quarters of the cost of developing software in the United States.

Another significant development over the last five years has been the fall of information technology (IT) from its erstwhile position as a core driver of competitive advantage for the knowledge-age business. The rapid advancement and embrace of IT by all major sectors of industry and commerce has led to a situation where most of the major competitors in any industry are nearly at par with each other in terms of their use of IT. Organizations that were not able to keep pace with competitors in using IT for enhancing their product, operations, market intelligence, customer insight, product delivery, and customer service, lost their competitive edge a long time ago and fell by the wayside. In the 21st century, the emphasis is on streamlining and focusing operations to meet the challenges of globalization of economies. Information technology has been relegated to a non-core activity that many companies no longer consider a primary source of competitive advantage and are therefore more and more willing to outsource. End-user IT organizations today want to focus only on those software products that their companies expect to help them directly in gaining a competitive edge over competitors. An example of this phenomenon is the rise of commoditized "on-demand" IT services from all major vendors in the last five years. This scenario encourages companies to outsource not just IT operations, but also increasingly - software design, development

and support. In many cases, the result is multiple vendors and dispersed teams collaborating on projects.

These and other factors that are contributing to the proliferation of global software development are listed below (Carmel and Agarwal, 2001; Herbsleb et al., 2000; Gopal et al., 2000):

### Drivers

- Competitive pressures on companies to reduce costs
- Competitive pressures on companies to be swift and flexible in responding to market demands and conditions
- Increasing use of outsourcing by companies to allow them to focus on their core competencies and on the main drivers of their value chain
- Rapid globalization of all sectors of industry and commerce like manufacturing banking, trading, and investments
- Desire of companies to utilize the best labor pools available in the world
- Companies' desire to increase their global presence
- Productivity advantages from being able to work across time zones
- Tight labor markets in the technology sector in the western economies

### Enablers

- Development and diffusion of computer technology
- Maturity of the software development process
- Advances in telecommunication technology
- Declining costs of computing and communication infrastructure
- Commoditization of information technology and IT services

This model of collaborative software development presents a variety of opportunities to companies to reduce their costs of software development and to increase their speed to market. However, implementation and management of a global software development model is a difficult task that presents a variety of procedural, logistic, technological, legal and management challenges. As more organizations adopt a global software

development model, there is an expectation that the global software development process will become mature and help produce quality software while surmounting all the challenges of this approach.

This model not only exacerbates and amplifies several challenges of traditional software development projects but the added dimension of virtuality also introduces a new set of issues. Some of the challenges arise from differences in infrastructure in different development locations, including network connectivity, development environment, and change and version management systems. Other major problems include issues arising due to interdependencies among work items and difficulties of coordination. Communication related problems are cited as the major challenge in GSD. Communication related problems have been attributed primarily to cultural disparity issues - different training, different work processes, different native languages and difference in work experiences, absence of informal communication and team-building due to distance and technological barriers to smooth communication. There are also problems related to lack of trust and lack of willingness to communicate openly across sites. In today's scenario, fear of loss of job by sharing their expertise creates new challenges to this phenomenon (Mockus and Herbsleb, 2001). Some common problems addressed in the literature relate to communication, coordination, geographical dispersion, team management, teamwork, and cultural differences (Battin et al., 2001; Carmel and Agarwal, 2001; Herbsleb et al., 2002).

The focus of this study will be to understand the different factors that play a role in global software development by doing a thorough literature survey and simultaneously doing a detailed study of a CMM Level 5 software company to get practical insights. It will help in understanding how the challenges of global software development are being addressed by the sample company and corroborating/denying the findings from the literature survey. A company that has been assessed at CMM level 5 (the highest level of the Capability Maturity Model of the Software Engineering Institute at the Carnegie Mellon University) was specifically chosen for this study. A CMM level 5 assessment indicates that the company follows highly evolved and mature software development processes and is able to quantitatively measure, analyze, predict, and improve the performance of its processes. A brief synopsis of CMM is included in section 3.3.1.3.

## **1.2. *Mechanics of Global Software Development***

Software companies have developed a wide variety of approaches for deploying a global development model. The global development model was extensively used by software consulting and services companies operating worldwide centers developing products and delivering outsourced IT services. In recent years this paradigm has been increasingly adopted by end-user organizations like large financial institutions and manufacturing corporations to operate IT delivery centers in various parts of the world. In a global software development environment, work is typically divided between on-site and off-site teams (Gopal et al., 2000). A small team is stationed at the client site to interface with the customer and to handle certain tasks like installation, system integration, user acceptance testing, and so on. Typically, initial requirements analysis is conducted at the client site by a small team and detailed requirement specifications are developed off-site. Project leaders and senior designers assemble the team and execute the development off-site. For larger projects, a small team may be stationed on-site during the detailed design and development phase, to coordinate with the customer and to give the customer a “local” single point of contact. Once the software is ready, it is sent to the on-site team for acceptance testing, installation, integration and rollout. Maintenance and ongoing support is provided by the off-site team, with a small on-site presence maintained, if required, to provide a local first point of contact to the users. This is a typical global software development project scenario.

## **2. Purpose of the Research**

The primary objective of this research is to study how collaborative practices, processes and technology impact the effective management of global software development teams through a literature survey and a case study of a software company specializing in global software development. The research question is: How do global software development teams manage team processes in virtual projects? Specifically, what methodological and team practices are effective for managing process? What technologies are used and how do they affect processes? Can we develop an integrated framework of these concepts that helps to explain existing practice and provide guidance for future research?

The research was performed by studying available literature on this topic and synthesizing it into an integrated model to provide a framework of key factors. A detailed case study of a CMM Level 5 organization that has significant expertise and experience in global software development was then performed. Findings of the survey were also organized in the framework provided by the integrated model. This case study was used as an initial test of the integrated model built from the literature survey.

A comparison of the findings from the literature survey with these insights from a practitioner was used to draw inferences about how closely the theoretical model follows the real issues faced by the industry, the practices and methodologies being used and some areas of concern that available research does not address adequately. This comparison was used to make recommendations to managers of global software projects and to identify certain areas of future research.

### **2.1. Scope**

The literature on issues related to global software development is voluminous and fragmented and shows rapid evolution that has kept pace with the evolution of technology, methodologies and best practices in this area. The key areas of study have been management methods and practices used in global software development projects

with respect to virtual teams, project management, and collaborative tools and technologies. The study primarily focuses on methodological and management practices and how these management practices are supported by current technology in a global development scenario. This research does not discuss details related to the financial aspects of global software development. It also does not address the social and economic fallouts of global software development resulting from relocation of work or outsourcing offshore, “near-shore” or locally.

The case study was exploratory in nature and was not used to perform formal statistical hypothesis testing or validation of the findings from literature survey. Nonetheless, the study makes a contribution by examining concepts in a real setting. Most existing empirical research is based on experiments conducted in a laboratory setting or of a conceptual nature. This study in contrast was conducted in a real software company using real-life projects as the source for data.

This thesis is structured as follows:

- Section 3 provides a theoretical framework based on the literature on global software development projects
- Section 4 describes the research method, including data gathering and data analysis methodology
- Section 5 provides findings and a discussion of their implications
- Section 6 contains the conclusions

### 3. Theoretical Framework

A conceptual framework was developed based on the relevant constructs and variables discovered from the literature survey. To develop the framework presented in Figure 1, an extensive literature review was conducted in the fields of global software development, virtual teams, collaborative tools, and distributed project management.

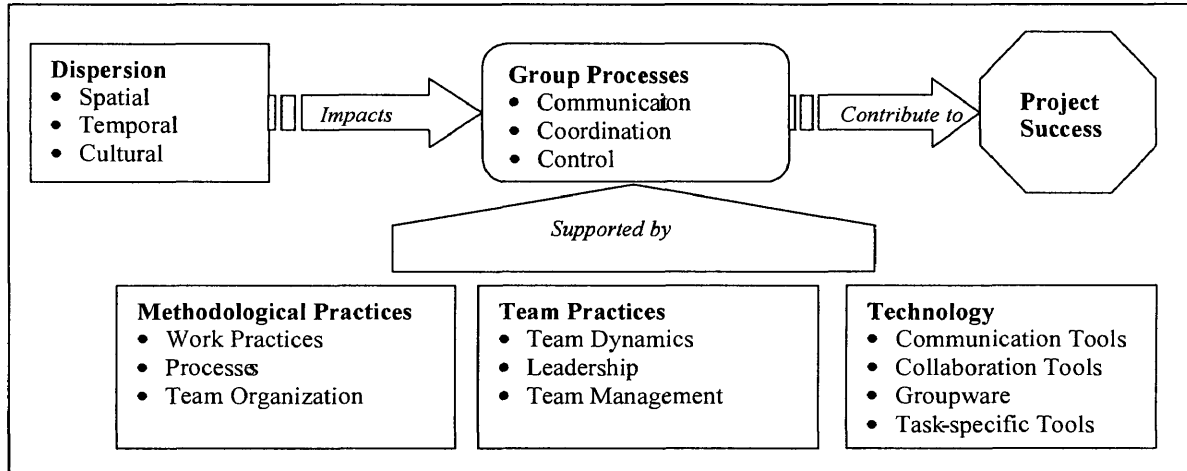


Figure 1: Theoretical Framework of the Study

This framework is modeled on the finding that the spatial, temporal, and cultural dispersion built-in in the global software development paradigm affects and challenges the traditional group processes of the project and ultimately impacts project success. To mitigate these effects, research suggests several actions that organizations can take. These actions can be broadly categorized into Methodological Practices, Team Practices, and use of Technology.

The literature review focused on those aspects of the constructs that were relevant to global software development. We developed the framework based on what appear to be the most important features and that could be confirmed. For example, the framework does not include individual characteristics of the team members as our focus is on team-level issues. Any attempt to model such a complex process requires zeroing in on certain aspects to the exclusion of others. No single model can capture all the potential factors involved. However, even recognizing this limitation, it is still useful to focus on

factors that are likely to make a difference and that can help to fill in some of the gaps. That is the intent of the current study.

Each component of the research framework is discussed below, with an elaboration of what we know and don't know about that component from the current literature.

### **3.1. Dispersion**

The basic premise of global software development is dispersion. The literature uses the words "dispersion" and "distance" interchangeably in the context of global software development. "Dispersion", however, is favored because it carries the connotation of more than a spatial separation, and it describes more accurately the idea of separation on multiple dimensions.

Dispersion has been used primarily in terms of distance and time. Some other dimensions of dispersion include organizational affiliation, culture, continuity of team membership, experience, availability, and technology (Khazanachi and Zigurs, 2005). Distance affects the process of communication, coordination, and control that is required in software development, with a direct consequence on how software is defined, constructed, tested, and delivered as well as how development is managed (Damian et al., 2003). Spatial and temporal dispersion can make team interactions disjointed by disrupting the traditional coordination of workflow by implicit reference to time, place, and sequence. According to Massey et al. (2003), temporal patterning defines the rhythms by which teams synchronize or coordinate their activities. The temporal aspects of the flow of work influence communicative, decisional, and interpersonal behaviors.

Carmel has referred to geographical dispersion as one of the centrifugal factors that pull global projects apart (Carmel and Agarwal, 2002). Most of the literature suggests that distance primarily affects communication, and that in turn creates problem in coordination and control.

Herbsleb and Moitra (2001) found that geographical dispersion also affects strategic issues like division of work across sites (on-site and off-site). Division of work is constrained by the availability of resources at the sites, and level of expertise in various technologies and infrastructure. Some other effects of geographical dispersion found in the literature include misalignment between senior and middle management,



implications related to loss of job, loss of control, and fear of the possibility of relocation and the need for extensive travel (Herbsleb and Moitra, 2001; Carmel and Agarwal, 2002).

The physically distributed environment also poses new problems for management of the software development lifecycle. Traditional problems related to the software lifecycle and process management, especially in the requirements and analysis phase, become more critical due to physical distance (Zanoni and Audy, 2004). Geographical dispersion also creates cultural distance. Issues related to culture mentioned in the literature include the need for structure, attitudes towards hierarchy, sense of time, and communication styles (Herbsleb and Moitra, 2001; Lanubile et al., 2003).

## **3.2. Group Processes**

### **3.2.1. Communication**

Communication is a major concern in global software development. There are many studies available on communication issues in the context of dispersed teams. Khazanchi and Zigurs, (2005) have defined communication as the process through which people convey meaning to one another via the exchange of messages and information in order to carry out project activities. There are two kinds of communication involved in project management - formal and informal. Formal communication is required for activities like updating project status, escalating project issues, and making groups responsible for particular work products (Herbsleb and Moitra, 2001). Formal communication has not been highlighted as the primary area of concern in the literature. Informal communication however, creates several challenges for the dispersed project team.

The absence of informal communication due to distance is highlighted as a major issue throughout the literature on global software development and virtual teams. Research suggests that informal communication plays an important role in the software development process. A study done by Herbsleb et al, 2003 on distance, dependencies and delay in global collaboration found that "*diminished communication across distance, and the loss of the subtle modes of face-to-face communication and coordination that co-located work affords, appear to have rather dramatic and unfortunate consequences*"

(pp 8, Conclusions). Some consequences mentioned were the need for more people for the same amount of work, difficulties in change management due to lack of informal consultation among team members, and difficulties in involving the right people in problem solving because the expertise of people available is not known. There is also fear of loss of proprietary information about products or schedules, leading to restricted information sharing and impaired informal communication (Herbsleb and Moitra, 2001). Another issue discussed in literature relates to communication problems resulting from inconsistencies in notations and terminologies (Battin et al., 2001). Handel and Herbsleb (2002) also found that a substantial reduction in “corridor talk” type of informal, ad-hoc communication tends to sometimes create coordination-related issues.

Damian and Zowghi (2003) specifically studied the effect of communication on the requirement definition phase of globally-dispersed projects. They found that respondents in both the companies that they surveyed displayed near unanimity that the most urgent areas of concern were ineffective requirements meetings and inability to resolve conflicts about requirements because of deficiencies in formal and informal communication between sites. They also found that these unresolved early-stage conflicts tend to affect the quality and consistency of work, mutual trust and interpersonal relationships throughout the life of the project.

### **3.2.2. Coordination**

Global software development is based on the premise that development activities can be continued seamlessly around the clock and around the globe. Software development takes place in phases and modules involving a great deal of coordination among teams across the globe. Carmel and Agarwal defined Coordination as the act of integrating each task with each organizational unit, so the unit contributes to the overall objective. Khazanchi and Zigurs, (2005) have defined coordination as the mechanism through which people and technological resources are combined to carry out specific activities in order to accomplish goals.

Most available literature finds that global teams are usually characterized by poor coordination. Some problems mentioned in the literature related to coordination include unrecognized conflicts among the assumptions made at different sites, difficulty

coordinating the pacing of work between sites, incorrect interpretation of communication, and difficulty in finding and establishing contact with the appropriate person (Herbsleb et al., 2000).

### **3.2.3. Control**

Control is the process of ensuring adherence to goals, policies, standards, or quality levels. Khazanchi and Zigurs (2005) have defined control as the process of monitoring and measuring project activities so as to anticipate and manage variances from project plans and organizational goals. Controls can be formal (such as budget and explicit guidelines) or informal (such as peer pressure) (Carmel and Agarwal, 2002). Pare and Dubé (1999) found evidence that control, which is a key activity in software development, gets more challenging in a distributed development environment. Both aspects of control – monitoring/measurement as well as corrective/directive action become complex and difficult in a distributed environment.

## **3.3. Practices that Support Group Processes**

Dispersion affects three key aspects of software development, i.e., communication, coordination and control, and that in turn creates the range of challenges discussed above. In the literature, a number of practices have been documented that organizations have adopted to overcome these challenges. These practices may be grouped into methodological practices, team practices, and use of technology. The current literature on each of these practices is discussed in the following sections.

### **3.3.1. Methodological Practices**

#### **3.3.1.1 Management Practices**

Diverse, global and geographically dispersed teams present new challenges to traditional project management approaches. There has not been much research that has tested the effectiveness of traditional practices in this new environment (Beise, 2004). Individual aspects of the Project Management Body of Knowledge (PMBOK) like time, risk and communication have been linked to project outcomes (Beise, 2004). Dubé and

Paré (1999) have proposed that members of dispersed teams should be brought together in the beginning to have startup sessions to address the challenges of communication. Battin et al. (2001) have described the strategy adopted in a real case study at Motorola. That project used “liaison engineers” who learned the details of the system from existing developers, completed system-level requirements and specifications, and communicated the information back to their development staff at home.

Beise (2004) found that communication and information flow must be more frequent and continuous for project effectiveness. Ebert and De Neve (2001) have suggested centralization of the project management function and decision-making for better coordination. They also found that it is important that the work is clearly demarcated up front among team members and teams across the globe. This added clarity of roles and division of work and responsibilities leads to better coordination and control.

Mechanisms like developing practical performance metrics, increased visibility via frequent deliverables, prototyping and early integration, and tighter definition of project reporting mechanisms have been proposed as ways of monitoring (Dubé and Paré, 1999; Beise, 2004). Risk management is also considered a critical success factor in global project management (Beise, 2004).

### **3.3.1.2 Object-Oriented Project Management Model**

Zanoni and Audy (2004) have proposed the adoption of specific languages and more formal and determined development processes to help software project managers address the issues of communication during key phases of the software development cycle. They have proposed the Object-Oriented Project Management model that is more suited to development in a distributed environment than the traditional “Procedural Project Management model”. Their model uses object-orientation as the basic developmental methodology. Object-oriented project management focuses on communication. They recommend the UML modeling language to describe and communicate system requirements of software, project, and code. UML focuses primarily on specifying and documenting system requirements. According to them, it is also important to standardize on UML during the developmental process and to use and link UML artifacts all through the various phases of the software development.

Their proposed project management model is divided into six phases, i.e., requirement definition, project exploration and definition, production processes, evaluation, transition, and integration. The proposed model differs from the present ones in seeking this conversation between the development process in a distributed environment and the management process, incorporating a spiral life cycle, the orientation to objects, and the UML language. The model calls for special emphasis on the definition of requirements and integration phases, which are outstanding as the most affected in the development environment of distributed software. UML artifacts can be linked and used during the various phases of the development process. In this model, the focus in all the phases of the project is on communication between the development teams and documentation of the activities, with the help of UML. The project management model based on UML focuses on the need for communication and coordination in global software projects due to dispersion.

### **3.3.1.3 The Capability Maturity Model**

A major factor in global software development is adherence to quality, and the Capability Maturity Model (CMM<sup>1</sup>) plays a significant role in this issue. In their article “Lessons from India Inc.”, Anthes and Vijayan (2001) quote a chief technology officer from an Indian company: “...to make this global concept work, there is no other option but to make it process-driven rather than people-driven. We had to follow good practices for documentation, communications, signoffs, revisions – all these needed to happen if this model was going to work.”.

Literature suggests that one of the potential benefits of the CMM is availability of thorough documentation of the entire software development process and its emphasis on repeatable results. The Capability Maturity Model helps organizations to decrease their dependence on specific individuals. Implementing quality measures such as CMM can help companies move from being people-dependent to becoming more process-reliant (Anthes and Vijayan, 2001). This reduction in people-dependence improves the

---

<sup>1</sup> The CMM model was upgraded by SEI in 2001 to a more comprehensive assessment model called the CMM-Integration (or CMMI). Software companies have to switch to CMMI at the time of their next revalidation or assessment.

performance of geographically-dispersed teams. Thorough documentation, standardization, measurement, process-orientation, training, and quality management, companies can remove many of the largest sources of coordination issues in global projects. For example, issues like operational ambiguities and the ad hoc nature of coordination can be addressed.

CMM-assessed organizations are required to have strong knowledge management capabilities and they have to deploy rigorous mechanisms and processes for building, refining and sharing repositories of knowledge, experiences, tools and techniques across projects. This institutionalized model of knowledge-sharing and continuous learning enables CMM-assessed companies to adapt to the challenges of global software development better and faster.

### ***Capability Maturity Model***

*The Capability Maturity Model for software is a reference model for appraising software process maturity and normative model for helping software organizations progress along an evolutionary path from ad-hoc chaotic processes to mature disciplined processes (Herbsleb et al., 1997).*

*The CMM is organized into five maturity levels and each maturity level is decomposed into several key process areas that indicate areas that organizations should focus on to improve their software processes.*

<b>CMM level</b>	<b>Characteristics</b>	<b>Key Process Areas</b>
<i>1. Initial</i>	<i>Undefined processes Success is dependent on competent people</i>	
<i>2. Repeatable</i>	<i>Basic Project Management Processes are established to track cost, schedule and functionality.</i>	<i>Requirement management Software project planning Software project tracking and oversight Software subcontract management Software Quality Assurance Software configuration management</i>
<i>3. Defined</i>	<i>Software process for both management and engineering activities is documented, standardized and integrated into a standard software process for the organization.</i>	<i>Organization Process Focus Organization process definition Training program Integrated software management Software Product Engineering Inter-group Coordination Peer Reviews</i>
<i>4. Managed</i>	<i>Product and Process Quality are quantitatively measured, understood and controlled.</i>	<i>Quantitative process management Software Quality management</i>
<i>5. Optimizing</i>	<i>Continuous process improvement is facilitated by quantitative feedback from the process and from piloting innovative ideas and technologies.</i>	<i>Defect prevention Technology change management Process change management</i>

**Table 1: Summary of CMM.**

**Source: Herbsleb, Zubrow, Goldenson, Hayes, and Paulk (1997)**

#### 3.3.1.4 Software Development Processes

Software development typically goes through well-defined requirements, analysis, design, implementation, and maintenance phases. As mentioned above, geographical dispersion creates difficulties in communication, coordination, and control that in turn pose problems in some of these stages. Requirements analysis and change management get more complicated in a dispersed environment. Several approaches have been mentioned in the literature to address issues faced in the development phases.

Ebert and De Neve (2001) have focused their approach on the functionality of the system under development. They recommend devising a project plan based on "requirement clusters" and making one team responsible for each cluster. In their study for achieving real incremental development, they have proposed the following process: analyze requirements and cluster them according to functionality; create a project plan on the basis of clustered requirements; and assign each set of requirements to a development team, making one team responsible for each increment. The authors found that the most important thing for global development was to combine concurrent engineering with continuous build and teamwork and to create responsibility for results. Carmel and Agarwal (2001) have talked about giving the ownership of individual software components, individual modules, releases or entire products to one team to smooth the software development process and to reduce ongoing collaboration within the team. Herbsleb et al. (2000), in their study of six software development organizations, found that organizations reduce requirements for cross-site communication by structuring teams in alignment with the product architecture as a mechanism for better coordination. Another approach adopted by the organizations was to reduce the coupling of cross-site work through upfront planning for the geographic split during the design phase. Herbsleb and Perry (1999) have suggested the following four ways to improve coordination:

1. Create organization units based on functional areas of expertise like system engineering, human interface, design, testing, and development (the Functional Area Model),

2. Put all work necessary to produce a product release in one organizational unit (the Project Model),
3. Co-locate development of a particular product component, and
4. Co-locate the activities from a given process stage (for example, design).

Teams working in different parts of the world often have different notations and terminologies for the same things. Battin et al. (2001) found that global organizations develop a set of common “work products” and vocabulary to address this issue. These are some examples of the strategies found in the literature to address communication, coordination and control in the software development phases.

### **3.3.1.5 Team Organization**

Ramesh and Dennis (2002) have proposed the concept of object-oriented teams for better management. Object-oriented teams strive to decouple team members through the use of semantically-rich media. Examples of semantically-rich media include document repositories, code repositories, and bug repositories. Object-oriented teams have well-defined processes, exchange information (inputs and outputs) with other objects through well defined semantically-rich interfaces, and produce a decreased flow of information. Object-oriented teams use routine, mature work processes with well-defined task deliverables and processes.

Ebert and De Neve (2001) have proposed the concept of coherent and collocated teams of fully allocated engineers. According to them, coherence in the work-breakdown can be achieved by splitting work according to feature content, which allows assembling a team that can implement a set of related functionality as opposed to artificial architectural split. They also recommend collocation of engineers working on such a set of coherent functionality in the same building, or even in the same room. They also suggest that engineers working in a same project should be responsible solely for the project and not be distracted by different tasks for other projects.

These two examples of different types of team organization suggest that teams can be organized along different dimensions in a dispersed work environment. These team organizations are used as a means to support group processes in a virtual environment to facilitate team effectiveness.



### **3.3.2. Team Practices**

#### **3.3.2.1 Dynamics of Virtual Teams**

Global software development is performed by globally-dispersed “virtual” teams. The literature on virtual teams has many definitions with common threads. These common threads include distance, organization, and separation of the teams on the basis of time zone differences. Some authors define global virtual teams as teams whose members share a common purpose and are located in at least two different countries (Ramesh and Dennis, 2002). Powell et al. (2004) found many definitions of virtual teams in their review of this literature. They defined virtual teams as groups of geographically, organizationally, and/or temporally-dispersed workers brought together by information and communication technologies to accomplish one or more organizational tasks (Powell et al., 2004). Virtual teams are often assembled in response to specific needs and are often short lived (Powell et al., 2004). Dubé and Paré (1999) have mentioned geographic dispersion, task or project duration, prior shared work experience, members’ assignment, membership stability, task interdependence, and cultural diversity as distinguishing characteristics of a virtual team. They identified three characteristics that specifically differentiate virtual teams from conventional ones. These include degree of reliance on information communication technologies (ICT), ICT availability and team members’ ICT proficiency. Massey et al. (2001) have emphasized the cultural separation of members of virtual teams. These cultural differences may be rooted in the country of origin, or in organizational or functional differences.

Virtual teams face a number of problems due to their geographical, temporal, and cultural distance. Some of the problems commonly mentioned in the literature are absence of face-to-face interaction and informal communication, cultural difference between virtual sites, difficulty of building trust, group cohesiveness, and group affiliation. Cohesiveness measures the extent to which members are attracted to the group and to each other. Group cohesiveness has been linked to a number of positive outcomes such as enhanced motivation, better decision making, and open communication. There has also been a link between team effectiveness and team member relationships. Stronger relational links have been associated with higher task performance and the effectiveness of information exchange. These relational links have

also been associated with enhanced creativity and motivation, increased morale, better decisions and fewer process losses (Pauleen and Yoong, 2001).

The literature on virtual teams emphasizes the importance of face-to-face communication over any technology-mediated communication. Some disadvantages mentioned in the literature due to lack of face-to-face interaction include lengthy and confusing team interaction and discussion leading to poorer comprehension and understanding, inability of individuals to observe others, respond to situations and to read others' facial expressions, gestures, tones and voice intonations (Brodia, 1997; Burtha, 2002).

Media richness and media synchronicity are two important characteristics of communication found in the literature. There are a number of theories associated with these two concepts. Synchronicity describes the ability of a medium to create a sense that all the participants are currently engaged in the communication event. Face-to-face meetings provide high degrees of synchronicity as participants have the opportunity to participate in real time. Some media attributes related to media synchronicity identified in the research include speed of interaction (also called speed of feedback), rehearsability and reprocessability (Carlson and George, 2004).

Rehearsability refers to a participant's ability to spend time in planning, editing and even rehearsing the actual content and manner of delivery of the message due to media time delays. Research has indicated that the speed of interaction afforded by the medium is related to rehearsability but this relationship is proved to be both negative (i.e., increasing interaction speed leads to decreasing opportunities for rehearsal) and not 100% correlated (Carlson and George, 2004).

Reprocessability refers to the media's capability to store the information allowing participants to review and analyze the material more than once and at subsequent points in time. However in the presence of today's surveillance technologies, any communication interaction can be electronically observed and recorded, facilitating subsequent reprocessing (Carlson and George, 2004).

Media Richness theory and Social Presence theory suggest that computer-mediated communication systems may eliminate the type of communication cues that individuals use to convey trust, warmth and attentiveness (Beranek, 2000). According to Beranek

(2000), the virtual context can constrain or even impede the development of trust in virtual teams. In the literature on virtual teams, trust has been found to play a role in problem-solving, organizational performance, organizational communication, and acceptance of feedback.

When feasible, using face-to-face meetings during project planning and limiting the use of asynchronous electronic communication to coordinate tasks such as scheduling, sharing results and sharing documentation, appears critical to the development of the team and its successful interaction (DeMeyer, 1991). Early face-to-face meetings during the team's launch phase have been found to improve the team's project definition, to foster socialization, trust and respect among team members, and to enhance the effectiveness of subsequent electronic communication (Ramesh and Dennis, 2002). Dubé and Paré (1999) found that starting a new project with face-to-face communication is a highly useful investment and total reliance on ICT may make coordinating and resolving conflicts more difficult. Burtha (2004) suggests incorporating frequent visits into a team leader's routine. Lanubile et al. (2003) have suggested solutions like kickoff meetings, periodically collocated meetings, initial cultural training, and group chats with photos to decrease social distance.

Keyzerman, 2003 suggests a "contracting discussions" approach for improving interpersonal trust, setting group expectations and decreasing social distance by providing an upfront opportunity to virtual teams and team members to negotiate expectations with one another and discussing how they will work together. Keyzerman also highlights the increased importance of maintaining personal credibility in a distributed scenario by delivering on commitments of quality and schedule.

Damian (2004) performed an empirical study of the effects of increasing levels of dispersion and use of multimedia groupware tools on group performance in a software requirements negotiation setting. She used four different configurations of co-located and distributed negotiators, facilitator and system analysts participating in a requirements engineering session. She had some interesting findings that are contrary to the above media richness theory and the established notion that increased opportunities for face-to-face communication in any phase of a software project can only improve overall team performance and vice versa. She found that the use of 'less rich'

communication technology (multimedia meeting via Microsoft Netmeeting), did not result in significant degradation of group performance as compared to face-to-face requirement sessions. Group performance was measured by the speed and quality of the collective decision. This study also indicated that the more impersonal communication medium in a distributed setting enhanced a team's ability to remain detached, unemotional, impersonal and objective in the interaction. This was positively related to better individual and group performance in terms of negotiating harder in favor of one's interest as well as group ability to achieve consensus on contentious issues. Participants in the study indicated that face-to-face interaction made them less willing to voice opinions and suggestions and made them less objective, and the face-to-face interaction either created feelings of sympathy and compassion for the co-located negotiator or distrust and personal conflict. This study indicates that certain stages of software development projects, especially the requirements phase, that involve a "negotiation" type of scenario, may actually benefit from a physical separation of the parties involved.

### **3.3.2.2 Leadership**

The literature on virtual team management has also focused on the importance of leadership and the issues and challenges faced by virtual leaders in virtual team management. The characteristics that distinguish virtual teams from conventional teams are the spatial distance between team members that restricts face-to-face communication and results in the use of technological communication to connect team members. This characteristic has an impact on two major leadership functions: performance management, and team development (Bell and Kozlowski, 2002).

The leadership factor has been studied extensively in multi-disciplinary literature both as an individual characteristic – that is, the specific abilities, soft skills and strategies for the leader of a virtual team, as well as a group characteristic – that is, the importance of leadership qualities like self-motivation, ownership and self-management in virtual team members.

Some challenges faced by virtual leaders include inability to meet one-to-one with members, inability of the project leader to reassure members of his/her own work ethic, leadership, support and control through continuous physical presence, and inability to communicate messages that the leader would like to keep confidential and unrecorded.

Kayworth and Leidner (2002) found that virtual team members look for qualities like concern for the members, empathy, understanding, strength and continuous feedback in their leader and it is a significant challenge for the leader of a dispersed team to meet these expectations. Steinfield et al. (1999) have described five types of operational needs arising from the "virtualness" of the team that the team leader needs to address during project planning. These include enhanced sharing of operational information like documents, designs, and other technical artifacts, a need for frequent real-time interaction, access to synchronous media and its effective use, information dissemination by communicating day-to-day project activities, and developments to group members and standardization and uniformity of group infrastructure. In a dispersed or virtual team scenario, leaders lose their informal networks that are a valuable tool for leaders to stay connected with their environment (Kayworth and Leidner, 2002).

Two broad approaches can be identified in the literature to address the challenges of leading dispersed teams. One solution is to develop specific skills and attitudes in team leaders that are unique to management of virtual teams. The other is to make changes in team functioning and work processes to alleviate some of these challenges.

Kayworth and Leidner (2001) suggest that a virtual leader should be skilled at playing the role of an "information mediator" with the kind of written communication skills that would enable the leader to clarify roles, maintain a structure and flow of messages, and exhibit an assertive yet caring persona. The same study found that the most effective leaders were those who were able to reduce the impact of team dispersion on regular communication, achieve quick resolution of team members' problems and provide continuous feedback and direction. Effective leaders were able to overcome the limitations posed by absence of face-to-face contact and could approach team members with a cordial yet assertive tone over electronic media. Virtual leaders also need to be aware of, and sensitive to, the cultural differences, communication and language barriers, rivalry and politics among teams, as well as disparities in technical proficiency and competence amongst dispersed teams (Dubé and Paré, 2001).

Another, somewhat radical approach proposed in the literature is to make the virtual teams "self-managed". This approach focuses on enabling team members of dispersed

teams to manage their own performance as far as possible and reducing the need for outside control and supervision. Some ways suggested for achieving this are by developing habitual routines (standard operating procedures) early in the project life cycle, setting explicit objectives and success criteria, creating a clear sense of mission, and developing an appropriate climate or tone for independent work (Bell and Kozlowski, 2002).

Massey et al. (2003) have explored ways in which leaders can improve temporal-coordination among temporally-dispersed teams. They found three generic temporal-patterning problems inherent in dispersed group activity and suggested generic managerial approaches to resolving each of them. These approaches include solving temporal ambiguity through task scheduling, addressing conflicting temporal interests and requirements through synchronization and pacing (i.e., aligning the pace of effort within the group and between dispersed groups), and managing the scarcity of temporal resources such as time through improved resource allocation management. Other approaches to address temporal coordination in virtual project teams include providing mechanisms for organizing synchronous group communication, for example frequent inter-team meetings, town-hall meetings, and providing a sequenced or structured process for work and problem-solving activities (Massey et al., 2003).

### ***3.3.3. Technological Environment***

Technology plays a key role in the whole process of global software development. It works as a connector that bridges spatial, temporal and cultural distances. Technology can be characterized on a variety of dimensions. Some of these dimensions include hardware or software infrastructure, level of support for information exchange, types of support provided, time-space configuration, or any number of characteristics of the underlying media (Khazanchi and Zigurs, 2005). Technology that enables work across the globe has been used with different notations including collaborative computing (Majchrzak et al., 2000; Attaran and Attaran, 2003), collaboration software (Kay, 2004), information and communication technologies (ICTs) (Pauleen and Yoong, 2001), groupware (DeFranco-Tommarello and Deek, 2002) and electronic systems for supporting collaborative work (McGrath and Hollingshead, 1994).

The literature in the following sections provides different perspectives of technology use in global software development. First, I review existing perspectives and attempt to bring together all those ideas into my unique perspective. An extensive review of collaboration technologies has been done to relate its use in the real world.

Zigurs and Buckland (1998) found three common themes among collaborative tools and technologies: support for communication, for process structuring, and for information processing. These authors defined group support systems technology as a set of communication, structuring, and information processing tools that are designed to work together to support the accomplishment of group tasks. Each aspect of the technology is defined as follows:

- Communication support can be defined as any aspect of technology that supports, enhances, or defines the capability of group members to communicate with each other. It includes elements such as simultaneous input, anonymous input, input feedback and a group display.
- Process structuring is any aspect of the technology that supports, enhances, or defines the process by which groups interact, including capabilities for agenda setting, agenda enforcement, facilitation, and creating a complete record of group interaction (via storing the agenda, all the input, the votes and so on).
- Information processing is the capability to gather share, aggregate, structure or evaluate information, including specialized templates such as stakeholder analysis or multiutility attribute analysis.

McGrath and Hollingshead (1994) had earlier classified electronic systems for supporting collaborative group work into four major categories. Over the last ten years, software tools have evolved and matured and have become progressively all-encompassing and these classification lines have been blurred. However, this classification still serves to clarify the primary issues that today's numerous collaborative software development tools address and the classification provides a clear framework for comparing the features and relative strengths and weaknesses of these tools in terms of their role in facilitating group processes. The four categories are:

- GCSS : Group (Internal) Communication Support Systems

- GISS : Group Information Support systems
- GXSS : Group External Communication Support Systems
- GPSS : Group Performance Support Systems

This classification includes the tools that support group tasks, i.e., communication support, process structuring and information processing, under its four categories. A detailed description of these four categories is provided below.

### **3.3.3.1 Group Communication Support Systems (GCSS)**

Electronic systems that facilitate communication within workgroups have been called Group Communication Support Systems. These have also been labeled as electronic meeting systems (EMS).

GCSS is used by group members who are spatially separated from each other - in different buildings, different cities, and different countries or merely in different rooms - to communicate with each other. The literature mentions six types of GCSS. These are interactive synchronous video systems (video walls, video-conferencing and videophones), non-interactive video (video tapes or laser disks), telephone conferences, voice messaging, interactive computer conferences, and non-interactive text /graphics (e-mail, conferences).

There are pros and cons of these systems as mentioned in the literature. While these systems allow team members to meet functionally, these electronic communication systems reduce the modalities, i.e., auditory, visual, "back-channel" and informal aspects by which group members communicate with each other. The extent of decrease in these modalities depends on the particular communication system used by the group. These reductions in modalities are the inevitable consequence of distributed groups and there can be mixed reactions as far as group effectiveness is concerned (McGrath and Hollingshead, 1994).

### **3.3.3.2 Group Information Support Systems (GISS)**

The GISS category includes systems that enhance sharing of the group's knowledge or information base. These types of systems allow virtual teams to share knowledge and information that has been gathered from sources other than on-line communication with



group members. These extra-group sources include quantitative databases such as sales records and cost data, and qualitative databases like information stored in libraries and archives, and minutes of the previous meetings. Some examples are program and object repositories, workflow systems, and project portals or dashboards.

McGrath and Hollingshead (1994) suggest that making information available to other group members through a GISS is not an easy task as it takes time and effort to acquire, store and retrieve information and additional effort to disseminate it to others.

Another negative feature is that of information overload since GISS may easily become a victim of runaway information dumping by members, making it difficult for group members to catalog and retrieve current information and information that is relevant to them. A similar result was apparent in a study by Majchrzak and Associates which found that a feature-rich collaboration and knowledge management tool was only used by the project groups as an elementary information storage system (Robey et al., 2004).

#### **3.3.3.3 Group's External Support System (GXSS)**

GXSS technologies deal with communication by group members with individuals and groups outside the reference workgroup. This function is a special case of both the GCSS and the GISS and can be done by using any of the three combinations of modalities (video, audio, text and graphics) and any of the patterns of spatial and temporal distribution. There has not been much research on the external communication function.

#### **3.3.3.4 Group's Performance Support Systems (GPSS)**

Electronic systems that provide direct performance support for groups incorporate an array of modules, each of which structures a different subset of a group's tasks. McGrath and Hollingshead (1994) provided an example of a typical system that might include the following tools/modules:

- Electronic brainstorming and other unstructured communication within the group
- Issue tracking
- Software defect tracking and root cause analysis
- Policy and standards formulation
- Requisitioning and managing assigned infrastructure and resources

According to McGrath and Hollingshead (1994), these GPSS facilitate some or all of the group's task activities like setting an agenda, identifying problems, generating alternatives, choosing among alternatives, and negotiating consensus with others.

### **3.3.3.5 Review of Tools**

This section provides a review of tools that facilitate team processes and aid in software development. This review is selective and is designed to provide examples of the capabilities of these tools.

Tools for virtual meetings: A virtual meeting place is a space that is set up by virtual teams on the web or Intranet to share documents and discuss ideas. It can also be used to organize meetings in a synchronous or asynchronous manner. These spaces include all documents created during the project, discussion archives, calendars, bulletin boards, and timelines as well as communication tools such as e-mail, instant messaging and videoconferencing. These virtual offices are available either as an in-house or a hosted product. The software combines communication, document management and project management features into a customizable virtual-team portal. Some software programs in this category include eRoom by Instinctive technology, Webex Meeting Center by Active Touchservices, Group Systems by GroupSystems Corporation (formerly Ventana), Netscape Virtual Office by Netopia, and Netmeeting from Microsoft.

Tools for team work: Collaborative technology is used to accomplish teamwork with geographically dispersed teams. It is like an electronic room that is closed after the work is done and the team is dispersed. Some software programs in this category include Instant! Teamroom from Lotus, Team Agenda, Teamspace from Invol corp., and Webtop Information Server from Kureo Technology Ltd.

Tools for project management: Project teams can access a special website as a hub for the project. Project teams can simultaneously access schedules and reports, delegate and accept tasks, manipulate information, submit status reports, enter timesheets and give instant feedback. Some software programs in this category include Active Project from Framework technologies, Teamcenter 2.5 from Inovie software, PLANVIEW 5.3 from PLANVIEW Software, and Project from Microsoft.

Please refer to Appendix A for a summary of other selected collaboration software products.

Research on collaborative tools indicates that the tools available in the market primarily serve the same broad range of functions that are indicated above. Giffin (2002) has also provided technical and organizational attributes of different Internet-based applications mentioned above. E-mail is best suited for one-to-one communication and can be used with many people with some success. Static websites are useful for dissemination of static information to large groups. Web-based groupware is best suited for structured communication allowing two-way interaction. Discussion groups are best suited for allowing large unassociated groups to follow a topic or thread of interest. Audio/Video conferencing is best suited for interactive communication of complex information between two parties. Text conferencing is useful for interactive communication between larger/more diverse groups where audio video conferencing is not feasible.

The literature also acknowledges the fact that these tools can not replace face-to-face and informal communication. According to McGrath and Hollingshead (1994), various types of electronic communication permit group members to "meet" functionally even when they are physically dispersed and operating at different times. However, it precludes or reduces the set of modalities like auditory, visual, non-verbal, and para-verbal. The extent of the reduction of these modalities is dependent on the particular GCSS in use. Other research has suggested that groups that use electronic networks extensively exhibit more co-operation and communication compared to groups that rely primarily on face to face collaboration (Easley et al., 2003).

### 3.4. Summary of Literature Survey

The findings from the literature survey were organized and presented along the lines of the theoretical framework in the preceding sections. Recurring threads of issues fitting into the broad categories of communication, coordination and control were reported in the literature as a result of spatial, temporal and cultural dispersion. Literature findings related to remediation of these issues were then categorized along the three axes of methodological processes, team practices and technology. The main findings from the literature survey are summarized in Table 2.

ISSUE	
Critical aspects	Solutions
<b>COMMUNICATION</b>	
<ul style="list-style-type: none"> <li>• Informal communication and exchange of ideas is restricted.</li> <li>• Reduced productivity because of inefficient communication.</li> <li>• Errors and rework because of semantic inconsistencies.</li> <li>• Restriction on information sharing because of fears about information security.</li> <li>• Miscommunication due to cultural differences</li> <li>• Ineffective meetings that fail to reach a decision</li> <li>• Reduced contact because of time zone differences</li> </ul>	<p><b>Methodological Practices</b></p> <ul style="list-style-type: none"> <li>• Developing a set of common "work products" and "vocabulary"</li> <li>• Well defined communication plan</li> </ul> <p><b>Team Practices</b></p> <ul style="list-style-type: none"> <li>• Specialized liaison managers</li> <li>• Frequent face-to-face meetings</li> </ul> <p><b>Technology</b></p> <ul style="list-style-type: none"> <li>• Use of collaboration tools and technologies and rich media</li> </ul>
<b>COORDINATION</b>	
<ul style="list-style-type: none"> <li>• Disparities in the understanding of overall objectives and issues</li> <li>• Loss of information during transfer of work units</li> <li>• Pacing of work between teams</li> <li>• Group cohesiveness and teamwork</li> </ul>	<p><b>Methodological Practices</b></p> <ul style="list-style-type: none"> <li>• Centralization of project management function and decision making</li> <li>• Use of the Project model - end-to-end responsibility for a module to one team.</li> <li>• Functional area model – work allocation based on clearly demarcated expertise areas of teams</li> <li>• Specialized coordinators</li> <li>• Early integration of components</li> <li>• Frequent and detailed sharing of objectives, status and issues</li> </ul> <p><b>Team Practices</b></p> <ul style="list-style-type: none"> <li>• Requirement clusters – End-to-end responsibility to a team to deliver an entire requirement cluster.</li> <li>• Team structures aligned with product architecture</li> <li>• Coherent and colocated teams</li> </ul> <p><b>Technology</b></p> <ul style="list-style-type: none"> <li>• Use of workflow management tools, shared source and documentation repositories and other GCSS, GISS, GXSS, GPSS tools</li> </ul>
<b>CONTROL</b>	
<ul style="list-style-type: none"> <li>• Project management challenges</li> </ul>	<p><b>Methodological Practices</b></p>

<ul style="list-style-type: none"> <li>• Change management is difficult</li> <li>• Leadership challenges and soft issues</li> </ul>	<ul style="list-style-type: none"> <li>• Object-oriented project management</li> <li>• Frequent deliverables, reviews and client signoffs</li> <li>• Prototyping</li> <li>• Risk management planning</li> <li>• Process-orientation and quantitative metrics – through CMM etc.</li> </ul> <p><b>Team Practices</b></p> <ul style="list-style-type: none"> <li>• Self-managing teams</li> <li>• Specialized training for managers/leaders</li> <li>• Upfront "Contracting Discussions" among teams and dispersed members</li> </ul> <p><b>Technology</b></p> <ul style="list-style-type: none"> <li>• Specialized distributed project management tools</li> </ul>
---	---

*Table 2: Summary of literature survey*

These findings will be examined using a CMM Level 5 assessed software development organization that specializes in global software development and software services delivery. As pointed out earlier in section 1.1, the CMM Level 5 assessed status of the company provides a reasonable degree of assurance that the company's software development processes are mature, have been independently assessed and that the company can quantitatively measure the success of these processes. This study aims to explore the integrated model that has been developed based on the available literature. It also attempts to uncover new insights related to best practices being followed in the industry in the area of global software development.

### ***3.5. Expected Findings from the Case Study***

The findings from the literature survey give rise to expectations of certain patterns to emerge from the case study. In general, dispersed projects should be using some of the above recommendations from literature. Specific expectations are:

1. More dispersed and complex projects would show a high usage of technology and collaboration tools in order to reduce the effects of dispersion.
2. Companies specializing in global development should be using very strong process frameworks that lay down work procedures and templates, communication plans, and shared project semantics and artifacts.
3. Some non-traditional approaches to project management might be observed, e.g., self-managing teams or specialized distributed management tools.

4. The integrated framework developed from the literature survey is useful for understanding actual practice and reflects the concerns and practices of global software development teams.

## **4. Research Method**

### ***4.1. Method of Study***

The empirical study was performed by conducting a case study of an IT consulting and services organization assessed at Level 5 of the Software Engineering Institute's CMM model. The selected organization has a long history of operating in the global development model. It has developed and evolved tools, practices, and methodologies to deal with some of the problems described in this thesis. This specific organization was chosen as the site for this study because the researcher had accessibility to senior management of the organization and could obtain the necessary support for conducting a detailed study. In addition, since this organization is a CMM Level 5, studying them was expected to provide insights on how institutionalized methodologies and practices can help alleviate some of the problems created by dispersion of teams.

### ***4.2. Profile of the Organization***

Momentum Technologies, Inc. is a Canadian company that specializes in information technology consulting. It was founded in 1999 and is privately held with Citi Group as a strategic investor. Momentum has its headquarters in Vancouver, British Columbia, and offices in the USA and India. Momentum's Software Development and Delivery Center in India has been assessed at SEI CMM Level 5 by KPMG and is also ISO 9001 certified.

Momentum provides offshore software services to companies that have incorporated outsourcing as a key component of their business strategy. A large part of Momentum's clientele is made up of North American software product companies and high-end electronics manufacturers. Momentum provides them outsourced software development, testing and maintenance services spanning the complete life cycle of products and applications. Their practice areas include requirements gathering, software design and development, quality assurance and testing, and application sustenance services.

Momentum also works actively in Canada to popularize the concept of global development among Canadian companies by sponsoring and participating in seminars, workshops and other events.

Momentum is a quality-driven company. It focuses on quality by making every team member in the company part of the quality processes and methodologies. Continuous process improvement is enabled through the Process Group (PG) and the Quality Assurance Group (QAG). The PG collates process improvement suggestions across the organization, evaluates them, pilots them, and makes modifications to the defined process based on the pilot efforts. The QAG subsequently verifies the implementation of these processes and provides feedback to the PG.

Through its quality focus, Momentum has been able to provide measurable improvement and significant benefits to its clients in the areas of reduced effort variance, reduced schedule variance, lesser defect density, and higher defect removal efficiency.

Momentum manages software projects for its remote clients primarily through a local project manager and a local technical consultant supported by teams at its offshore development centers.

### ***4.3. Data Collection Methodology***

Two sources of data collection were used. A questionnaire was used to collect descriptive information on the distributed projects that the respondents worked on, as well as on their use of specific tools, technologies and practices (see Appendix C for the questionnaire). The questionnaire was developed and used by Khazanchi and Zigurs in their 2005 study for the Project Management Institute. In addition to the questionnaire, participants were interviewed via e-mail and some of them were contacted via telephone for follow-up questions and clarifications. The interview was designed along the lines of the integrative model to facilitate comparison to the results of the literature survey that had also been similarly organized. Interview questions were about interviewees' concerns related to communication, coordination and control and specific practices adopted by them to address their concerns (see Appendix D for the interview questions).



The unit of analysis in this study was the project, which means that all questions were about a representative project that each participant worked on. All participants were answering about different projects, which provided desired diversity in their experiences. Based on the answers to the questionnaire, these projects were classified into Lean, Hybrid and Extreme projects. This classification broadly indicates the overall degree of dispersion, scope, risk and complexity of the project. The classification was based on the scheme developed by Khazanchi and Zigurs. The classification is not a specific part of the theoretical framework, but instead is intended as a way of describing projects and exploring differences between them.

A pilot study was conducted on three people from the software development field to validate that the questions were clear and unambiguous. There were no changes made in the final version of the questionnaire since the pilot respondents did not report any difficulty in understanding and responding to the questions.

The questionnaire and interview questions were circulated via e-mail to seventeen people that included test engineers, senior developers, team leads, project leads, and project managers. The questionnaire was first sent to the Vice President of Service Delivery and he forwarded the questionnaire to the respondents. Respondents were requested to send their responses directly to the researcher to avoid any bias in the study. Twelve people responded to the questionnaire and interview questions. Five of the respondents were contacted again through e-mail and telephone to get more information on some of their responses. All the respondents have been assigned participant ids to ensure anonymity.

#### ***4.4. Study Participants***

The questionnaire consisted of items related to the description of the distributed project on which the participants had worked, their role in the project, team composition and distribution, project risk, level of technical innovation, and use of technology. This section was examined primarily to categorize projects into lean, hybrid and extreme projects and also to see the use of technology by the participants. Table 3 below provides a snapshot of the kinds of projects on which the participants worked. These projects were

categorized as lean, hybrid and extreme by calculating project complexity, scope, risk, and success (refer to Table 5).

Participant ID #	Project Role	Description of the project
1	Developer	The system consolidates and integrates data from various systems and does proactive management and optimization of IT operations for large networks.
2	PM	Reverse engineering and design rediscovery of some modules
3	TL	Develop a platform for entering and tracking requests by customers.
4	Developer	Develop an intranet to provide web based collaborative tools
5	TL	Develop a complaint registration system in .NET
6	Dev	Develop an intranet Application
7	Tester	Develop a web based application
8	Dev	Develop a web based application
9	Dev	Development of a comprehensive Operations Management Solution
10	PM/BA	To replace manual "change request" to advance automation
11	PM	Develop software for marketing strategy development using the hit information related to the end customers' web sites and portals
12	TL	Develop an IT management product to monitor and manage integrated applications

*Table 3: Types of projects worked on by the participants*

The above table shows that a variety of projects were covered as part of this study and that they appear to vary in their degree of complexity. Also, the participants represent a full range of roles in a software project.

#### **4.5. Data Analysis Methodology**

A quantitative and qualitative analysis was performed on the data collected from the questionnaire and interviews. Quantitative data consisted of answers to the questionnaire where the respondents chose a numerical answer from a set of scaled choices. The responses were coded on a scale of 1 to 5 in an increasing order of importance, complexity or severity of the attribute being addressed by that question. Certain questions had three choices, for example, question #11 on cultural diversity (homogenous, hybrid or diverse). Such questions were coded as 1, 3 or 5 in the increasing order of the contribution of that attribute to dispersion, to ensure consistency in the numerical values that were assigned across all questions.

Qualitative analysis was based on interpretation of descriptive answers to interview questions within the framework provided by the literature survey. Each response was compared to each element in the theoretical framework and marked as either agreeing with, disagreeing with, or not mentioning the element. The responses were tabulated and totals calculated, thus showing the extent to which the interview responses agreed with or disagreed with the elements of the theoretical framework. The elements examined were the group process elements of communication, coordination, and control, and the methodological practices, team practices, and technology that support group process.

In the first step of quantitative analysis, each participant's answers to certain interrelated questions were combined to determine key characteristics or "concepts" of the project. The key project characteristics that were derived from the raw data are: project complexity, risk, scope, degree of technology use, and overall degree of project success (Khazanchi and Zigurs, 2005). Table 4 below shows the questionnaire items that were used to measure complexity, risk, scope, technology use, and project success. Appendix C shows a copy of the questionnaire.

Concept	Related Questionnaire Items	Blending Approach
Project complexity	#3 – <i>Team size</i> #11 – <i>Cultural diversity</i> #12 – <i>Language differences</i> #13 – <i>Proficiency in virtual team technology</i> #16 – <i>Overall complexity</i> #23 – <i>Availability of historical knowledge</i> #26 – <i>Resource availability</i> #27 – <i>Disparities in individual personalities</i>	Arithmetic mean – 1 to 5 – represents increasing complexity
Project scope	#4 – <i>Project Duration</i> #15 – <i>Overall Scope</i> #24 – <i>Level of Innovation</i>	Arithmetic mean – 1 to 5 – represents increasing size of scope
Project risk	#17 – <i>Programmatic risk</i> #18 – <i>Technical/Engineering Risk</i> #19 – <i>Quality Risk</i> #20 – <i>Logistical Risk</i> #21 – <i>Deployment Risk</i> #22 – <i>Overall Risk</i>	Arithmetic mean – 1 to 5 – represents increasing degree of risk inherent in the project
Technology Use	Degree of use of 16 different types of tools and technologies for collaboration, e.g., video conferencing, e-mail, instant messaging. Scored from 1 to 5, indicating "never used" to "almost always used".	Arithmetic mean – 1 to 5 – represents increasing degree of use of collaborative tools and technology
Project success	#6 – <i>Schedule</i> #7 – <i>Budget</i> #8 – <i>Goals and Requirements</i> #9 – <i>Overall Success</i>	Arithmetic mean – 1 to 5 – represents increasing degree of success

Table 4: Items used to measure complexity, scope, risk, technology use, and success

In the second step of the analysis, the projects were characterized as lean, hybrid, or extreme based on the scope, complexity and risk scores (Khazanchi and Zigurs, 2005). This characterization was done by averaging the scores for these three derived characteristics into an overall score. All three characteristics were given equal weight in this calculation. Projects scoring under 3.0 were classified as Lean. Projects scoring from 3.0 to 4.0 were classified as Hybrid. None of the projects scored more than 4.0 to qualify to be classified as Extreme.

Qualitative data came from detailed interview questions related to themes that emerged in the theoretical framework developed for this study. Some respondents were contacted by e-mail or phone for follow up to clarify their answers when there was such a need. The answers from each of the twelve participants were tabulated. Patterns in these data were identified by looking for consistent comments across all the replies. This analysis served to compare the themes found in the data to the themes from the literature survey (as summarized in Table 2). I also looked for responses that were either contrary to

findings from the literature survey or that provided new insights that were not found in the literature.

## 5. Findings and Discussion

### 5.1. Findings from the Questionnaire

The first step of the quantitative analysis was to merge the complexity, scope and risk scores of the projects to assign an overall project type to each project – Lean, Hybrid or Extreme, as shown in the table below. Six of the twelve projects were found to be Lean (total score of less than 2.99) and six were Hybrid (total score of 2.99 to 4).<sup>2</sup> None were Extreme. Appendix G shows the details of the individual scope, complexity and risk characteristics.

ID#	Scope	Complexity	Risk	Average	Project type
4	2.33	2.00	2.17	2.17	Type I: Lean
8	1.67	2.38	2.67	2.24	Type I: Lean
6	2.00	2.86	2.83	2.56	Type I: Lean
7	2.00	3.50	2.50	2.67	Type I: Lean
11	3.00	2.50	2.50	2.67	Type I: Lean
12	2.33	3.63	2.67	2.88	Type I: Lean
10	2.33	3.63	3.00	2.99	Type II: Hybrid
3	4.00	2.88	2.17	3.01	Type II: Hybrid
2	2.67	2.75	3.67	3.03	Type II: Hybrid
9	2.67	3.63	2.83	3.04	Type II: Hybrid
5	3.67	2.75	3.50	3.31	Type II: Hybrid
1	3.67	3.29	3.00	3.32	Type II: Hybrid

Table 5: Actual scores of complexity, scope, risk and overall project type

Table 6 below shows the calculated degree of success for each project. Details of the success factors are shown in Appendix G.

---

<sup>2</sup> The characterization of projects as Lean or Hybrid is based on and similar to Khazanchi and Zigurs (2005). Project #10, which had an average of 2.99, was so close to the 3.01 score for project #3 that it made more sense to identify it with the Hybrid types rather than the Lean types.

ID#	Project type	Degree of Success 1=Failure, 5=Extremely Successful
1	Type II: Hybrid	4.67
2	Type II: Hybrid	5.00
3	Type II: Hybrid	4.50
4	Type I: Lean	4.00
5	Type II: Hybrid	5.00
6	Type I: Lean	5.00
7	Type I: Lean	4.00
8	Type I: Lean	4.25
9	Type II: Hybrid	3.67
10	Type II: Hybrid	5.00
11	Type I: Lean	4.00
12	Type I: Lean	3.67

Table 6: Overall success ratings of each project

Table 7 below shows the calculated degree of technology use for each project. Details of the 16 different technologies are in Appendix G.

ID#	Project type	Technology Use 1=Non-existent, 5=Very High
1	Type II: Hybrid	2.38
2	Type II: Hybrid	1.81
3	Type II: Hybrid	1.75
4	Type I: Lean	1.94
5	Type II: Hybrid	2.25
6	Type I: Lean	2.06
7	Type I: Lean	2.69
8	Type I: Lean	2.50
9	Type II: Hybrid	2.69
10	Type II: Hybrid	1.75
11	Type I: Lean	2.19
12	Type I: Lean	2.75

Table 7: Project type and use of technology

Finally, the overall technology use and overall degree of success were grouped by the type of project to aid in analysis. Table 8 below shows the calculated degree of technology use and overall project success scores for each of the projects.

ID#	Project type	Technology Use	Degree of Success
4	Type I: Lean	1.94	4.00
6	Type I: Lean	2.06	5.00
7	Type I: Lean	2.69	4.00
8	Type I: Lean	2.50	4.25
11	Type I: Lean	2.19	4.00
12	Type I: Lean	2.75	3.67
<b>Average for Lean Projects</b>		<b>2.35</b>	<b>4.15</b>
<b>Std. Dev. for Lean Projects</b>		<b>0.34</b>	<b>0.45</b>
1	Type II: Hybrid	2.38	4.67
2	Type II: Hybrid	1.81	5.00
3	Type II: Hybrid	1.75	4.50
5	Type II: Hybrid	2.25	5.00
9	Type II: Hybrid	2.69	3.67
10	Type II: Hybrid	1.75	5.00
<b>Average for Hybrid Projects</b>		<b>2.10</b>	<b>4.64</b>
<b>Std. Dev. for Hybrid Projects</b>		<b>0.39</b>	<b>0.52</b>

*Table 8: Project type, use of technology and project success*

The above table presents an interesting finding. Lean projects had a higher technology use than hybrid projects. This is a counterintuitive finding since technology has the potential to make a significant contribution to supporting highly complex and virtual projects. One explanation for this finding is that senior project personnel in this study were found to be significantly less intensive technology users than developers and testers (as reported later in this section). I had only two respondents in leadership roles in the lean project subset (1 manager and 1 team lead) compared to 4 junior positions (developers and testers). On the other hand, the hybrid project sample set had four senior personnel (2 managers and 2 team leads) and only 2 developers. Managers and team leads in our sample set primarily used email, telephone, instant messaging and conference calling (in that order), whereas the more junior team members used a wider range of tools, most significantly, joint document editing tools and workflow tools.

Use of technology does not appear to affect project success. This result may be due to the small size of the data set, or use of complex tools and technology may be adding to project complexity and the learning curve.



Participants were asked about the use of sixteen tools and technologies in their projects. The diagram below shows the prevalence of these tools and technologies within the respondent group<sup>3</sup>.

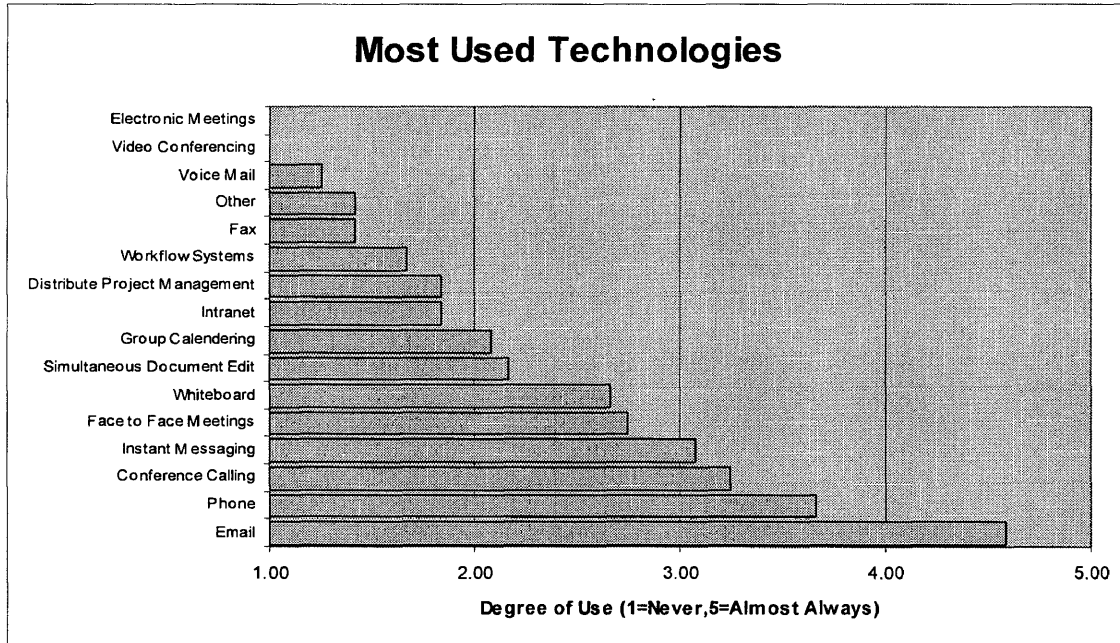


Figure 2: Use of Collaboration Tools and Technology

This diagram clearly illustrates that traditional technologies like email, phone and conference call have the highest usage. Whiteboards and instant messaging are also preferred whereas other newer technologies have low to medium use.

Although the unit of analysis for this study is the project, the data on technology use at the individual level also reveals the pattern of technology use by project team members in different roles, as the following table shows.

Role	Technology Use 1=Non-existent, 5=Very High
Tester	2.69
Developer	2.31
Team leader	2.25
Project Manager/Business Analyst	1.92

Table 9: Technology use by role in project

<sup>3</sup> All these technologies were available for use within the company.

This table shows that use of technology is higher for testers and developers. Team leads and project managers use technology less compared to the technical member of the team. Further, senior staff rely more on more traditional methods such as telephone, face to face and email.

## **5.2. Findings from Interviews**

### **5.2.1. Issues Faced**

This section includes items related to the findings from the literature survey. Findings from the survey were analyzed to identify emerging themes and how these findings relate to the findings from the literature survey. The following table provides the items included in this section.

- |  |
|--|
| <ul style="list-style-type: none"> <li>• Specific problems experienced in their project with respect to communication among the members of the virtual team.</li> <li>• Specific problems experienced in their project with respect to coordination during the virtual project.</li> <li>• Specific problems experienced with respect to control of the virtual project.</li> <li>• Specific methodology practices used to make the project a success.</li> <li>• Specific team practices used to make the project a success.</li> <li>• Specific technologies and ways that one has used these technologies to make the project a success.</li> <li>• Practices that were used but not found helpful in their project.</li> </ul> |
|--|

*Table 10: Focus of interview questions*

Responses to each item from all twelve participants were summarized and tabulated to analyze in detail how they related to the theoretical framework developed for this study. Table 11 below shows the extent to which each element of the framework was mentioned in the interview data. A checkmark shows that the element was mentioned by that participant as being important. An "x" indicates that the participant disagreed with the issue. A blank means that the issue was not mentioned by that participant. Thus, Table 11 shows whether and to what extent the interviews reinforce the literature findings.

Issue	Participant												Count
	1	2	3	4	5	6	7	8	9	10	11	12	
<b>Communication</b>													
1. Informal communication and exchange of ideas is restricted			✓			✓							2
2. Reduced productivity because of inefficient communication	✓	✓				✓	✓		✓		✓	✓	7
3. Errors and rework because of semantic inconsistencies													0
4. Restrictions on information - information security concerns													0
5. Miscommunication due to cultural differences		✓							✗	✓			2
<b>Coordination</b>													
1. Disparities in the understanding of overall objectives		✓	✓				✓						3
2. Loss of information during transfer of work units	✓							✓					2
3. Pacing of work between teams	✓								✓	✓		✓	4
4. Group Cohesiveness and team work							✓		✓				2
<b>Control</b>													
1. Project management challenges		✓	✓	✗	✓		✓	✗	✓	✓	✗	✗	6
2. Change management is difficult													0
3. Leadership and soft issues													0

✓ - Agree ✗ - Disagree blank – No Response

Table 11: Problems discussed in literature and their occurrence in the sample projects

Table 12 below shows issues that were mentioned by the respondents that were not in the theoretical framework, that is, additional problems that add to the model that was derived from the literature.

<b>Issue</b>
<b>Communication</b>
Difficulties in understanding accent
Keeping track of additional e-mail volume is difficult
Need to work during holidays and off-hours
Lack of motivation among team members to take on additional challenges of dispersed projects
<b>Coordination</b>
Delays in getting remote resources allocated
Overhead of documentation and communication in explaining even small details
Difficulty in scheduling meetings that fall within working hours of all team members
Coordinating with client resources i.e. domain experts and testers. More difficult to work with them remotely (as compared to own resources located offsite).
<b>Control</b>
Delays in getting responses to status inquiries from remote team members
Hard to work around skill gaps of remote team members (as compared to managing skill shortfalls of local workers).
Gradual loss of morale because of having to work unconventional hours to have work time overlap with the remote team

*Table 12: Additional problems reported in the sample projects*

#### **5.2.1.1. Communication**

The diagram below presents the main communication issues discovered through the interviews. Some of the issues are very similar to the findings of the literature survey. Participants also provided some interesting experiences from real-life situations that illustrate problems related to communication styles and cultural context. One interviewee provided a specific illustration of the difference in communication style in global teams.

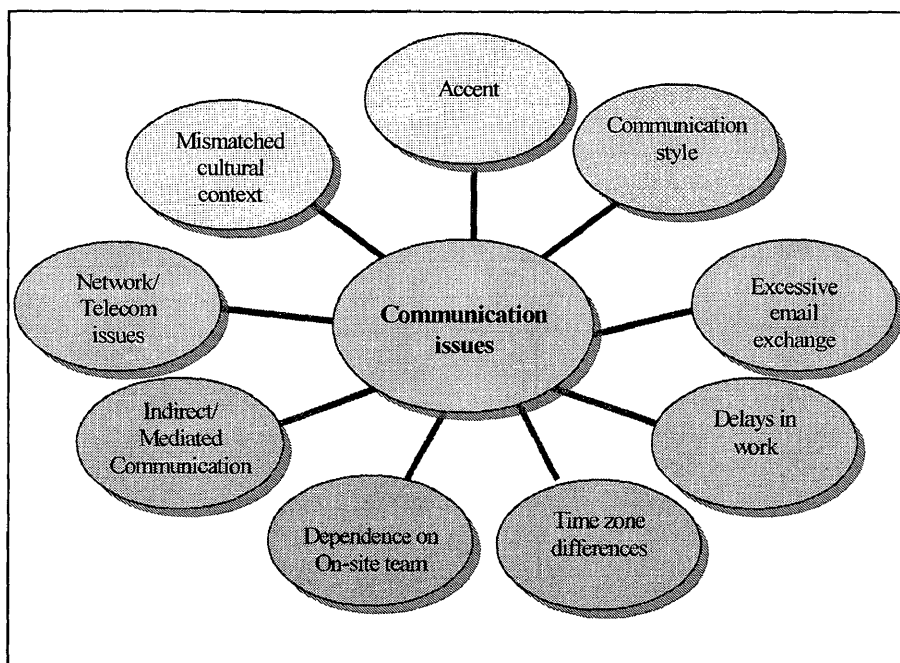


Figure 3: Communication issues found through the Interviews

### Communication and cultural context: an illustration

The following quotes from the interview data illustrate communication and cultural issues:

*"English being the second language for most team members was not their natural language of thought. This reflected in the e-mails being structured rather incoherently at times and required substantial rewrite effort to bring the points out more clearly. The style also tends to swing towards too casual when attempt is made to be natural in expression." [Participant #2]*

*"During telephonic conferences, the members tend to get nervous, at least during the early phase of their careers. This would affect the customer's confidence in ability to deliver. The usage of colloquialism by customers sometimes confuses the team members at offshore. Terms like "dangling in the air", "caught in the headlights" and "stepping up to the plate" are not very easy to comprehend contextually for many Indians. Sometimes, the remarks made in the lighter tone remain unappreciated because of the lack of awareness of the cultural background." [Participant #2]*

The most common communication-related problem that was reported was reduced productivity because of inefficient communication. Some other findings from the literature survey like reduced informal communications and exchange of ideas were very similar to the findings from the study. Another observation that clearly emerged was

that time difference in the global teams introduces a number of problems for the team members, ranging from delay in critical tasks by 24 hours and a false image in the mind of the client that the team is working 24 hours for them.

Another interesting observation is related to technology. Technologies like e-mail and instant messaging were among the most used technologies by participants of this study (see Figure 2). One participant reported that use of these technologies also resulted in problems like difficulty in keeping track of all the mails and messages, inability to discuss urgent issues immediately and need to be online in non-working hours. Participants also reported specific problems related to the telecommunications network. Some specific examples included difficulty in getting a connection, voices not being clear and frequent disconnections.

One more communication problem reported by the interviewees was due to the presence of an additional layer in the form of an on-site person to coordinate the two teams. Participants viewed that person as adding complexity and dependence in their tasks. On-site coordinators were typically added to remote projects to be the focal points for communication between customer and off-site teams. Their presence was expected to facilitate communication, provide a local single-point-of-contact to the client and to insulate the client from communication issues of language barriers or cultural mismatch with less-trained off-site personnel. However, the use of on-site coordinators appeared to introduce some new problems.

One respondent provided three example scenarios related to this issue:

*"Scenario 1: we would like to ask a our client a query but sometime our on-site team intercepts the query, tries to resolve it and that response might not be the exact match with the customer's.*

*Scenario 2: We ask a query and they forward it to the client then we get delays due to this extra loop.*

*Scenario 3: Sometimes they modify client's response and respond back with what they understood. Sometime it causes issue of miscommunication."* [Participant #1]

Most of the communication issues mentioned above were for the teams that had a large time zone difference where communications were often delayed. Large teams exhibited more communication-related issues, possibly due to the excessive convergence of communication channels into one or two coordinators. Two participants who were working on a smaller team reported fewer communication-related problems. In their

view, language was never a problem as all the team members offshore and on-site came from the same cultural background. These participants also reported using instant messaging and e-mails to share knowledge.

### 5.2.1.2. Coordination

The following diagram presents the coordination-related problems that emerged from the study. These problems broadly include project scope concerns, technical issues, schedule-related issues and issues of time zone difference.

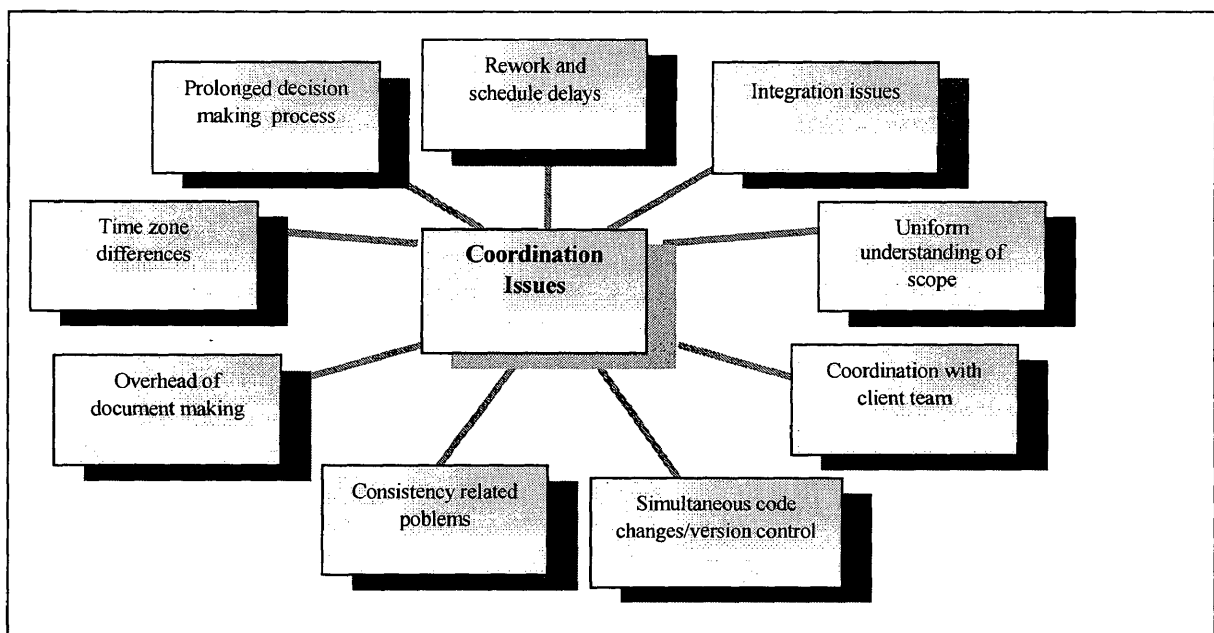


Figure 4: Coordination issues found through Interviews

Most of the findings are consistent with the findings of the literature survey. Some of the responses provided a sense of the gravity of the problem. Scope control emerged as a dominant concern of the respondents. Analysis of two of the responses clearly suggested that absence of face-to-face communication led to problems related to scope control. Following are the responses of the two respondents.

*“As you are not discussing face to face, each piece of information has to be written in detail so that there is no communication gap and no information is left out. It would be definitely less time consuming if you are sitting in front of the customer as you can discuss it right then*

*and there and make a small one page note and get it signed as both the parties understand what do they mean”. [Participant #3]*

*“Often there is a mismatch of thoughts in customer's mind about what we are supposed to, or not supposed to, implement as part of the work. Unverified assumptions lie at the root of this problem”. [Participant #2]*

The most frequently-reported problem was related to pacing and synchronization of work between teams. Other coordination-related problems mentioned include prolonged decision making process, difficulty in understanding the expectations from the team and language-related issues. One respondent said that team members at times have to work in their non-working hours due to time zone differences.

Another coordination-related problem that clearly emerged from the study was related to the actual component of the work. Specific examples include two teams working on the same code repository and interdependent code leading to rework, schedule delays and integration issues.

#### **5.2.1.3. Control**

Respondents did not indicate many issues related to control in global projects other than challenges faced by project managers. Most of these issues matched the literature findings. The study results indicated issues related to rework, schedule delays and quality control. Other team management-related issues included monitoring the team members who are placed on-site. It was difficult for the managers to authenticate the time spent by the members on the project.



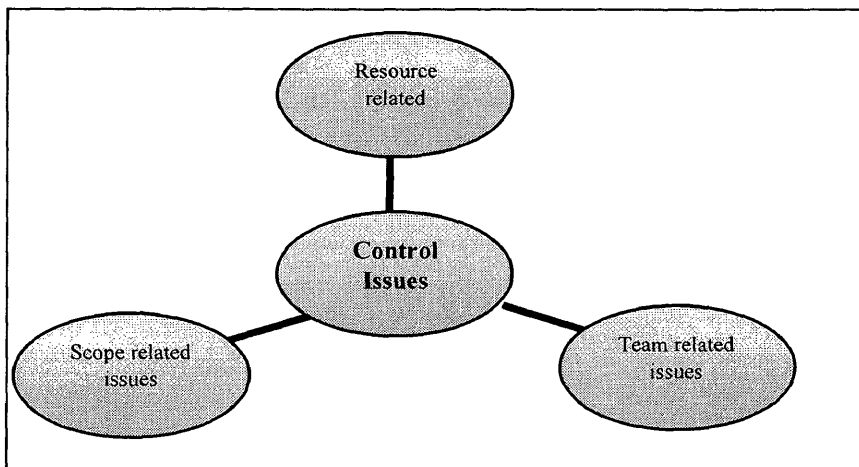


Figure 5: Control issues found through Interviews

Another interesting observation was with regards to scope control. Problems related to control were very similar to the problems related to coordination. The following quote illustrates the problem of scope control.

*"Managing client expectations was difficult at times – there were a number of feature requests after delivery that was clearly out of scope (from our perspective) but the client felt they were within scope. The client also posted bugs that were related to their testing environment and not our application."*  
 [Participant #10]

Another observation was with regard to team size. The results indicated that people who were working in small teams had no problems with coordination and control. Interestingly, a project manager also expressed frustration with regard to team management of the client's personnel. His response reflected his inability to work effectively with the remote client's team members which in turn caused delay in their work.

In the verbal interview, two respondents brought up another issue – that of gradual loss of morale because of having to work during unconventional hours on an extended basis so as to achieve some work time overlap with the remote team.



Strategies Recommended in Literature	Participant ID												Count
	1	2	3	4	5	6	7	8	9	10	11	12	
"Contracting Discussions" among teams and members							✓						1
<b>Technology</b>													
Use of collaboration tools and technology and rich media	✓		✓					✓	✓		✓	✓	6
Use of work flow management tools		✓											1
Specialized project management tools			✓										1

Table 13: Strategies recommended in literature that are actually in use in the company

The interview also revealed certain techniques and practices that had not been identified in the literature survey, as shown in table 14 below.

Practice/Strategy	Benefit
<b>Methodological Practices</b>	
Electronic log sheets to log and trace all communication between teams	Improve accountability, traceability and control
Rotation of tasks to prevent continuous shift-work	Prevent fatigue and loss of morale from continuously working unconventional hours in order to have work time overlap between remote teams
<b>Team practices</b>	
Joint review of work products by on-site and off-site teams	Promote a feeling of joint ownership and responsibility. Promote trust among teams.
Knowledge sharing among teams through special sessions	Homogenize the knowledge base of the distributed teams and reduce skill imbalances. Give a better picture of overall project objectives to all teams.
Even allocation of responsibilities to create a sense of participation, accountability and joint ownership	Address the feeling of disconnection felt by the teams that are away from the power center.
Rotation of work among team members to reduce dependencies and build redundancy in case of communication failure	Address the risks associated with general communication breakdowns and inability to make contact with specific members during crisis.
Rescheduling of work hours to provide overlapping hours	Reduce the effects of temporal dispersion.
<b>Technology</b>	
Log all internet messaging communications into formal minutes of meeting.	Prevent loss of information resulting from ad-hoc electronic chatting instead of formal meetings.

Table 14: Strategies that had not been found in the literature survey

### 5.2.2.1. Methodological Practices

The results indicated that methodological practices followed in the company were closely related to the practices discovered in the literature. The most common practice cited was

the use of structured CMM level processes to standardize and improve group work (6 respondents). This was followed by the practices of dividing work into smaller units with frequent deliveries and use of prototyping and early integration to prevent problems of inconsistency/incompatibility. One other common practice clearly evident from the interviews was scope signoff at the beginning of the project.

### **Specific Practices**

One respondent mentioned following specific practices at the various stages of development:

- UI mockups and requirements traceability matrix to ensure requirements are clearly understood
- Low level design document, Unit and integration test specification and QA specifications for design, development and testing signed off at the beginning
- Tight change request control, scope-creep control and risk planning.

Some other specific practices included frequent interim deliveries to the client, planning additional training and ramp-up time for each member and having shared repositories of code and documents.

### **Frequent checkpoints with the client**

The results revealed that frequent requirement specification and design checkpoints with the client help in ensuring that the client has seen, reviewed and approved all artifacts. This helped the team to reduce iterations of rework.

### **Use of log sheets (Queries and Assumption sheet)**

Log sheets are used by the teams to log all the communication with the client and on-site team members. It is primarily used to keep track of queries and responses. This log sheet contains information like what the query was, when it was initiated, who initiated it, who responded to it (on-site team or Client), on what date and what the final action point was. According to the respondents, this kind of tracking helped them in controlling miscommunication and resolving contentious disputes.

### **Frequent team trainings**

Team training was mentioned by a significant number of respondents. The main focus of this training was knowledge sharing and knowledge transfer. The training was also sometimes customer-led to train new remote members on the project.

### **Self-managed progress tracking and work scheduling**

Respondents mentioned metrics and techniques like 'steps of doneness', self-status tracking, earned value calculation, introduction of infra track, and issue sheets. All of these techniques were intended to make the teams/team members more and more self-managed. "Steps of Doneness" specifically was a checklist of steps given to developers to be followed to calculate how much of a task had been completed for consistent reporting across the team. One specific example from the respondent included ten such steps. Self-Status tracking was done with the use of shared spreadsheets that local and remote members could update themselves. Earned value calculation with the help of percentage assigned to each task indicated the amount of work done for each item. All these methods decreased the need for verbal communication for status gathering.

### **Issues Sheet**

There was a shared "Integration Issues" template that was specifically used to track technical and integration dependencies between modules developed at different locations.

### **Use of CMM Level 5 Assessed Processes**

Strong, standardized, institutionalized processes remove ambiguities and increase consistency and repeatability. Strong processes reduce dependence on individuals and mitigate the effects of distance and absence of strong interpersonal relationships. The CMM's strong focus on quantitative measurements and continuous improvement gives the company an edge in identifying, analyzing and addressing problem in its processes. There is an emphasis on collecting, recording and sharing organizational knowledge and experiences. This allows each new project to learn from and build upon the experiences and learning from past projects.

### **5.2.2.2. Team Practices**

Frequent team training and meetings clearly emerged as an established practice. The primary objectives were co-planning of short-term objectives (like a "contracted discussion") and knowledge sharing. Team members used overlapping work hours for knowledge transfer sessions. Data also revealed that team leaders and project managers had more frequent formal team meetings to check status and communicate issues. Team members were also given more responsibility than what their role demanded to keep them motivated (to accept the extra workload and challenges of working in a distributed environment), and also to make them more self-managed. Work was often rotated among the team members to reduce the dependencies on individuals and to cross-train, as a risk mitigation strategy in case there is a communication failure between teams or between a team and a remote member.

### **5.2.2.3. Use of Technology**

Some of the technologies mentioned by the respondents that facilitated global development are J2EE Design Patterns, Lucene, Hibernate, Enterprise Java Beans, CVS, Bugzilla, and Microsoft Project Plan. Specific examples included using remote desktop access to hold application walkthroughs to help in rapid completion of knowledge transfer.

Off-site teams were often connected to the remote client's network through VPN. This allowed them to operate from a remote location in a near-seamless fashion. Shared source code repositories were also used extensively to improve coordination. Technologies like internet chat, teleconference and e-mail were ubiquitous.

Project managers were aware that excessive use of technology and specialized tools added to the complexity level and may become an overhead. Two respondents stated that they had tried using simple home grown tools for defect tracking and for code synchronization but quickly found that the simple approach did not work in a multi-location setting. Both had to revert to specialized software solutions for this activity.

#### **5.2.2.4. Practices not found useful**

Very few respondents could recall any practices that had been tried but had proven unsuccessful. Unsuccessful practices mentioned were:

Self-directed learning approach for project training needs of remote members – Electronic training and background material about the project objectives, design and coding standards was prepared at the main project location and given to new remote members to study on their own. Such remote training was found ineffective. It was found that face-to-face meetings and physical training sessions were necessary for training remote staff. This required periodic travel by senior members to the remote location to train batches of new staff.

Homegrown control tools to reduce complexity – One project tried using simple shared Excel sheets for defect reporting and tracking to keep this activity simple. However they had to revert to a complex specialized defect tracking software. They found that it was difficult to track defects without a special-purpose tool in a multi-location scenario where defect originators (programmers and designers), defect reporters (QA analysts/testers) and defect monitors (team leads, project manager, and customer) were at different locations. Another project tried to use daily manual source code synchronizations across sites because the shared source repository was complex and was slow over the network. They quickly realized that the manual synchronization was introducing inconsistencies and they had to fall back to using a shared repository and accept the complexity and latency.

UML Artifacts - One respondent found that UML artifacts did not help in clarifying requirements with a remote customer because the customer did not understand UML and insisted on more traditional devices.

## 6. Conclusion and Implications

### 6.1. Conclusions

This study found that many of the communication, coordination and control related issues in global software development that have been reported in the large body of academic research are indeed echoed in practice, at least in the single organization that was investigated in this study. However, many of the issues that literature predicts were not found to be of significant concern in the target organization. The most striking inconsistencies were found in the following areas:

1. Higher error rates and rework because of semantic inconsistencies were not reported
2. Change management was not reported to be an issue
3. Restrictions on information flow because of information security concerns were not reported to be an issue

Institutionalized use of highly-evolved and mature work practices appears to be the major factor that is attenuating the first two issues. Indeed, process-orientation, quantitative metrics and use of CMM processes was the most frequently-used strategy quoted by the respondents for ensuring success of distributed projects.

To understand the third observation, one needs to look at the ubiquitous use of distributed software development, the vast strides in network security over the last few years and the increased business imperatives to drive down costs of IT development. It appears that as global software development has become more of a norm than an exception, the old barriers of suspicion, fears of breaches of security and concerns about theft of intellectual property have been significantly allayed. Rapid technological advancements in secure data transmission using multi-point security, tight encryption and secure virtual private networks have contributed to this trend.

With these structural problems having been addressed to a great degree, the main area of concern now is the operational side. Operational issues include delays in communication, cultural differences, pacing of work, management and control issues, team coordination, and cross-training of teams.



Dispersed software development teams depend heavily on technology to overcome the effects of dispersion. However, the tendency is to use those technologies that provide a quick, practical, uncomplicated and inexpensive surrogate to face-to-face communication. For example e-mail, internet messaging/chat, whiteboards and selected collaboration tools are preferred to highly media-rich technologies like video conferencing.

Some of the specific expectations listed in section 3.5 were not met. Traditional project management techniques and tools were being used. Specialized distributed management tools or concepts like object-oriented project management or self-managed teams were not found. However, the theme of team-empowerment and increased responsibility did emerge, but not to the extent of fully self-managing teams.

No consistent relationship was found between high use of collaborative technology and project complexity or project success.

## **6.2. *Recommendations to Managers***

The most significant lesson from this study is that a strong process framework can considerably reduce the negative effects of dispersion in global software projects. A strong process framework reduces dependence on individuals, and on interactions between individuals, and enforces a systems approach on the software development process. It makes it easier to achieve clearly demarcated, input-output based coupling between dispersed teams and individuals and encapsulates the complexities of individual tasks into well-defined and well-documented units of work that are easier to monitor, coordinate and control. A strong process framework lays down responsibilities, work templates and standards, shared semantics and artifacts, performance criteria, measurement techniques, and other important project parameters and makes it easier for the project manager to orchestrate well-integrated, efficient and error-free collaboration between dispersed teams. A framework like CMM/CMMI with an emphasis on continuous improvement (at Level 5) promotes organizational knowledge management and knowledge sharing and helps organizations that are new to global development to evolve effective processes faster.

While evaluating the use of collaborative technologies for distributed projects, managers must be cognizant of the fact that technology does bring with it an added cost in the form of complexity. They must strike the right balance between the efficiencies that technology can add and this extra burden of a steep learning curve.

This study revealed knowledge-sharing and cross-training between remote teams to be an area in which managers should expend a lot of effort. It was found that face-to-face contact and traditional training (as opposed to electronically-delivered training) is valuable.

Specifically in project teams that work across wide time zone differences with certain teams changing their work hours to achieve overlap, it was found that the job design should accommodate plans for frequent job rotation in order to avoid fatigue and loss of morale.

Managers must coach and empower their staff to be more self-directed and self-managed in order to compensate for the reduced opportunity for direct supervision and control.

### ***6.3. Limitations of this Study***

There were no Extreme projects in the sample, but the variation between Lean and Hybrid did help to show some differences. In addition, the projects varied in terms of their focus. Although this variation might explain some of the differences in management practices, it is generally reflective of the types of projects that are relevant for the focus of this study. This study was based on a very small sample set limited to one organization and it was not possible to identify significant correlations among various parameters. The study can still be used as a starting point for more in-depth research on this topic that will allow detailed statistical analysis.

Project success measure was collected through a self-report by participants rather than through an independent assessment of project outcomes against success criteria. This introduces a certain degree of bias in this metric. Further, rating of the degree of project's success provided by non-managerial technical staff may not accurately

represent the success as measured by the company or by the client. However, self-reporting is a very common way of reporting and has been found successful as a data-collection method for research.

#### **6.4. Future Area of Research**

A larger and more varied sample might have given us insights into the following questions, which point to future research:

- Does use of technology significantly improve the probability of success of a distributed project?
- Does industry recognize this and tend to use heavyweight technology for highly virtual and complex projects?
- Is there a significant overhead and learning curve associated with use of complex technology and how does it affect individual success criteria like cost, timeliness and quality?
- Are there differences in the challenges faced and strategies used by dispersed teams based on other parameters, e.g.,
  - the type of work they do (for example, software product development versus maintenance and support)
  - organizational affiliations (for example client-service provider relationship versus multi-location in-house IT organization)
  - Degree of spatial, cultural or temporal separation (for example, truly global teams versus teams that are located, say, just in different cities)

Global software development has rapidly become ubiquitous as it is an inevitable answer to many pressing demands of the market like cost reduction, skill availability, speed-to-market and round-the-clock operations. Developing countries like India, China, Philippines and others have become software factories of the world, much like the Pacific Rim countries became the electronics factories a few decades ago. It will be interesting to study if there is a more fundamental and more significant qualitative shift accompanying this quantitative shift. Is the refinement of the global development model a precursor to more and more value-added software development work like product conceptualization and design being performed globally? What new challenges does that

bring? This study has provided some understanding of these issues and developed a foundation for practice and future research in this challenging and important area.

## References

- Anthes, Gary. H. and Vijayan, Jaikumar., "Lessons from India Inc.", *Computerworld*, 35(14), 2001.
- Attaran, Moshen., Attaran, Sharmin., "The coming age of virtual teaming: Guidelines for managers", *International Journal of Management*, 20(2), 2003, pp.171-178.
- Battin, R.D., Crocker, R., Kreidler, J. and Subramanian, K., "Leveraging Resources in Global Software Development Software", *IEEE Software*, 18(2), 2001, pp 70-77.
- Beise, Catherine M., "Employees and impact on work: IT project management and virtual teams", *Proceedings of the 2004 SIGMIS Conference on Computer Personnel Research: Careers, Culture, and Ethics in a Networked Environment*, 2004.
- Bell B.S. and Kozlowski S.W.J., "A typology of virtual teams: Implications for effective leader ship", *Group and Organization Management*, 27(1), 2002, pp. 14-49.
- Beranek, P.M., "The impacts of Relational and Trust Development Training on Virtual Teams, an Exploratory Investigation", *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, 2000.
- Blackburn, Joseph D. and Scudder, Gary D., "Improving Speed and Productivity of Software Development: A global survey of Software Developers", *IEEE transactions on Software Engineering*, 22(12),1996, pp. 875-885.
- Burtha, Mike., "Learning the secrets of long –distance Leadership", *KM Review*, 7(1), 2004, pp. 24-29.
- Carmel, E. and Aggarwal, R., "Tactical Approaches for Alleviating Distance in Global Software Development", *IEEE Software*, March/April 2001, pp. 22-29.
- Damian, Daniela., Lanubile, Filippo. and Oppenheimer, Heather L., "Addressing the challenges of software industry globalization: The workshop on global software development", *Proceedings of the 25th International Conference on Software Engineering (ICSE'03)*, 2003.
- Damian, D., "An empirical study of requirements engineering in distributed software projects :Is distance negotiation more effective?", *Software Engineering Conference*,2001, pp149-152.
- Damian, Daniela E.and Zowghi, Didar , " An insight into the interplay between culture, conflict and distance in globally distributed requirements negotiations", *Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS, 2003)*
- Damian, D. and Eberlein, Armin, "An empirical study of groupware support for requirements negotiations in distributed software development",  
<http://www.enel.ucalgary.ca/People/eberlein/publications/DamianEberlein.PDF>
- De Souza, Cleidson R.B., Redmiles, David., and Dourish, Paul., ""Breaking The Code", Moving Between Private And Public Work In Collaborative Software Development", *Proceedings of the 2003 International ACM SIGGROUP conference on supporting group work*. Retrieved from ACM digital library on June 26, 2004.
- DeFranco-Tommarello, J. and Deek, F.P., "Collaborative Software Development: A Discussion of Problem Solving Models and Groupware Technologies", *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, 2002.
- Dubé, L. and Paré, G., "The Multi-faceted Nature of Virtual Teams. In Pauleen, D.J. (ed). *Virtual Teams: Projects, protocols, and Processes*. Idea Group Publishing, 2004.

- Dube', Line. and Pare' Guy., "Global applications of Collaborative Technology :Global virtual teams", *Communications of the ACM*, 44(12), 2001.
- Easley, Robert F., Devaraj, Sarv. and Crant, J. Michael. "Relating collaborative technology use to teamwork quality and performance: An empirical analysis", *Journal of Management Information Systems*, 19(4), 2003, pp 247-268.
- Ebert, C. and De Neve, P., "Surviving Global Software Development", *IEEE Software*, 18(2), 2001, pp. 62-69.
- Fontana, John., "Groove release improves collaboration software", *Network World Framingham*, 21(29), 2004, pp.21-22.
- Giffin, Steve D., "A Taxonomy of Internet Applications for Project Management Communications", *Project Management Journal*, 33(4), 2002, pp. 39-48.
- Gillard, Sharlett ., "IT Project Management: A conceptual View", *Journal of American Academy of Business*, 5(1/2), 2004, pp. 381-386.
- Gopal, Anandasivam., Mukhopadhyay, Tridas. and Krishnan, Mayuram S., " The role of software processes and communication in offshore software development", *Communications of the ACM*, (45) 4, 2000, pp.193 -200.
- Greif, Irene (2004), "The socialization of collaboration", URL: <http://www.computerworld.com/softwaretopics/software/apps/story/0,10801,95472,00.html>, Last accessed: October1, 2004.
- Griffith, Teri L., Sawyer, John E., Neale. and Margaret, A., "Virtualness and Knowledge in teams : Managing the love triangle of organizations, individuals and information technology", *MIS Quarterly*, 27(2), 2003, pp. 265-287.
- Grinter, Rebecca E., Herbsleb, James D., Perry, Dawayne E., "The geography of d coordination: dealing with distance In R&D work", *Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work*, 1999.
- Gutwin, Carl., Penner, Reagan. and Schneider, Kevin., "Group awareness in distributed software development", *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*, 2004.
- Handel, Mark. and Herbsleb, James D., "IM everywhere: What is chat doing in the workplace?", *Proceedings of the 2002 Conference on Computer Supported Cooperative Work*, 2002.
- Herbsleb, James ., Zubrow, David., Goldenson, Dennis., Hayes, Will. and Paulk, Mark., "Software Quality and the Capability Maturity Model", *Communications of the ACM*, 40(6), 1997, pp.30-40.
- Herbsleb, James D., Mockus, Audris., Finholt, Thomas A. and Grinter, Rebecca E., " Distance, dependencies and delay in global collaboration", *Proceedings of the 2000 ACM conference on Computer Supported Cooperative Work*, 2000.
- Kay, Russell., *Collaboration software vendors*, July 2004, <[http://www.computerworld.com/softwaretopics/software/groupware/story/0,10801,94542,00.html?from=story\\_package](http://www.computerworld.com/softwaretopics/software/groupware/story/0,10801,94542,00.html?from=story_package)>
- Kayworth, Timothy R., Leidner, Dorothy E., "Leadership Effectiveness in Global Virtual Teams", *Journal of Management Information Systems*, 18(3), Winter 2001/2002, pp. 7-40.
- Keyzerman, Y., "Trust in virtual teams", *Proceedings of the Professional Communication Conference*, IEEE International, 2003, pp. 391 - 400.
- Khazanchi, D. and Zigurs, I. *Patterns of Effective Management of Virtual Projects: An Exploratory Study*. Newtown Square, PA: Project Management Institute, 2005.

- Lanubile, F., Damian, D. and Oppenheimer, Heather L., "Global Software Development: Technical, Organizational and Social Challenges", *ACM Digital Library*, 28(6), 2003.
- Majchrzak, A., Rice, R.E., Malhotra, A., and King, N., "Technology adaptation: The case of a computer supported inter-organizational virtual team", *MIS Quarterly*, 24(4), 2000, pp.569-600.
- Massey, A.P., Montoya-Weiss, M.M. and Hung, Y., "Because Time Matters: Temporal Coordination in Global Virtual Project Teams", *Journal of Management Information Systems*, 19(4), 2003, pp. 129-155.
- Massey, Anne P., Hung, Yu-Ting Caisy., Montoya-Weiss, Mitzi. And Ramesh, V., "When culture and style aren't about clothes: perceptions of task-technology "fit" in global virtual teams", *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*, 2001.
- Mockus, A. and Herbsleb, J., "Challenges of global software development"" Software Metrics Symposium, proceedings, seventh international, 2001, pp 182-184
- Mcgrath, J.E. and Hollingshead, A. B., *Groups interacting with technology: Ideas, Evidence, Issues and an Agenda*. Sage Publications, Thousand Oaks, CA, 1994.
- Melymuka, Kathleen., "Meeting of the Minds: Technology for business 'Swarming'", URL: <http://www.computerworld.com/managementtopics/management/story/0,10801,83401,00.html>, Last accessed: October 24, 2004.
- Carlson, J.R. and George, J. F., "Media appropriateness in the conduct and discovery of deceptive communication : The relative influence of richness and synchronicity", *Group Decision and Negotiation*, 13(2), 2004, pp 191 – 210.
- Pare`, Guy. And Dube`, Line., "Virtual teams: An exploratory study of key challenges and strategies", *Proceedings of the 20<sup>th</sup> International conference on Information Systems*, 1999.
- Pauleen, D.J. and Yoong, P., "Relationship building and the use of ICT in boundary-crossing virtual teams: A facilitator's perspective", *Journal of Information Technology*, 16, 2001, pp. 205-220.
- Phillips, Dwayne., "Project Management: Filling In The Gaps", *IEEE Software*, 13(4), 1996, pp. 17-18.
- Powell, A., Piccoli, G. and Ives, B., "Virtual Teams: A Review of Current Literature and Directions for Future Research", *The Data Base for Advances in Information Systems*, 35(1), 2004, pp. 6-36.
- "Quick Study: Collaboration Software", URL: <http://www.computerworld.com/softwaretopics/software/groupware/story/0,10801,94547,00.html>, Last accessed: October 1, 2004.
- Ramesh, V., Dennis, A. R., "The Object-Oriented Team: Lesson for Virtual Teams for Global Software Development", *Proceedings of the 35<sup>th</sup> Annual Hawaii International Conference*, 2002.
- Robb, Drew "Collaboration gets it together", 2002, URL: <http://www.computerworld.com/softwaretopics/software/groupware/story/0,10801,76490,00.html>, Last accessed: October 1, 2004.
- Saeki, M., "Communication, Collaboration and Cooperation in Software Development – How should we support group work in Software Development?", *Software Engineering Conference, Asia Pacific*, 1995.
- Steinfeld, Charles. Jang, Chyng-Yang., Plaff, Ben., "Supporting virtual team collaboration: the team scope system", *Proceedings of the International ACM SIGGROUP Conference on Supporting groupwork*, 1999 .
- Wilson, Shauna., " Forming Virtual teams", *Asq's ... Annual Quality Congress Proceedings*, Milwaukee, 57, 2003, pp. 489-499.

- Zanoni, Roberto. and Audy, Jorge Luis Nicolas., "Project management model for a physically distributed software development environment", *Proceedings of the 36<sup>th</sup> Annual Hawaii International Conference on system sciences*, 2003, pp. 249-256.
- Zigurs, I. and Buckland, B., "A theory of Task/Technology Fit and Group Support System Effectiveness", *MIS Quarterly*, 22(3), 1998, pp. 313-334.



## Appendix A: Collaboration Software Products

The table below summarizes some of the software products available in the market currently, The products described represent examples of tools that can support different aspects of the systems development process, with a primary focus on collaboration.

Company	Product	Description
Adobe Systems Inc <a href="http://www.adobe.com">www.adobe.com</a>	Document Services for collaboration	Streamlines document reviews via e-mail or web browser Helps protect sensitive business documents Creates searchable digital archives Enables structured and unstructured processes
Advance Reality Inc. <a href="http://www.adavncereality.com">www.adavncereality.com</a>	Presence-AR	Adds synchronous collaboration capabilities to existing and new software applications Allows users to collaborate on the same data using different applications Enables collaboration across firewalls, LAN's and dial-up connections Provides secure collaboration through support of encryption, authentication and access control systems. Similar to groove. The difference is that multiple users can be working on the same document in real time. With groove, still one person needs to be ion the control.
Axista Inc. <a href="http://www.axista.com">www.axista.com</a>	Xcolla	a Web – based project management tool Offers web – based access to real-time project data such as project deliverables, task monitors, project templates, meeting, events and documents. Access to project data from anywhere in the world.
Centra Software Inc. <a href="http://www.centra.com">www.centra.com</a>	Centra 7	Empowers effective change management with a single platform for communication and training Providing training to users Maintain ongoing communication with stakeholders Provides real time communication, learning and collaboration over the web
Citadon Inc. <a href="http://www.citadon.com">www.citadon.com</a>	Citadon collaboration software	Business process automation within and between companies Secure document management Enterprise collaboration and regulatory compliance Project risk mitigation and corporate governance monitoring Communication facilitation of geographically dispersed teams
CollabNet Inc. <a href="http://www.collab.net">www.collab.net</a>	COLLABNET Enterprise edition	Specifically targeted for global software development Provides tools to support multiple software development locations Provides 24x7 development and support
Colligo Networks Inc. <a href="http://www.colligo.com">www.colligo.com</a>	Colligo workgroup edition	Provides instant wireless networking anywhere – 1-to-1 or many to many Secure and private : built in authentication and 168 bit encryption

Company	Product	Description
		<p>Communicate and collaborate in real time</p> <p>Share files, folders, printers and internet connections</p> <p>Application integration: Outlook, netmeeting, lotus notes</p>
<p>Compoze Software Inc. www.compoze.com</p>	<p>Compoze portlets</p>	<p>compose portlets* add collaboration to portals, driving adoptions with functionality used everyday- mail, calendar, contacts and tasks stored in Microsoft Exchange and lotus domino.</p> <p><i>*Portlets are applications that are viewed inside a portal framework from a web browser.</i></p> <p>Portlets can be quickly installed in a portal and cover a wide range of functions like providing news and searching content.</p>
<p>Groove Networks Inc. Groove Workspace</p>	<p><a href="http://groove.net">groove.net</a></p>	<p>Virtual office allows teams of people to work together over a network as if they were in the same physical location</p> <p>Everyone has same set of information</p> <p>Aware of each other through electronic peripheral vision</p> <p>File sharing</p> <p>Management of formal and informal projects to large scale business processes</p>
<p>IBM www.lotus.com</p>	<p>Lotus Domino Express</p>	<p>Lotus notes and domino based products are used to build messaging systems and core business applications where people need to interact- like discussion databases, helpdesk, project tracking or CRM.</p> <p>Lotus Workplace integrated collaborative products connect people with business processes using a single open platform. Users can access to collaborative tools such as messaging, e-meetings and calendaring and scheduling in the context of work they are engaged in.</p>
<p>Kubi software Inc. Kubisoftware.com</p>	<p>Kubi client, Kubi Services</p>	<p>Collaborative-e-mail software, provides n alternative to traditional project management tools and approaches that rely on inefficient Email processes</p> <p>Provides teams with a virtual workspace that is accessible 24*7and allows participants to work with top level view of all projects as well as in the context of a given project.</p> <p>Users have quick access to most accurate, up to date version of most critical project documents, schedules, outstanding tasks, and brainstorming sessions.</p> <p>A central repository frees users from unstructured E-mail interactions, thus streamlining business processes and making it easier to compete projects on time.</p>
<p>Microsoft Corp. www.microsoft.com</p>	<p>Windows SharePoint Services</p>	<p>Helps organizations increase individual and team productivity by enabling them to create website s for information sharing and document collaboration.</p> <p>Provides document libraries</p> <p>Meeting workplace sites</p> <p>Lists</p> <p>Document workplace sites</p> <p>Surveys</p> <p>Templates</p> <p>Threaded view discussion boards</p>

Company	Product	Description
Oracle Corp. <a href="http://otn.oracle.com">http://otn.oracle.com</a>	Oracle Collaboration Suite	Integrates messaging, calendaring, file sharing, real-time communications, wireless access, calendar and time management and voicemail and fax services.
Vignette Corp <a href="http://www.vignette.com">www.vignette.com</a>	V7 Collaboration software	<p>Support sophisticated online and offline communications among co-workers and partners and customers.</p> <p>Business workplaces provide Web-based shared workplaces.</p> <p>Strategic account management enables information and knowledge sharing among colleagues and teams members</p> <p>Project Delivery enables program and project managers to streamline the work and management of teams over widely dispersed geographies.</p> <p>Viagnette Dialog delivers highly personalized content to individual recipients through on-line and off-line touch points.</p>

## Appendix B: Cover Letter for IRB

Dear Participant,

I am conducting a research study on issues related to global software development as part of my thesis as a graduate student at University of Nebraska, - at Omaha. I am asking for your help in this study by participating in an interview. Your participation will take approximately 45-60 minutes.

All responses will be confidential. Most results will be reported at an aggregate level. In all cases, your identity will be made anonymous in any reporting of results. You are free to withdraw your consent to participate and may discontinue your participation in the study at any time without any consequence.

There are no anticipated risks, compensation or other direct benefits to you as a participant in this study. However, there will be indirect benefits. In particular, we will share results with you and other people who could benefit from them in the improvement of global software development.

If you have any questions about this research project, please contact us. Questions or concerns about research participants' rights may be directed to the Institutional Review Board, 402 - 559 - 6463.

The information from this study may be published or presented at meetings, but your identity will not be revealed.

Thanks for your participation in this study.

Sincerely,

Mudita Agarwal (mudita\_\_gupta@yahoo.com)

Dr. Ilze Zigurs, PKI 284 E, University of Nebraska at Omaha, 402-554-3182

250-04-EX

## Appendix C: Questionnaire

Source: Khazanchi, D. and Zigurs, I. *Patterns of Effective Management of Virtual Projects: An Exploratory Study*. Newtown Square, PA: Project Management Institute, 2005.

Please answer all questions to the best of your knowledge. The first question asks you to briefly describe a virtual project that you worked on within the last year, and the remaining questions are about that specific project. For open-ended questions, just type your answer in the blank space after the question. For the rest of the questions, just mark your choice with an "X".

1. Briefly describe the purpose of the virtual project in which you participated during the last twelve months. This project will be the basis for the ideas that you enter in the next agenda item. The rest of the questionnaire asks more detailed questions about this project.
  
2. What was your role in the project?
  - Project Manager
  - Developer/Programmer/Software Engineer
  - Business Analyst
  - Domain Expert
  - Business Manager
  - Other, please specify \_\_\_\_\_
  
3. What was the size of your project team?
  - Small (up to 5 persons)
  - Medium (6 to 15 persons)
  - Large (greater than 15 persons)
  
4. What was the planned schedule for the project?
  - less than 6 months
  - 7 to 12 months
  - Greater than 12 months
  
5. What was the approximate budget for the project in US dollars? \_\_\_\_\_
  
6. Overall, the project was completed as scheduled.
  - Strongly Agree
  - Agree
  - Neutral
  - Disagree
  - Strongly Disagree
  
7. Overall, the project was completed within budget.

- Strongly Agree
  - Agree
  - Neutral
  - Disagree
  - Strongly Disagree
8. Overall, the project met its goal and specified requirements.
- Strongly Agree
  - Agree
  - Neutral
  - Disagree
  - Strongly Disagree
9. Overall, the project was a success.
- Strongly Agree
  - Agree
  - Neutral
  - Disagree
  - Strongly Disagree
10. What was the greatest time difference between you and other project team members?
- Time zone difference was less than 3 hours
  - Time zone difference was between 4 and 9 hours
  - Time zone difference was greater than 10 hours
11. Which phrase best describes the cultural background of the project team members?
- Same culture (homogeneous)
  - Different culture (heterogeneous)
  - Different but team members had similar cultural traits or value systems (hybrid)
12. Which phrase best describes the language differences prevalent between the team members participating in the project?
- Same language (homogeneous)
  - Different languages -- e.g, U.S. and France (heterogeneous)\_
  - Same language, but no shared meaning -- e.g., U.S. and East Indian English (hybrid)
13. Which phrase best describes the proficiency of project team members with virtual team technology?
- Novice (first-time users)
  - User (used technology previously and familiar with main concepts)
  - Expert (completely familiar with the technology)

14. Which statement best describes the number of organizations or firms represented by project team members?
- Team members represented a single organization (intra-organization)
  - Team members represented two different organizations
  - Team members represented more than two different organizations
15. Which phrase best characterizes the overall scope of the project?
- Very large
  - Somewhat large
  - Medium
  - Somewhat small
  - Very small
16. Which phrase best characterizes the overall complexity of the project?
- Extremely complex
  - Somewhat complex
  - Average complexity
  - Somewhat simple
  - Extremely simple
17. Which phrase best characterizes the programmatic risk of the project (e.g., schedule, cost, political issues)?
- Very high risk
  - Somewhat risky
  - Average or medium risk
  - Low risk
  - Very low risk
18. Which phrase best characterizes the technical and engineering risk of the project (e.g., requirements, security, performance, safety)?
- Very high risk
  - Somewhat risky
  - Average or medium risk
  - Low risk
  - Very low risk
19. Which phrase best characterizes the quality risk of the project (e.g., implementation, maintenance, software engineering)?
- Very high risk
  - Somewhat risky
  - Average or medium risk
  - Low risk
  - Very low risk

20. Which phrase best characterizes the logistical risk of the project (e.g., making resources available when and where needed)?
- Very high risk
  - Somewhat risky
  - Average or medium risk
  - Low risk
  - Very low risk
21. Which phrase best characterizes the deployment risk of the project (e.g., training, system integration)?
- Very high risk
  - Somewhat risky
  - Average or medium risk
  - Low risk
  - Very low risk
22. Which phrase best characterizes the overall risk of the project?
- Very high risk
  - Somewhat risky
  - Average or medium risk
  - Low risk
  - Very low risk
23. Which phrase best characterizes the availability of historical knowledge needed to conduct the project's activities?
- Knowledge was explicit
  - Knowledge was implicit
  - Neither of the above, please specify \_\_\_\_\_
24. Which phrase best characterizes the level of innovation inherent in the project?
- Extremely innovative project (brings with it radical change)
  - Somewhat innovative
  - A mix of innovation and traditional (brings with it incremental change)
  - Somewhat traditional
  - Extremely traditional project (little or no change)
25. What was the gender composition of the project team?
- Female-dominated (more than 75% members are females)
  - Male-dominated (more than 75% members are males)
  - Mixed
26. Which phrase best describes the degree of resources available for the project?
- Resources were redundant at each site
  - Resources were complimentary at each site
  - Other, please specify \_\_\_\_\_
27. Which phrase best describes the personality of a majority of the project team members?



- Extremely Homogeneous
- A mixture of personality groups
- Extremely Heterogeneous

28. What was the dominant managerial challenge on this project, that is, what was the one major thing that the team had to pay attention to during the project?
29. How often did you personally use video conferencing (room and/or desktop) to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
30. How often did you personally use fax to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
31. How often did you personally use email to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
32. How often did you personally use voice mail to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
33. How often did you personally use the telephone to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
34. How often did you personally use Web-based intranet tools (example: groove.net) to work with team members on the project?

- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
35. How often did you personally use conference calling to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
36. How often did you personally use face-to-face meetings to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
37. How often did you personally use an electronic meeting system (e.g., WebIQ, GroupSystems, Facilitate.com) to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
38. How often did you personally use instant messaging to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
39. How often did you personally use simultaneous document editing to work with team members on the project?
- Never
  - Seldom
  - Moderately often
  - Frequently
  - Almost always
40. How often did you personally use group calendaring to work with team members on the project?

- Never
- Seldom
- Moderately often
- Frequently
- Almost always

41. How often did you personally use distributed project management tools to work with team members on the project?

- Never
- Seldom
- Moderately often
- Frequently
- Almost always

42. How often did you personally use a workflow system to work with team members on the project?

- Never
- Seldom
- Moderately often
- Frequently
- Almost always

43. How often did you personally use a shared whiteboard to work with team members on the project?

- Never
- Seldom
- Moderately often
- Frequently
- Almost always

44. How often did you personally use any other technologies not mentioned in the above questions to work with team members on the project?

- Never
- Seldom
- Moderately often
- Frequently
- Almost always

**THANK YOU FOR YOUR TIME!**

## **Appendix D: Interview Questions**

1. What specific problems have you experienced in your project with respect to communication among the members of the virtual team?
2. What specific problems have you experienced in your project with respect to coordination during the virtual project?
3. What specific problems have you experienced with respect to control of the virtual project?
4. What specific methodology practices have helped to make the project a success?
5. What specific team practices have helped to make the project a success?
6. What specific technologies and ways that you have used those technologies have helped to make the project a success?
7. What practices used by you were not helpful in your project?

## Appendix E: Responses of the interviewees

ID #	Question	E mail response
1	Communication issues	Time difference that introduces one days delay in the critical tasks.
	Coordination issues	<ul style="list-style-type: none"> <li>We share the same code repository. At times both the teams work on the same piece of code, work on interdependent code. This leads to rework and schedule delays.</li> <li>Server side API's are developed by the client and presentation tier is developed by us. These are developed in parallel, leading to integration issues.</li> </ul>
	Control issues	None
	Methodology practices	<ul style="list-style-type: none"> <li>Well laid out processes</li> <li>Continuous improvement by defect tracking and prevention</li> <li>Good risk management</li> <li>Rotation of tasks and responsibilities to prevent continuous shift-work</li> </ul>
	Team Practices	<ul style="list-style-type: none"> <li>All the teams do a morning meet.</li> <li>Regular project status meets</li> <li>On-site visits</li> </ul>
	Technology	<ul style="list-style-type: none"> <li>At the kickoff of each new release, the client gives technical presentations to the team remotely.</li> <li>Using VPN we are connected to the clients network.</li> <li>Repositories are synchronized automatically.</li> </ul>
	Not helpful practices	No response
2	Communication related issue	<ul style="list-style-type: none"> <li>Accent</li> <li>Time zones</li> <li>*Style of communication</li> <li>*Cultural context</li> </ul> <p><u>Additional Clarification</u>            *. Style of communication and cultural context            What were the specific issues/ some examples?            English being the second language for most team members, was not their natural language of thought. This reflected in the mails being structured rather incoherently at times and required substantial re write effort to bring the points out more clearly. The style also tends to swing towards too casual when attempt is made to be natural in _expression . Example is usage of terms like "apprx" for "approximately" and "wd" for "would".</p> <p>During telephonic conferences, the members tend to get nervous, at least during the early phase of their careers. This would affect the customer's confidence in ability to deliver. The usage of colloquialism by customers sometimes confuses the team members at offshore. Terms like "dangling in the air", "caught in the head lights" and "stepping up to the plate" are not very easy to comprehend contextually for many Indians. Sometimes, the remarks made in the lighter tone remain unappreciated because of the lack of awareness of the cultural background.</p>
	Coordination related issues	<ul style="list-style-type: none"> <li>Sharing a uniform understanding of scope</li> <li>Getting resources in time</li> </ul> <p><u>Additional clarification</u>            . Do you think scope control is more of a problem in virtual projects? Can you please elaborate?            Scope means the extent and content of the work package that has to be delivered. Often there is a mismatch of thoughts in customer's mind about what we are supposed to, or not supposed to, implement as part of the work. Unverified assumptions lie at the root of this problem. Organizations use many methods to arrive at a common understanding of EXACTLY what is to be done under a given contract, with their customers. E.g., signoff on the basis of documented description of work to be done, showing a prototype or using some external reference with qualifications (all features provided by MS Excel, except for copy-paste feature) etc.</p>
	Control related issues	<ul style="list-style-type: none"> <li>Resource availability</li> </ul>

		<ul style="list-style-type: none"> <li>• Scope control</li> <li>• Response to queries in time</li> </ul>
	Methodology practices	<ul style="list-style-type: none"> <li>• Scope signoff at project start</li> <li>• Periodic verification by making interim deliveries</li> </ul>
	Team practices	<ul style="list-style-type: none"> <li>• Joint reviews of work products by onsite and offsite teams</li> <li>• Regular project status review</li> </ul>
	Technology use	We used remote desktop access to hold application walkthroughs. These helped in rapid completion of knowledge transfer
	Not helpful practices	Self study solution for some of the project training needs
3	Communication related issues	Sometimes it becomes really difficult to understand the issues/comments. For this we have to do lot of mails exchange or telephonic conversation.
	Coordination related issues	<ul style="list-style-type: none"> <li>- It is difficult to convince (for reviews, comments, technical issues) a team member with whom you are not discussing face to face.</li> </ul> <p>Document making becomes an overhead, as we have to explain each bit of information.</p> <p><u>Additional Response</u></p> <ul style="list-style-type: none"> <li>- * Respondent felt that in a virtual project, signing off is a major issue. As you are not discussing face to face, each piece of information has to be written in detail so that there is no communication gap and no information is left out. It would be definitely less time consuming if you are sitting in front of the customer as you can discuss it right then and there and make a small one page note and get it signed as both the parties understand what do they mean.</li> </ul>
	Control related issues	<p>It always become difficult to authenticate the time spent on certain task by virtual team</p> <p><u>Additional clarification</u></p> <p>Since there is no person monitoring the team over there, it really becomes an issue for the manager. Attimes this problem is more with team members who are new to the company and are sent directly to the client site.</p>
	Methodological practices	<ul style="list-style-type: none"> <li>- Having planned Rampup time for each Team member</li> <li>- Having common repository for code and tracking documents</li> <li>- Parallel Integration testing with CUT (with some delay in CUT and IT start)</li> <li>- Interim Deliveries to Client</li> <li>- Working smartly with Trainees so that they will complete certain % of project work in their induction period</li> </ul>
	Team Practices	<ul style="list-style-type: none"> <li>- Always keep team motivated</li> <li>- Enabling them for responsibility</li> <li>- Giving them chance to grow professionally by executing more responsibilities then the Role demand</li> </ul>
	Technology use	<ul style="list-style-type: none"> <li>- Close Monitoring of the Task in terms of efforts and schedule help to analyze the risk</li> </ul> <p>Dividing the total individual CUT task into multiple sub tasks and tracking based on the completion of each subtask.</p>
	Practices that were not helpful	Not able to recall any practice which followed but not successful. We followed the practice of revising the approach if it doesn't work
ID# 4	Communication	We were a team of 4, so the communication wasn't much of a problem.
	Coordination	Same as above
	Control	Same as above
	Methodological Practices	Good Team Dynamics and Etiquettes Excellent Resource Planning
	Team Practices	Excellent Team Coordination, Knowledge Sharing among teams
	Technology use	J2EE Design Patterns, Lucene, Oracle, Hibemate, EJB et al
	Practices not useful	Not of any I can think of.
ID 5	Communication	Differences in opinion and lack of motivation were some of the issues.
	Coordination	Telephone lines not good.
	Control	The problems stated above. Plus the lack of requisite skillsets in the team members.

<p>Methodological Practices</p>	<ul style="list-style-type: none"> <li>- Good use case and design templates</li> <li>- Frequent checkpoints with the client.</li> <li>- *Innovative tracking and scheduling methods.             <ol style="list-style-type: none"> <li>1. Frequent Team trainings. <b>Steps of Doneness:</b> We have introduced this concept to provide the developer with a checklist of what steps need to be followed before they can say that a work item has been completed. These steps can be changed to suit a project. The Steps of Doneness that we have proposed for our project are presented below:                 <table border="1" data-bbox="553 373 1495 850"> <thead> <tr> <th colspan="2">Steps of Doneness</th> </tr> <tr> <th>Sr. No.</th> <th>Steps</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Adhere to Coding Guidelines</td> </tr> <tr> <td>2</td> <td>Study Requirement and Design document and identify the issues</td> </tr> <tr> <td>3</td> <td>Resolve all issues.</td> </tr> <tr> <td>4</td> <td>Make code skeleton i.e. put proper comments</td> </tr> <tr> <td>5</td> <td>Get reviewed</td> </tr> <tr> <td>6</td> <td>Do Coding and Unit Testing</td> </tr> <tr> <td>7</td> <td>Get tagging done for Code Review from SCM Coordinator.</td> </tr> <tr> <td>8</td> <td>Release for Code Review</td> </tr> <tr> <td>9</td> <td>In-corporate Code review comments</td> </tr> <tr> <td>10</td> <td>Update Requirements and Design Document</td> </tr> </tbody> </table> </li> </ol> </li> <li>2. <b>Status Tracking</b> – To reduce the need for status tracking with team members, we have created a spreadsheet listing an owner along with the tasks assigned to him/her. Separate columns are provided for each of the steps presented in point 1 above. Two hard copies of this sheet are pasted on the 2 TL's boards. At the end of a day a developer indicates all the steps that have been completed in a particular day as "done" in his TL's sheet. The soft copy of this sheet is updated on a weekly basis.</li> <li>3. <b>Ease of Earned Value calculation</b> – Each step has been assigned a percentage that indicates the amount of work done for a work item. Using the task sheet mentioned in point 2 above, a lead is able to calculate the earned value on a weekly basis. Again, there is no verbal communication required for status checks.</li> <li>4. <b>Introduction of an "Infra" Track</b> – To meet client's aggressive time-to-market needs, we completed the design for the proposed framework by the end of December and began development of this framework in January parallel to the RS and Design phase. This was done by the "Infra Track" and helped in having a ready base for the feature development that started in Feb end.</li> <li>5. <b>Issues Sheet</b> – The standard "Issues and Input" template is being used diligently in the project to track technical and integration dependencies between modules.</li> <li>6. <b>Frequent client checkpoints in Phase I</b> – Weekly RS and Design checkpoints with the client ensured that the client had seen all artifacts atleast once before the final delivery date. This helped us reduce iterations after the Phase I and begin work on Phase II smoothly.</li> </ul> <p>II. Can you also elaborate on team trainings? What are the key things that you focus on in these trainings with respect to virtual projects?</p> <p>Team trainings can be conducted in the following ways:</p> <ol style="list-style-type: none"> <li>a. We hold formal vendor-led team trainings when necessary. E.g. our team went through training by Mercury representatives for their testing tool called QuickTest Pro.</li> <li>b. Formal trainings are held for team members newly joining a project to ensure a proper business and technology ramp-up happens.</li> <li>c. Semi-formal and informal trainings are done in the form of "knowledge Transfer" sessions to ensure team members gain context of what is happening on the other tracks of a project.</li> </ol> <p>III. Does CMM play a role in virtual project management?</p> <p>It sure does. It basically lays out a framework for the Project Manager/Lead to follow. The PM/PL knows that though s/he can get creative there are certain minimum processes that are required to ensure the cross-geography communication happens effectively. E.g. we use standard Query sheets to get requirement clarifications, share standard project plans with the clients and report project status following certain standards.</p>	Steps of Doneness		Sr. No.	Steps	1	Adhere to Coding Guidelines	2	Study Requirement and Design document and identify the issues	3	Resolve all issues.	4	Make code skeleton i.e. put proper comments	5	Get reviewed	6	Do Coding and Unit Testing	7	Get tagging done for Code Review from SCM Coordinator.	8	Release for Code Review	9	In-corporate Code review comments	10	Update Requirements and Design Document
Steps of Doneness																									
Sr. No.	Steps																								
1	Adhere to Coding Guidelines																								
2	Study Requirement and Design document and identify the issues																								
3	Resolve all issues.																								
4	Make code skeleton i.e. put proper comments																								
5	Get reviewed																								
6	Do Coding and Unit Testing																								
7	Get tagging done for Code Review from SCM Coordinator.																								
8	Release for Code Review																								
9	In-corporate Code review comments																								
10	Update Requirements and Design Document																								

	Team Practices	- Frequent Team trainings. - Team meets to check project morale
	Use of technology	<i>Constant reviews of work items done by team members who are fairly new to the industry</i>
	Practices that were not useful	Can't think of any.
6	Communication	Dependency, in our case we were dependent on our virtual team member for *requirement gathering. **Motivation, is another area <u>Additional clarification</u>  Respondent didn't know the virtual team member although they were from the same company. The team member on the client site could not clarify things, thus delaying the requirement gathering process. Time difference of 12 hours add further delay thus at each stage requirement gathering takes 3-4 days as generally you get the requirement in two days and then the off-site team studies it and sends it back with queries to the other site related to couple of features. At times, off-site members have to talk to the clients directly  Motivation was intended towards trainees who were not willing to do the documentation required in a virtual project. They prefer to learn technical components
	Coordination	N/A
	Control	N/A
	Methodological Practices	Trust building and knowing the virtual team members in person.... <u>Additional clarification</u> Problems are more if you don't know the virtual team member and you don't have a good rapport with that person. Respondent said that now he has good rapport with that person so those issues are not there.
	Team Practices	Even allocation of responsibilities to create a sense of participation, accountability and joint ownership
	Use of technology	<i>ASP.Net was used</i>
	Practices that were not useful	N/A
7	Communication	In the project most of the communication was via Emails or Chat on MSN. So problems experienced were: <ul style="list-style-type: none"> <li>• Urgent issues cannot be discussed then and there</li> <li>• Keeping track of all mails was bit difficult.</li> <li>• Have to be online during "Off hours" here</li> </ul>
	Coordination	- Prolonged Decision making process - Have to match up all even minor issues - Difficulty/Confusion getting a real grasp on what is expected of our team (deliverables) - Planning team meetings /activities during times that are non-working hours for the part of team members. - For few team members language was the issue
	Control	Meeting schedule and quality expectation during the time when few old team members left the organization and new replacements were being trained. Had to put extra effort, time and energy.
	Methodological Practices	<ul style="list-style-type: none"> <li>• Best Practices</li> <li>• Frequent Knowledge Transfer meetings</li> <li>• Defining Preliminary Schedule</li> <li>• Making realistic Schedule</li> </ul>
	Team Practices	<ul style="list-style-type: none"> <li>• Meeting every morning before actually starting on work for the day.</li> <li>• Demonstration by each team (Development, Testing, Linux, Support, Graphics) from time to time</li> </ul>



		<ul style="list-style-type: none"> <li>Rotation of work among team members to reduce dependencies and build redundancy in case of communication failure</li> </ul>
	Use of technology	<ul style="list-style-type: none"> <li>Using Struts framework for building Java web application</li> <li>Using JUNIT for unit level testing</li> </ul>
	Practices that were not useful	<b>Using excel sheets for reporting bugs internally</b>
ID #	Question	E mail response
8	Communication	As all the team members with whom I had to deal were from the same cultural background, hence the language was never a problem. There was always a friendly environment among the team members. To share some information we used to move from our seat to the others seat, and have face to face talk. We also used instant messaging and emails to share knowledge.
	Coordination	As such coordination was never a problem in our project. Being a small team we all had a good coordination, we all were very clear about our task. We were using Visual Source Safe for the management of project. We had to face some problems regarding consistency throughout the project, eg all same kind of text boxes should be of same size etc. So at times we had to coordinate a lot regarding the consistency issues.
	Control	There was no problem regarding the control of the problem. Every thing was quite smooth, as I told it was a small project.
	Methodological Practices	As the schedules were tight, we executed Integration testing parallel with coding, this helped us a lot in achieving the best results.
	Team Practices	Every team member was always ready to help other. We all worked together to achieve the target. We kept on reviewing the flaws in our previous deliveries and try to resolve in the next ones.
	Use of technology	<b>We used Dot Net and SQL Server for this project.</b> The most important Visual Source safe was the tool through which the same data could be shared by all the members.
	Practices that were not useful	In this project there was a module in which we had to make a tree view structure. Though there is a tool which could have solved our purpose, but rather than buying that tool we ourselves developed the tree structure. And this took a lot of time.
9	Communication	<ul style="list-style-type: none"> <li>- Accent issues.</li> <li>- Not able to connect while conference call/ voice not clear.</li> </ul>
	Coordination	Code updates/merging.
	Control	<ul style="list-style-type: none"> <li>Backup resources.</li> <li>Network issues.</li> </ul>
	Methodological Practices	<ul style="list-style-type: none"> <li>- Automatic builds and notifications.</li> </ul> <p>Estimation schedule. Issues reporting/tracking using bugzilla..</p>
	Team Practices	Team knowledge sharing sessions.
	Use of technology	<ul style="list-style-type: none"> <li>- CVS, Bugzilla, Microsoft Project Plan, ANT, Eclipse</li> <li>- CMM Processes</li> </ul>
	Practices that were not useful	Manual repository upload
10	Communication	None – fortunately, for this project, there were no communication problems. Most likely, that is due to the project team members experience in the onshore-offshore model. We have learnt from previous mistakes – this project went very smooth.
	Coordination	Co-ordination of my project team members was fine – I had issues coordinating our client resources (domain experts and testers). Even though they had their own project manager, sometimes you need to

		constantly stay on top of them.
	Control	Managing client expectations was difficult at times – there were a number of feature requests after delivery that were clearly out of scope (from our perspective) but the client felt they were within scope. The client also posted bugs that were related to their testing environment and not our application.
	Methodological	Our methodology for collecting, analyzing, developing, and maintaining requirements made things clear for our technical designers and developers to code to the specifications – specifically, Low Level Requirements document, UI Mockups, Requirements traceability matrix. Our methodologies for design, development, and testing ensured a quality software product – specifically, Low Level Design document, Unit and Integration Test specification, QA Test specifications. Lastly, our methodologies for managing the project and scope control – change request control, risk mitigation, schedules, etc.
	Team	
	Technology	
	Not useful	I began to document Use Cases, but realized the client was not interested and for the project size, it was not worth it – instead, I presented UI Mockups to the client and our project team members. We have found this method to work well in the offshore model
11	Communication	<p>We are following the process of off-shore development model, where our end customers are located in remote areas, e.g. Canada and USA. We have our on-site team, including sales team, product manager and client representative residing in those remote areas. We are following a process where we are not directly interacting with the end customers, for us only point of contact is our on-site team.</p> <p>By following above mentioned process, our end customers feels like we have a team working 24 hours for them, that includes on-site and off-site team working in different time zone.</p> <p>But we have a disadvantage of having an extra mediator as it increases the response time from the customer, also sometimes miscommunication happens, as we have an extra loop in place. Following are some of the scenarios:</p> <p>Scenario 1: we like to ask a query with our client but sometime our on-site team responds the query, tries to resolve it by them and that response might not be the exact match with the customer.</p> <p>Scenario 2: We ask a query and they forward it the Client then we might got a delay due to this extra loop.</p> <p>Scenario 3: Some time they modify client's response and respond us back with what they understood with that. Sometime it causes issue of miscommunication.</p>
	Coordination	Already mentioned above.
	Control	We don't really face any issue in terms of control as our roles are very clearly defined
	Methodological	We use log sheets to control the communication gaps, where we track all of our queries and respective responses, we normally calls it the "Queries and Assumption Log". It contains all the information required to provide the complete details of any communication with (either client or on-site team, e.g. queries). This information provides us the detail like what was the query, when it was initiated, who initiated it, who responded to it (on-site team or Client), on what date and what was the final action point. This kind of tracking helped us in controlling miscommunication if any arises.
	Team	Already mentioned above the best practice we are following. We also convert all of our discussion whether in MSN chat and Teleconference into "Minutes of Meeting" (MOM) having all of the action items and who will be responsible to complete a particular action point.
	Technology	To control it we are using MSN chat session, Teleconference and E-mails.
	Not useful	None
12	Communication related issues	Time difference that introduces one days delay in the critical tasks.
	Coordination related issues	<ul style="list-style-type: none"> <li>We share the same code repository. At times both the teams work on the same piece of code, work on interdependent code. This leads to rework and schedule delays.</li> <li>Server side API's are developed by the client and presentation tier is developed by us. These are developed in parallel, leading to integration issues.</li> </ul>
	Control related issues	None

	Methodology Practices	<ul style="list-style-type: none"> <li>• Well laid out processes</li> <li>• Continuous improvement by defect tracking and prevention</li> <li>• Good risk management</li> <li>• Rotation of tasks and responsibilities</li> <li>• Grooming the members to take up higher roles</li> </ul>
	Team Practices	<ul style="list-style-type: none"> <li>• All the teams do a morning meet.</li> <li>• Regular project status meets</li> <li>• On-site visits</li> </ul>
	Technology	<ul style="list-style-type: none"> <li>• At the kick of new release, the client gives technical presentations to the team.</li> <li>• We are connected to the client's network over VPN</li> <li>• Repositories are synchronized automatically.</li> </ul>
	Not helpful Practices	No response

## Appendix F: Tabulated Questionnaire Results

Participant Id	1	2	3	4	5	6	7	8	9	10	11	12	MEAN	STD. DEV
#2 Role in the project?	Dev	PM	TL	Dev	TL	Dev	Tester	Dev	Dev	PM/BA	PM	TL		
#3 size of your project team?	5	1	5	3	5	1	3	3	5	1	1	5	3.17	1.80
#4 planned schedule for the project?	5	1	3	1	3	1	1	1	1	1	3	1	1.83	1.34
#6 Overall, the project was completed as scheduled	4	5	4	5	5	5	4	4	4	5	5	4	4.50	0.52
#7 Overall the project was completed within budget		5	4	3	5	5	4	3		5	3		4.11	0.93
#8 Overall, the project met its goal and specific requirements	5	5	5	4	5	5	4	5	4	5	4	4	4.58	0.51
#9 Overall, the project was a success.	5	5	5	4	5	5	4	5	3	5	4	3	4.42	0.79
#10 greatest time difference between team members?	5	5	5	1	5	5	5	5	5	5	5	5	4.67	1.15
#11 cultural background of team members?	3	5	3	1	3	5	5	1	3	5	3	3	3.33	1.44
#12 language differences between team members t?	1	1	1	1	1	1	5	1	1	3	1	1	1.50	1.24
#13 proficiency in virtual team technology?	1	1	3	1	5	3	3	3	3	3	3	3	2.67	1.15
#14 number of organizations or firms represented ?	1	3	3	1	5	1	1	1	3	3	1	3	2.17	1.34
#15 overall scope of the project?	4	4	5	3	4	2	3	2	4	3	3	3	3.33	0.89
#16 overall complexity of the project?	5	3	4	3	3	4	3	4	4	4	3	4	3.67	0.65
#17 programmatic risk of the project	3	4	2	3	3	3	2	4	3	3	2	4	3.00	0.74
#18 technical and engineering risk of the project	3	4	3	2	4	3	3	2	2	3	2	2	2.75	0.75
#19 quality risk	3	4	2	2	4	3	2	2	4	4	3	3	3.00	0.85
#20 ogistical risk	3	3	3	2	4	3	2	3	2	3	4	2	2.83	0.72
#21 deployment risk	3	3	1	2	2	2	3	2	3	2	2	2	2.25	0.62
#22 overall risk	3	4	2	2	4	3	3	3	3	3	2	3	2.92	0.67
#23 availability of historical knowledge		3	1	1	1	5	1	1	5	5	5	5	3.00	2.00
#24 level of innovation	2	3	4	3	4	3	2	2	3	3	3	3	2.92	0.67
#25 gender composition of the project team?	3	3	2	2	2	3	2	3	3	2	2	2	2.42	0.51
#26 degree of resources available for the project?	5	5	3	3	1		5	5	5	5	1	5	3.91	1.64

Participant Id	1	2	3	4	5	6	7	8	9	10	11	12		
#27 personality of a majority of the project team members?	3	3	3	3	3	1	3	1	3	3	3	3	2.67	0.78
#29: use video conferencing	1	1	1	1	1	1	1	1	1	1	1	1	1.00	0.00
#30 use fax	2	1	1	1	1	1	2	1	2	1	2	2	1.42	0.51
#31 use email	5	5	5	4	5	5	5	4	4	5	4	4	4.58	0.51
#32 use voice mail	1	1	2	1	1	1	2	1	1	1	2	1	1.25	0.45
#33 use the telephone	5	5	3	2	4	5	3	2	4	3	4	4	3.67	1.07
#34 use Web-based intranet tools	1	1	3	1	1	1	2	1	4	1	2	4	1.83	1.19
#35 use conference calling	5	4	2	1	3	4	3	4	4	3	2	4	3.25	1.14
#36 use face-to-face meetings	5	1	2	3	4	4	3	5	2	1	1	2	2.75	1.48
#37 use an electronic meeting system	1	1	1	1	1	1	1	1	1	1	1	1	1.00	0.00
#38 use instant messaging	3	2	2	4	3	1	4	3	3	5	4	3	3.08	1.08
#39 use simultaneous document editing	1	1	1	2	5	1	4	2	3	1	1	4	2.17	1.47
#40 use group calendaring	1	1	1	3	1	1	3	2	4	1	3	4	2.08	1.24
#41 use distributed project management tools	1	1	1	4	1	1	2	5	1	1	3	1	1.83	1.40
#42 use a workflow system	1	1	1	1	1	1	3	4	2	1	2	2	1.67	0.98
#43 use a shared whiteboard	4	1	1	1	3	4	3	2	5	1	2	5	2.67	1.56
#44 use any other technologies not mentioned	1	2	1	1	1	1	2	2	2	1	1	2	1.42	0.51

Note: All responses are on a 5-point scale, with 1 being lowest and 5 being highest.

## Appendix G: Calculation of Project Characteristics

Note: All responses are on a 5-point scale, with 1 being lowest and 5 being highest.

### Project Scope

Participant ID	Q#4	Q#15	Q#24	Scope Means
1	5	4	2	3.67
2	1	4	3	2.67
3	3	5	4	4.00
4	1	3	3	2.33
5	3	4	4	3.67
6	1	2	3	2.00
7	1	3	2	2.00
8	1	2	2	1.67
9	1	4	3	2.67
10	1	3	3	2.33
11	3	3	3	3.00
12	1	3	3	2.33

Question #4 – Duration  
 #15 – Scope  
 #24 – Innovation

### Project Complexity

Participant ID	#3	#11	#12	#13	#16	#23	#26	#27	Complexity Means
1	5	3	1	1	5		5	3	3.29
2	1	5	1	1	3	3	5	3	2.75
3	5	3	1	3	4	1	3	3	2.88
4	3	1	1	1	3	1	3	3	2.00
5	5	3	1	5	3	1	1	3	2.75
6	1	5	1	3	4	5		1	2.86
7	3	5	5	3	3	1	5	3	3.50
8	3	1	1	3	4	1	5	1	2.38
9	5	3	1	3	4	5	5	3	3.63
10	1	5	3	3	4	5	5	3	3.63
11	1	3	1	3	3	5	1	3	2.50
12	5	3	1	3	4	5	5	3	3.63

Question #3 – Team size  
 #11 – Cultural homogeneity  
 #12 – Language differences  
 #13 – Proficiency in virtual team technology  
 #16 – Overall complexity  
 #23 – Availability of historical knowledge  
 #26 – Resource availability  
 #27 – Disparities in individual personalities

## ***Project Risk***

<b>Participant ID</b>	<b>#17</b>	<b>#18</b>	<b>#19</b>	<b>#20</b>	<b>#21</b>	<b>#22</b>	<b>Risk Means</b>
1	3	3	3	3	3	3	3.00
2	4	4	4	3	3	4	3.67
3	2	3	2	3	1	2	2.17
4	3	2	2	2	2	2	2.17
5	3	4	4	4	2	4	3.50
6	3	3	3	3	2	3	2.83
7	2	3	2	2	3	3	2.50
8	4	2	2	3	2	3	2.67
9	3	2	4	2	3	3	2.83
10	3	3	4	3	2	3	3.00
11	2	2	3	4	2	2	2.50
12	4	2	3	2	2	3	2.67

Question #17 – Programmatic risk  
 #18 – Technical/Engineering Risk  
 #19 – Quality Risk  
 #20 – Logistical Risk  
 #21 – Deployment Risk  
 #22 – Overall Risk

## Technology Use

Participant ID	Vid.Conf.	Fax	Email	VoiceMail	Phone	Intranet	Conf.Call	Face2Face	Elec.Mtg.	Inst.Msg.	Simult. Doc.Editing	Group.Calender	Distr. Proj.Mgmt	Workflow Systems	Whiteboard	Other	Technology Use Means
1	1	2	5	1	5	1	5	5	1	3	1	1	1	1	4	1	2.38
2	1	1	5	1	5	1	4	1	1	2	1	1	1	1	1	2	1.81
3	1	1	5	2	3	3	2	2	1	2	1	1	1	1	1	1	1.75
4	1	1	4	1	2	1	1	3	1	4	2	3	4	1	1	1	1.94
5	1	1	5	1	4	1	3	4	1	3	5	1	1	1	3	1	2.25
6	1	1	5	1	5	1	4	4	1	1	1	1	1	1	4	1	2.06
7	1	2	5	2	3	2	3	3	1	4	4	3	2	3	3	2	2.69
8	1	1	4	1	2	1	4	5	1	3	2	2	5	4	2	2	2.50
9	1	2	4	1	4	4	4	2	1	3	3	4	1	2	5	2	2.69
10	1	1	5	1	3	1	3	1	1	5	1	1	1	1	1	1	1.75
11	1	2	4	2	4	2	2	1	1	4	1	3	3	2	2	1	2.19
12	1	2	4	1	4	4	4	2	1	3	4	4	1	2	5	2	2.75

1.00	1.42	4.58	1.25	3.67	1.83	3.25	2.75	1.00	3.08	2.17	2.08	1.83	1.67	2.67	1.42	2.23
0.00	0.51	0.51	0.45	1.07	1.19	1.14	1.48	0.00	1.08	1.47	1.24	1.40	0.98	1.56	0.51	0.37

## Degree of Success

Participant ID	#6	#7	#8	#9	Success Means
1	4		5	5	4.67
2	5	5	5	5	5.00
3	4	4	5	5	4.50
4	5	3	4	4	4.00
5	5	5	5	5	5.00
6	5	5	5	5	5.00
7	4	4	4	4	4.00
8	4	3	5	5	4.25
9	4		4	3	3.67
10	5	5	5	5	5.00
11	5	3	4	4	4.00
12	4		4	3	3.67

- Question #6 – Schedule  
 #7 – Budget  
 #8 – Goals and Requirements  
 #9 – Overall Success