Massachusetts Institute of Technology
Engineering Systems Division

Working Paper Series

ESD-WP-2007-16

THE USE OF INFORMATION SYSTEMS IN COLLOCATED AND DISTRIBUTED TEAMS: A TEST OF THE 24-HOUR KNOWLEDGE FACTORY

Satwik Seshasai[1], Alan J. Malter[2], and Amar Gupta[3]

[1]Doctoral Candidate, Engineering Systems, MIT
Cambridge, MA 02138 USA
satwik@us.ibm.com

[2]Assistant Professor, Department of Marketing
Eller College of Management, University of Arizona
Tucson, AZ 85718 USA
amalter@eller.arizona.edu

[3]Thomas R. Brown Professor of Management and Technology
Eller College of Management, University of Arizona
Tucson, AZ 85721 USA
agupta@arizona.edu

March 2007

The Use of Information Systems in Collocated and Distributed Teams:

A Test of the 24-Hour Knowledge Factory

Satwik Seshasai
Doctoral Candidate, Engineering Systems, MIT
Advisory Software Engineer and Development Manager, Team Collaboration
International Business Machines (IBM), Corp
Suite 2313
Five Technology Park Drive
Westford, MA 01886 USA
Tel. 1-978-399-6241
Email: satwik@us.ibm.com

Alan J. Malter
Assistant Professor, Department of Marketing
Eller College of Management, University of Arizona
McClelland Hall, Room 320, P.O. Box 210108
Tucson, AZ 85718 USA
Tel. 1-520-626-7353, Fax. 1-520-621-7483
Email: amalter@eller.arizona.edu

Amar Gupta
Thomas R. Brown Professor of Management and Technology
Eller College of Management, University of Arizona
McClelland Hall, Room 417H, P.O. Box 210108
Tucson, AZ 85721 USA
Tel. 1-520-626-9842, Fax. 1-520-621-8105
Email: agupta@arizona.edu

The Use of Information Systems in Collocated and Distributed Teams:
A Test of the 24-Hour Knowledge Factory

Author Biographies:

Satwik Seshasai is the Development Manager for QuickPlace, IBM's enterprise team collaboration platform, and is currently pursuing a doctoral degree in Engineering Systems at the Massachusetts Institute of Technology. He holds two Masters degrees from MIT - in Computer Science, and Technology and Policy - and has published research in leading journals on technology and managerial aspects of distributed teams and knowledge sharing. Assisting professors Amar Gupta and Lester Thurow, he helped found the nation's first MBA course on Offshore Outsourcing, at MIT. As an Advisory Software Engineer at IBM, he has worked with many Fortune 500 companies to develop and deploy software for team collaboration across many countries and individuals.

Alan Malter is Assistant Professor of Marketing at the University of Arizona. He earned his PhD in 2000 from the University of Wisconsin-Madison. His research examines the changing role of marketing and global strategy; industry clusters and the effects of geographic proximity on knowledge transfer, innovation, and firm performance; and tacit knowledge in managerial and consumer decision-making. His research has appeared in the *Journal of Marketing, MIT Sloan Management Review, Journal of Product Innovation Management, International Journal of Research in Marketing, Organizational Dynamics,* and other journals. He previously worked as a trade analyst and consultant on export development for The World Bank.

Amar Gupta is Tom Brown Endowed Chair of Management and Technology; Professor of Entrepreneurship, Management Information Systems, Management of Organizations, and Computer Science at the University of Arizona since 2004. Earlier, he was with the MIT Sloan School of Management (1979-2004); for half of this 25-year period, he served as the founding Co-Director of the Productivity from Information Technology (PROFIT) initiative. He has published over 100 papers, and serves as Associate Editor of *ACM Transactions on Internet Technology* and the *Information Resources Management Journal*. At the University of Arizona, Professor Gupta is the chief architect of new multi-degree graduate programs that involve concurrent study of management, entrepreneurship, and one specific technical or scientific domain. He has nurtured the development of several key technologies that are in widespread use today, and is currently focusing on the area of the 24-Hour Knowledge Factory.

The Use of Information Systems in Collocated and Distributed Teams:
A Test of the 24-Hour Knowledge Factory

Abstract

Recent academic and policy studies focus on offshoring as a cost-of-labor driven activity that has a direct impact on employment opportunities in the countries involved. This paper broadens this perspective by introducing and evaluating the 24-hour knowledge factory as a model of information systems offshoring that leverages other strategic factors beyond cost savings. A true 24-hour knowledge factory ensures that progress is being made on information systems related tasks at all times of day by utilizing talented information systems professionals around the globe. Many organizations currently implement other variants of offshoring that appear similar but are fundamentally distinct. The typical model is a service provider framework in which an offshore site provides service to the central site, often with two centers and a distinction between a primary center and secondary center. Entire tasks are often outsourced to the lower-cost overseas site and sent back when completed. In contrast, the 24-hour knowledge factory involves continuous and collaborative round-the-clock knowledge production achieved by sequentially and progressively distributing the knowledge creation task around the globe, completing one cycle every 24 hours. Thus, the 24-hour knowledge factory creates a virtual distributed team, in contrast to a team that is collocated in one site, either onshore or offshore. By organizing knowledge tasks in this way, the 24-hour knowledge factory has the potential to work faster, to provide cheaper solutions, and to achieve better overall performance. Previous studies have examined individual teams over time and explored various benefits of distributing work to distant teams, but have not directly compared the effect of collocation versus geographic distribution on the use of information systems and the overall performance over time of two real-world teams working on a similar task in controlled conditions. This paper highlights the concept of the 24-hour knowledge factory and tests the model in a controlled field experiment that directly compares the use of information systems and subsequent performance in collocated and globally distributed software development teams. The central finding is that while collocation versus geographic distribution changes the way teams use information systems and interact at key points during a project, each type of team has the potential to use information systems to leverage its inherent advantages, to overcome disadvantages, and ultimately, to perform equally well. In other words, one organizational structure is not inherently superior nor must structure pre-determine performance. Geographic distance introduces new challenges but these can be overcome – and even leveraged for strategic advantage. In sum, our findings suggest that firms can apply the 24-hour knowledge factory model to transition from a service provider framework in which offshoring is a short-term and unilateral cost-saving tactic to a strategic partnership between centers in which offshoring becomes a core component of a global corporate strategy.

Keywords: 24-hour knowledge factory; information systems; collocated teams; virtual distributed teams; offshoring; outsourcing; innovation; group process.

The Use of Information Systems in Collocated and Distributed Teams: A Test of the 24-Hour Knowledge Factory

INTRODUCTION

The Association of Computing Machinery (ACM) released a report in 2006 that took a comprehensive look at offshoring but proceeded with a base assumption that offshoring is largely about cost savings (Asprey, Mayadas and Vardi 2006). The report does describe the opportunities for all nations to benefit based on economic theories of comparative advantage; however, it does not discuss the opportunity to utilize a globally distributed workforce to transform the dynamic within which offshoring is conducted. A recent editorial in *MIS Quarterly* suggests acceptance of the shift of information systems jobs to other countries and preparation of students for a global job marketplace (Weber 2004). Similar reports from the Institute of Electrical and Electronics Engineers (IEEE) and the National Society of Professional Engineers (NSPE) also describe offshoring in the context of jobs being gained or lost due purely to cost savings between nations (IEEE 2004; White 2004). The NSPE report declares that offshoring should only be done when the talent cannot be found in the United States and suggests this policy direction as the correct means for preserving jobs within the United States. The IEEE's approach is slightly less defensive, as they advocate training programs within the United States to help U.S. workers gain skills to compete with international workers. These reports all present the employment aspect as a competitive zero-sum situation, where work can only be done in one country or the other.

Previous scholarly research in psychology and business strategy reinforces the idea that product development might be done most productively in one location. Studies on inter-personal communication have shown that geographic distance reduces the opportunity for face-to-face

interaction (Conrath 1973), which is considered necessary for transferring tacit knowledge between individuals and organizations (Porter 1998; Sternberg 1991; Tallman et al. 2004). Thus, physical distance is widely considered detrimental to inter-personal and inter-organizational collaboration, leading many firms in the 1980's and 1990's to prefer collocating large cross-functional teams at a single site (Eppinger and Chitkara 2006). However, recent advances in information technology have enabled virtual distributed teams to work effectively without meeting face-to-face (Cummings 2004; Mazneski and Chudoba 2000), suggesting potential for a new model of distributed knowledge production that can leverage geographic distance for strategic advantage.

This paper seeks to further broaden the view of offshoring from simply a cost saving exercise to a truly modern means of utilizing a globally distributed workforce in a manner which has only recently become possible, due to advances in information systems. By employing this methodology both within the development of IS tools to facilitate the model as well as within organizations producing the information systems, it becomes possible to consider offshoring in a broader perspective where the interests of workers in high-income economies are aligned with workers in other countries and customers worldwide. We introduce and test the 24-hour knowledge factory model (Gupta and Seshasai 2004), which advocates continuous work on knowledge-based tasks by individuals located in time zones that allow for 24-hour engagement.

The term "24-hour knowledge factory" connotes a globally distributed work environment in which members of the global team work on a project around the clock, i.e., a project which literally "follows the sun" (Treinen and Miller-Frost 2006; Yap 2005). Each member of the team works the normal workday hours that pertain to his or her time zone. At the end of such a workday, a fellow team member located in a different time zone continues the same task. This

concept flows from the fundamental belief that, in most cases, a person can work most effectively during the normal daytime work period (roughly from 9 am to 5 pm). While one can temporarily work during the night, such a mode of operation is not convenient or optimal over an extended period of time. Further, by having three sets of individuals perform work over a 24 hour period, the objective is to drastically reduce the time needed to develop information systems. In fact, one can distinguish between two kinds of environment that are of direct relevance to the information systems community. One is the design, development, and implementation of information systems in a manner that leverages the new paradigm. The other is the development of new information system approaches that can help to apply the 24-hour knowledge factory paradigm to a broad range of white-collar activities ranging from medical services to logistics planning, and from financial analysis to product design.

When and how should firms employ the 24-hour Knowledge Factory paradigm in the context of their information systems? What are the short-term and long-term implications of this paradigm? In order to answer these critical questions, one needs to look at historical precedents as well as to analyze a number of interrelated technical, strategic, organizational, and economic issues. The notion of the 24-hour knowledge factory can be traced back to the industrial revolution. Since installed manufacturing equipment was scarce and costly, different sets of employees were scheduled to work in successive shifts so that the manufacturing facilities could be used on a round-the-clock basis. The use of the 8-hour shift system evolved over time. Initially, each worker was directed to work 12-16 hours a day so that each machine could be used for an extended period of time. Then, the notion of having two shifts evolved. Based on new legislation on both sides of the Atlantic, the work hours were gradually reduced. The introduction of the shift system yielded benefits in terms of higher productivity of each machine,

reduced production times, and lower prices to customers. However, it also created social and health issues by requiring the person to work in an urban setting, usually away from other members of the family, and also at odd hours and changing work schedules determined by the idiosyncrasies of the manager in charge of assigning workers to different shifts.

The advent of electronic computers and the diminishing costs for telecommunications enabled the development of 24-hour Call Centers. Depending on the time of the call, it is automatically directed to a call center that is active at that time. Using a cluster of three to four call centers located in time zones 6-8 hours apart from the time zone of the neighboring call center, one can ensure that all employees of these geographically distributed call centers are working during daytime in their respective countries. The notion of multiple support centers was subsequently adopted for supporting global communications networks over time. Now it has become feasible to use a geographically distributed workforce of highly trained professionals to complete a knowledge-based production task such as product development in a much shorter timeframe than a similar team based at one location.

By involving specialized microchip design engineers located at multiple places around the world, a semiconductor chip design firm may create virtual 24-hour knowledge factories. Ideally, this structure allows for an efficient design process with faster turnaround time, which is one of the major potential benefits of distributing work across time zones (Gupta and Seshasai 2004; Treinen and Miller-Frost 2006; see also Eppinger and Chitkara 2006; Majchrzak et al. 2000). The 24-hour knowledge factory model is best suited for knowledge tasks that involve a sequence of stages where one stage must be completed before the next stage begins, with the output of one sub-task serving as input for the next. Hence, the ability to complete these stages in faster succession will speed completion of the overall project. Suitable tasks will also be modular, so

that natural breaks that can serve as hand-off points. Personnel must also adapt, and be willing to trust and yield control to fellow team members who continue the same task in subsequent shifts.

The 24-hour knowledge factory model provides the firm with access to high-talent designers who would otherwise have to move to a different country, or to work at odd hours of the night, often referred to as the "graveyard shift." The creation of professional service teams that transcend geographic boundaries and temporal boundaries offers the potential to change the face of many industries. This innovation has the potential to dramatically impact the manner in which companies design, develop, implement, test, and maintain their diverse repertoire of information system assets. In the past, people around the world deemed the time difference between fellow workers to be a major impediment when implementing information systems; they thought the time difference would hinder their ability to perform the work and add significant overheads, time delays, and costs. Now, the perception has switched around; for many projects, the time difference is viewed as a strategic plus, as it enables the creation of the virtual 24-hour knowledge factory highlighted in this paper. However, both views are based on largely untested assumptions regarding the nature of work by collocated and distributed teams and the feasibility of handing off tasks across shifts in the 24-hour knowledge factory model.

Research related to this area has been motivated partly by expert perspectives on the changing nature of organizations and IT. Venkatraman and Subramaniam (2002) have suggested that future research on strategy and competitive advantage should focus on individuals, intellectual capital, and relationships between individuals. The individuals involved are inevitably distributed around the globe and the challenge is to build systems that leverage global talent (Venkatraman 2004). The 24-hour knowledge factory is the epitome of leveraging global talent, specializing in the creative use of the available time afforded by a global workday. The

treatment of temporal structures as defining characteristics of organizations can lead to further understanding of these organizations (Orlikowski and Yates 2002). Sambamurthy and colleagues (2003) tie these factors directly to the IT investments that firms make by providing a framework for understanding that IT investments are indeed key influencers of firm performance along with factors such as agility and capability-building. Others have shown that structural diversity in work groups, including geographic diversity of team member locations, increases exposure to external knowledge sources and thus enhances knowledge sharing (Cummings 2004). Our paper extends these guiding frameworks, delineates the core principals and corresponding research areas related to the 24-hour knowledge factory, and tests the key feature of this model in a controlled field experiment.

Group Dynamics and Tacit Knowledge

There has long been a sense that face-to-face contact can facilitate creative interaction and produce more and better ideas (Osborn 1957). However, there is an equally long history of experimental findings that show that the aggregate output of so-called "nominal" or "concocted" groups of individuals working alone outperform "real" groups of the same number of individuals working together in person on creative tasks such as idea generation (Lorge et al. 1958; Mullen, Johnson, and Salas 1991; Taylor, Berry, and Block 1958). Real interactive groups consistently incur a "process loss" during group interaction that nominal groups avoid (Steiner 1972). Later experiments (Diehl and Stroebe 1987, 1991) found that the inability of all real group members to contribute their ideas simultaneously created a bottleneck that blocked potentially valuable contributions of some members and thus reduced the effectiveness of real groups. Subsequent studies have shown that providing real interactive groups with information technology tools enabled simultaneous creative production and removed social inhibition, thus eliminating the

9

production blocking problem (Paulus et al. 1996). In fact, real groups were found to be even more productive than nominal groups (Dennis and Valacich 1993; Valacich, Dennis, and Connolly 1994). Since globally distributed teams share key characteristics with nominal groups (members working more independently than collaboratively), and also with electronic interacting groups (team member interaction mediated by technology), the social psychology literature on small group dynamics implies that global virtual teams may in fact enjoy certain advantages relative to collocated teams. With the aid of electronic communication, the advantage of distributed teams over collocated interactive teams grows even further as group size increases (Gallupe et al. 1992).

While electronic tools allow distributed teams to work interactively (to some extent) and productively on creative tasks, they do not resolve the challenge of tacit knowledge, considered essential to innovative activities but difficult to transfer without face-to-face interaction (Kogut and Zander 1992; Nonaka, Toyama, and Konno 2000; Sternberg et al. 2000). Accessibility to ambient tacit knowledge has been posited as a major reason for firms to locate in close geographic proximity to other organizations within the same industry (Audretsch and Stephan 1996; Porter 1998; Rosenfeld 1997; Tallman et al. 2004). If correct, globally distributed teams may be missing a key ingredient that would help them function more effectively, suggesting that collocated product development teams may be preferred after all. The ultimate viability of the 24-hour knowledge factory will depend on the ability of distributed group members to work effectively on creative tasks and to work effectively in the absence of non-codifiable elements of tacit knowledge.

Prior studies suggesting contradictory advantages and disadvantages of distributed and collocated teams do not permit formulation of specific directional predictions about how each

type of team would perform on a similar product development task. Furthermore, the literatures

on the advantages of geographic proximity in innovation and on group process and creativity are

subject to important limitations. For example, the case for the necessity of physical proximity to

foster innovation is based mostly on selective anecdotal evidence, and has been refuted by a

recent large-scale survey study of high-technology firms that found no effect of geographic

distance on collaborative new product development outcomes (Ganesan, Malter and Rindfleisch

2005). Similarly, most experimental studies of group idea-generation and problem-solving

examined only very small ad hoc groups (3 to 5 persons) composed of relative strangers

(undergraduate students) who worked without leadership on a very short-term task (often only 10

to 60 minutes) involving a problem of no inherent interest or relevance to participants (Dunnette

1964; Dunnette, Campbell and Jaastad 1963). In contrast, real-world product development teams

consist of highly trained professionals working with similarly qualified colleagues on an

engaging and urgent problem, often over an extended period of time ranging from many weeks

to more than one year. Such teams are supervised by management, motivated by incentives, and

their performance carries real consequences for individual members and the organization.

Therefore, theoretical questions regarding the impact of collocation or distribution of members

on the effectiveness of product development teams should be evaluated using real teams working

in appropriate field settings and, ideally, tested under controlled conditions.

Case Examples of the 24-Hour Knowledge Factory

A number of recent studies describe the experience of organizations that have been early

adopters of the 24-hour knowledge factory model. Selected examples are presented in the table

below to provide context for our field study testing the implementation of the model. These

studies are mostly anecdotal but offer initial evidence that global teams are capable of sharing

knowledge and acting as local teams (Ishizaki 2005). Others have shown that roles can be effectively distributed across distant locations (Chandler 2001), and that fostering a sense of collective ownership can facilitate effective daily hand-offs of work items across shifts and around the globe (Yap 2005). However, a real test of the viability of the 24-hour knowledge factory requires a more systematic evaluation of the key assumptions underlying the model.

INSERT TABLE 1 ABOUT HERE

FIELD STUDY

Method

In a recent longitudinal study of a virtual product development team distributed across three organizations and three distant locations, Majchrzak and colleagues (2000) collected ethnographic and quantitative data to develop an in-depth understanding of the operation of the team from a social, organizational and technical perspective. They monitored the team's use of technology and organizational processes in implementing the multi-site development effort during the 10-month project using participant observation, recorded meeting minutes, electronic log files, weekly questionnaires and periodic interviews. Rich data of this type allowed the authors to observe the team's initial attempts to overcome misalignment through organizational changes, followed by subsequent success after organizational changes were combined with technological changes adapted to the team's geographic distribution. Other studies have examined multiple virtual teams, ranging from case studies of multiple teams (Maznevski and Chudoba 2000) to a large-scale survey of hundreds of teams (Cummings 2004), but these studies sought teams with maximally diverse structural characteristics and teams working on different

types of projects. While these types of studies provide new insights about the dynamics and boundary conditions of virtual teams, they are not designed to directly compare the effect of geographic collocation or distribution on group process and performance.

Our longitudinal study of collocated and distributed software development teams at the IBM Corporation follows the general approach of Majchrzak et al. (2000), collecting similar comprehensive data but with three key differences: (1) we compare two teams within one firm (rather than one team across three firms); (2) we manipulate the key variable of organizational structure (geographic distribution); and (3) we use more objective and finer-grained data on team member interaction and project outcomes. In contrast to descriptive case studies, the present study is a controlled field experiment comparing two teams with nearly identical characteristics except for the critical variable of interest: collocation versus geographic distribution of team members. The design is a "quasi-experiment" (Cook and Campbell 1979) in the sense that team members were not randomly assigned to each type of team, but similar team composition and controlling for other possible explanatory factors allows us to infer that collocation or distribution caused any observed differences in team performance. Each team was assigned the same number of members (7) with similar qualifications and experience, and each team worked on a nearly identical software development project (two parts of the same software package), subject to the identical time schedule (12 months) and deadlines, in the same corporation, and under the same project management and work rules. Importantly, the collocated team members all worked on the software during the same work hours, whereas the globally distributed team members shifted work back and forth across time zones in an asynchronous manner. The two sequential work shifts of the distributed team are less than the three consecutive 8-hour work shifts in the ideal 24-hour knowledge factory model, but dispersion of the team across 10 time

zones forced team members to work more independently during their respective shifts, providing a conservative test of the key feature of the model.

The present study employed state-of-the-art technology to collect more detailed and objective measures of group interaction and performance than previous published studies of product development teams. For example, software was created to draw from source code control systems and bug report databases that track the specific knowledge work product in real time, as it was created by each team member and by the team collectively, rather than relying on retrospective and perceptual self-reports by individual team members. Quantitative and qualitative data were collected systematically from the two teams over a one-year period (calendar year 2004). The teams' main project deliverable was on a one-year timeframe, so this period covers every major point in the project lifecycle from its inception to completion and delivery of the end product. Within this year, the teams also devoted a significant amount of time to short-term tasks such as attending to customer deployment issues and fixing bugs for maintenance releases; as such, the one-year timeframe also provides an opportunity to gain insights on knowledge sharing for all scopes and varieties of tasks. Figure 1 presents a diagram of the inputs, processes, and outputs involved with the IBM software development task. As noted above, the key differentiating input factor between the two groups was geography. All members of the collocated team were based at a single location in the U.S. In the globally distributed team, some members were based at an IBM center in India and the rest at an IBM center in the U.S.

INSERT FIGURE 1 ABOUT HERE

During the 12 months of the software development project, various types of data were collected. The data sources and measures were designed to provide a more complete picture of the knowledge sharing that occurred over time in terms of technical, organizational, social, and group process dimensions. The experimental design and quantitative measures enabled direct comparisons between the collocated and distributed groups on the key dimensions of interest.

Primary Data – Personal Interviews

Hour-long personal interviews were conducted with each of the developers on each team. While the focus of these interviews was primarily to gain qualitative insight, specific quantitative questions were asked in order to elicit the developers' own views of their knowledge sharing requirements. The following measures were collected:

- Number of informal interactions per week (informal interactions with fellow team members that did not begin with an intention of discussing business).

- Number of informal interactions with main developers

- Number of formal interactions with main developers

- Percent of informal communication in person

- Percent of informal communication via instant messaging

- Percent of informal communication via phone

- Number of tactical decisions made informally (decisions that are minor in scope, i.e., with minimal knowledge sharing requirements and minimal impact on other developers' work).

- Number of strategic decisions made informally (decisions major in scope, with significant knowledge sharing requirements and long-term impact on other developers' work).

- Number of strategic decisions that were speeded up informally

- Number of tactical decisions that were speeded up informally

Primary Data – Weekly Meetings over One Year, Coding System for Tasks, and Status

The weekly meetings of each team were analyzed to provide insight into the formal task allocation and knowledge sharing on a group-wide basis for each team. The minutes were taken by the project manager for the teams, who kept item-by-item records of the discussion. The Collocated Team held one face-to-face team-wide meeting per week, while the Distributed Team held one weekly face-to-face meeting for only the U.S.-based team members and one weekly coordination meeting via telephone between the development leads from the U.S. and India. Each of these three weekly meetings was analyzed for the entire year.

A manual review of the three meetings per week (one meeting for collocated team, one meeting for U.S. team members of the distributed U.S.-India team, and one U.S.-India team joint session) was conducted to distill the following data, with respect to each developer:

- Main Developers Interacted With
- Tactical Tasks Assigned
- Strategic Tasks Assigned
- Tactical Status Requests
- Strategic Status Requests
- Developer Input Requested
- Developer-to-Developer Information Requests

Archival Data – Software Problem Reports

Each development team kept track of fixes requested or made to the code base via Software Problem Reports (SPRs). These SPRs contained information on the problem being reported, as well as the history of knowledge provided by various developers in resolving the

issue and information regarding the actual fix to the issue. SPRs were stored in a central

database for each team. For this study, a software tool was written to collect the data from the

SPR archive. This tool analyzed SPRs fixed over the 12-month period of study, and collected

the specific data described below, for each developer, on a weekly basis:

- Main Developers Overlapped With – For a given SPR's primary developer, a listing of the

  other developers who provided input into the SPR.

- Average Delay Between Developer Inputs – For a given SPR, the average time between one

  developer's input and another developer's input.

- Ratio of Collaborative to Individual SPRs – A collaborative SPR is defined as one which

  includes input from more than one developer.

- Average Time to Resolution – The average time it takes for an SPR to move from being

  approved by the management team to be fixed to actually being logged as fixed.

Archival Data – Source Code Control System

Each of the two teams used a source control system to log the modifications made to each

element of the source code for the team's product. The source control system stored the date,

time, developer making the change, and a comment regarding the particular change. The

comments often cite particular SPRs if there was an SPR that initiated the particular change to be

made.

The goal of collecting data from the source control system was to ensure a clear depiction

of the technical system, which would complement the social and organizational systems

described by the other forms of data that were collected. The latter data provided a

representation of the technical dependencies between developers on the teams, and the rate of

technical collaboration within the teams. The following data were collected, with respect to each developer, on a weekly basis:

- Main Developers Interacted With – in modules for which multiple developers check in code, and the rate of shared check-ins with each of the other developers.

- Delay Between Check-Ins – the average time difference between modifications to a particular module.

- Reciprocal Check-Ins – the rate of check-ins: one developer performs a check-in, followed in time by another developer performing a check-in to the same module.

- If SPR cited, Average Developer Input on SPR – if the comments in a source control check-in refer to an SPR, the SPR was consulted to determine the number of updates posted to the SPR.

- Average Time Since Last Check-In – this provided an idea of the amount of code that was actively being modified at that specific point in time.

- Average Modules Checked In Per Build – Periodically, the source code is built into an executable version of the product. The frequency of this activity ranges from once or twice daily in the testing and fixing stages of the project lifecycle, to once every week in the design and implementation stages of the project lifecycle.

Archival Data – Group Email Exchanges

A software tool was written to analyze electronic mail (e-mail) sent to all members of each team. A "thread" refers to the entire set of messages written in response to an initial electronic broadcast or request for information. These data provided insights into the use of broadcast messages to share knowledge on the teams. The following data were collected, with respect to each developer, on a weekly basis:

- Main Developers Interacted With

- Threads Contributed To

- Average Delay Between Responses For Initiated Threads

- Number of Threads Initiated

- Average Length of Initiated Threads

- Average Number of Developers Per Thread Initiated

Results

The data collected above provided a dashboard of key indicators of the weekly

functioning and performance of the distributed (offshore and onshore) and collocated (onshore

only) teams over the complete 12 months of the project. Direct comparisons of outcomes for the

key process variables for the distributed and collocated teams, respectively, are presented in

Table 2. As seen from this table, members of the collocated team communicated much more

frequently via emails than did members of the globally distributed team, which is perhaps

surprising given that many of the collocated team members worked in the same hallway of the

same building. On average, there were also more developers per code element for the collocated

team, compared to the distributed team. The two teams differed most dramatically in the number

of source code modifications prior to the "feature freeze" deadline in week 41, with the

distributed team making 84.1 modifications compared to only 14.1 modifications by the

collocated team. In other words, the collocated team was able to approach a key product

development deadline with far fewer last-minute changes and their work on the software code

was more collaborative and involved more consultation than the work by the globally distributed

team. The two teams also used team meetings for very different purposes; the distributed team

meetings featured a significantly higher percentage of tactical (cf. strategic) agenda items and also much higher percentage of assignment items (cf. status items). But ultimately, despite the very different usage of information systems and meeting behavior, each team exhibited similar performance in terms of the quality and speed of their work (measured by weekly SPR actions and average time to resolve SPR's, respectively). These results are further elaborated in the sections to follow. Importantly, the globally distributed team achieved a similar level of performance at a lower nominal cost due to the lower wages of team members located in India, as compared to the United States.

INSERT TABLE 2 ABOUT HERE

Key Themes

The use of dashboard-style tools to provide continuous monitoring of team dynamics and task performance and the subsequent analysis of the data lead to a number of themes that enable one to evaluate the efficacy of the more generalized 24-hour knowledge factory model.

1. Same Technologies and Processes Can Be Applied Differently

The two teams clearly had very different ways of applying the same, or very similar, information technologies and processes. For example, electronic mail was used on the distributed team much more for discussions; however, on the collocated team, the use of e-mail was restricted to announcements and very short discussions. Similarly, the higher level of modifications to the SPR's in the distributed team suggested that they used this technology as a means of transferring information between team members and maintaining a record of status, while the collocated team could rely on synchronous communication for information sharing and

status reporting. In the case of meeting minutes, while the agenda categories were generally the same between meetings, the number of tactical items and number of task assignments were much higher in the distributed team. The distributed team used the meeting process as a means for addressing needs that the collocated team satisfied outside of the context of the formal weekly meeting.

These differences suggest that technology and processes that support knowledge sharing in a similar task can be built and implemented to explicitly serve different purposes (cf. Maznevski and Chudoba 2000). Barley (1986) provides a framework for assessing technology's role in a knowledge-based work environment and suggests that the context in which the work is performed can significantly impact the way the technology is used. Teams would be well-served to assess their own uses of technologies and to determine if explicit changes need to be made to the particular technologies which are being used. For example, extensive use of electronic mail for discussions may warrant a greater use of discussion databases that are designed for long threaded discussions, while the use of electronic mail for announcements may warrant a greater use of instant messaging systems or the creation of a dynamic team web site where announcements may be posted.

The differences also suggest that managers should be cognizant of differing uses of technology and process when assessing their own team's progress. Managers need to adjust their expectations on the level and style of communication based on the geographic structure of the team. Collecting data similar to what was collected in this study to provide a "dashboard" for managers to continually assess the level of collaboration and progress on their team could be very useful. However, misaligned expectations could lead to gross misinterpretations of these data. For example, if a team accomplishes most of its tasks in an ad hoc and informal manner,

periodic monitoring of a system such as the SPR system may indicate that the team is making much less progress than is actually the case, since the SPR database may be missing key tacit knowledge that is discussed in informal (and hence unrecorded) interactions between collocated team members. One needs to understand the normal use of technology and processes and then adapt expectations appropriately. The development manager of both teams acknowledged this need for adapting management practices to each team. He stated that when he first took over the collocated team, he had been used to the processes of the distributed team and "that didn't work well for the team." He had since altered his use of e-mail, in-person visits, and meetings to adjust to the differing needs of each team, and achieved greater success in the resulting knowledge flow on each team.

2. Social Relationships and Technical Productivity Are Linked

The degree of social interaction between developers on a team was shown to have an impact on the technical productivity of the team, which then led to tighter social relationships. Developers on both teams cited the comfort level between team members as important in facilitating creative discussions, where developers do not have to worry about feeling embarrassed by a poor idea. The source code data showed that the collocated team had more examples of code elements that were modified by multiple team members, and interviews confirmed that this was likely because of the greater degree of social interaction on this team, rather than any technical reason that one software product merited more intertwined technical interaction than the other. The interview sessions also revealed many cases where a casual or informal interaction led to a technical decision being made. While such social relationships are much easier to form when collocated, the experience of the one U.S. developer on the distributed team who traveled to India suggests that social relationships can be built across distant

geographic and cultural boundaries and these relationships can indeed be leveraged to satisfy technical goals.

3. Collocated/Distributed Teams Behave Differently at Different Project Stages

The data showed that each team exhibited a different pattern of activity at the different stages of the project schedule. In the first three months of the study, as the teams approached a final product release, the distributed team spent 100% of their meetings discussing tactical issues, and had significant SPR activity. In the next seven months of the study, the teams approached "feature freeze", which represents the major milestone for software developers during a release when code for all features in the new release is submitted for incorporation into the product. As this milestone was passed, the collocated team had a lot of coordination to do with regard to the source code, and also saw an increase in e-mail activity which did not occur during the period before the milestone. None of these behaviors were cited in the interviews as being particularly positive or negative; they were instead seen by each team member as a normal reaction to various stages of the project for their particular type of team (collocated or globally distributed). Managers need to be aware of such differing patterns of activity which may surround key milestones of a project, and set expectations appropriately given the geographic distribution of the team they are managing.

4. Geographic Structure Does Not Define Destiny

The geographic structure of the teams in this study led to different forms of value being achieved from their knowledge sharing processes; however it does not follow that the destiny of each team is necessarily defined by its structure, as discussed below and as shown in Figure 2. The structure of the distributed team led them to have a higher degree of documented decisions – which is apparent in the data through increasing use of emails, tactical meeting items, and SPRs.

Interviews with members of this team confirmed that a very valuable, though perhaps, unintended outcome of this documentation process was that the history of the team's decision making was better retained. Interviews with the collocated team indicated that it would not have been feasible to enforce the same level of documentation on a collocated team, though this need should not be seen as a barrier to retaining team decision history. Instead, alternative processes such as scheduled time for documentation or the implementation of automated documentation tools could be used to achieve the same benefit for the collocated team. Similarly, the collocated team cited more frequent informal communication as a process which led to higher degrees of finding new and creative solutions. Even though these informal meetings generally occurred face-to-face, the distributed team could still achieve a similar outcome. Two suggestions, provided during interviews with the distributed team on this specific topic were a one-time face-to-face meeting that would introduce team members and bring a social component to the relationships, and the use of explicitly informal phone calls where no agenda or topic is preplanned so that there would be potential for team members to discuss any open-ended topic. Table 3 summarizes the major advantages of globally distributed and collocated teams.

INSERT FIGURE 2 ABOUT HERE

INSERT TABLE 3 ABOUT HERE

The examples above highlight the more general theme that structure does not need to define destiny. Organizations having teams with different structures should make explicit attempts to understand the value obtained by different team structures and determine ways to

adapt the processes that may be more natural in one structure to the other in order to achieve similar desired outcomes in multiple scenarios.

5. Individuals Heavily Influence Coordinated Development

Individual differences in personality and behavior on a distributed team can have a significant impact on the success of coordinated development efforts. For example, U.S.-based Developer 1 on the distributed team made a significant number of contacts with members of the Indian team; this led to his collaboration with them being much more frequent and useful. The project manager who kept minutes of meetings from both teams was very meticulous in the description of meeting items; this led to better history retention on both teams and also to more accurate data collection for this study. Another example is Developer 4 on the collocated team, who cited that he strongly preferred collocated teams and adapted his personal work style to fit the collocated model. This is an example of an individual who recognized the individual effort needed to work in a distributed versus collocated manner, and identified the most appropriate mode for his circumstance. In all of these cases, the behavior of an individual was not predetermined by the geographic structure of the team, but it did significantly impact team success.

6. Advantages of Geographic Distribution

Our finding that both collocated and geographically distributed teams were capable of successful collaboration suggests that common themes in the literatures on offshoring (offshoring is a win-lose zero-sum proposition), innovation (geographic distribution is a barrier to overcome) and social and organizational psychology (face-to-face groups are less productive) may all be inaccurate. Instead, geographic distribution should be seen as a potential asset which can be leveraged, especially across time zones. Numerous benefits from leveraging a dispersed

geographic structure were cited in interviews with the distributed team. Examples include an increase in documentation and history retention; the ability to share short-term tasks which required immediate attention so that work could be performed around the clock; and a more structured and explicit definition of work tasks and distribution of work items. The interviews cited a common understanding that many of these behaviors were a natural reaction to the geographic structure; however, the value of these behaviors may only have been realized and appreciated in retrospect. The core theme is that when managing a distributed team, the distribution should be looked upon as a positive characteristic to leverage rather than a barrier to overcome. Similarly, the performance of the collocated team showed the value of informal face-to-face interaction on a creative knowledge-production task despite having less explicit documentation and the potential for greater social inhibition and other inefficiencies.

7. Leveraging Organizational Assets, Decision-making Processes, and Cultural Diversity

At the organizational level, the data collected in this study demonstrates the potential for organizations to assess tacit knowledge capital that is not readily quantifiable. Organizations currently assess knowledge capital by counting the number of patents filed or tabulating features on existing products, but with the data capture tools used in this study, organizations can assess knowledge capital at a much more granular level. One can now assess the dependency on one development site or another, and on one developer or another. This allows the flow of knowledge capital to be tracked, and is especially important in domains where knowledge flows dynamically between geographic locations.

Further, the data collection techniques used in this study offer the potential to expose the true locus of decision-making within a group, both in collocated and distributed scenarios. Whereas on the surface it may appear that decisions are made in meetings held at or near the

corporate headquarters, it may become apparent through the data analysis that individuals on the team reach consensus through informal email discussion or hallway conversations and then use the meetings as a formal approval mechanism.

From the cultural perspective, an organization's culture is greatly influenced by its geographic structure. On the surface, this study was not an examination of contrasting regional cultures, but the impact of regional cultures on organizational cultures can be seen in the data. For example, it is possible to analyze the proclivity towards strict processes or the tendency to make decisions in ad hoc brainstorming sessions. Thus, the interrelated structural, political and cultural dimensions of organizations could be profoundly impacted by the introduction of detailed data on knowledge sharing within an organization.

Conclusions

In this paper we introduced the 24-hour knowledge factory model, considered prior theory and findings in social psychology and business strategy that are relevant to implementing the model, developed a dashboard methodology for using primary and archival data to monitor use of the model, and conducted an experimental test of the key feature of the model – the ability of distributed teams to collaborate successfully around-the-clock and around-the-globe. We find that offshore decentralization of product development can succeed with proper design and management of the dispersed team, and strategic use of information systems. These findings suggest that firms can transition from simple offshore service operations to global partnerships by implementing a 24-hour knowledge factory and leveraging geographic distance for strategic gain.

The implications for information systems theory and practice of each of the key themes identified in our field study are summarized in Table 4 below. Our findings show that information systems can play a major role in facilitating work productivity and social relationships among globally distributed team members who do not have an opportunity to meet face-to-face during a project, supporting and extending prior findings on electronic group brainstorming (e.g., Gallupe et al. 1992). These systems facilitated effective group interaction while also preserving some advantages enjoyed by "nominal" groups of individual team members working independently without process loss from direct personal interaction. The unique dashboard measures developed for this study also allow managers to better monitor progress of the distributed team, which is another critical challenge posed by a 24-hour knowledge factory operation.

This study also demonstrated that the geographic structure of a team (collocated or globally distributed) does not predetermine team outcomes. With increased understanding of the dynamics of each type of team, as learned in this study, we find that information systems can be employed to help each team increase its effectiveness. Neither structure is inherently superior; both are workable models with proper adaptations. The results also show that geographic distribution can be leveraged as an asset to take advantage of the possibility of continuous engagement on tasks across time zones. Distribution of the team across multiple time zones provides exposure and access to greater knowledge resources, thus increasing the knowledge capabilities of the product development team.

INSERT TABLE 4 ABOUT HERE

Our study also amplifies and extends the findings of previous studies of virtual teams. For example, Cummings (2004) focused on external knowledge sharing as opposed to intra-team knowledge sharing, and links structural diversity to a higher degree of external knowledge sharing. Our study focuses primarily on internal knowledge sharing but reaches a similar conclusion that having structural diversity does lead to a change in knowledge sharing and knowledge reuse. We examined one specific form of structural diversity, which is the distribution of team members geographically. The information systems developed in this study to capture rich data can also be used to capture external knowledge sharing of the type examined by Cummings, but at a more granular level.

The design used by Maznevski and Chudoba (2000) incorporated face-to-face visits, and in our case there was no face-to-face interaction at all. They found that communication patterns are adapted to the task, whereas in our paper, the two teams worked on nearly identical tasks (a control in the experiment) yet the communication patterns that developed were drastically different. We conclude that the communication patterns are a result of structural diversity, and that one pattern is not inherently superior. Instead, we suggest that the patterns must be understood in order to realize the potential benefits of each type of structure.

Our study provides an important extension to Majchrzak and colleagues (2000) who find that the role of technology in facilitating virtual teams is more important than organizational structure. In contrast, we propose a more granular look at the use of technology by mining the email system, source control system and software problem report system of each team, in order to not rely exclusively on interviews and meeting minutes to capture the use of technology in team processes. We cover both the social and technical networks by looking at critical factors such as number of developers per code element. Based on this approach, we show that

technology not only has a role in social knowledge sharing but also in the technical architecture of the work.

The present study also offers a number of implications for improving the management of information systems in practice. Specifically, we found that project milestones cause different levels of usage in the supporting information system. Thus, managers need to design the system to adapt; for example, to anticipate spikes before milestones for globally distributed teams. In addition, we found that open access and availability within information systems can allow members of a distributed team to view and participate in any aspect of the endeavor. Overall, this study suggests guidelines for adapting the design of information systems for the different structural characteristics and processes of teams in a 24-hour knowledge factory.

Limitations

Though the present study provides an important contribution by directly comparing the performance of two product development teams with different geographic structures, many of the controls in the experimental design which are strengths of the study are also limitations. For instance, we limited our focus to an in-depth investigation of only two teams, one distributed and one collocated, both within the same multi-national corporation. Also, the distributed team in the present study operated in only two time zones, providing a conservative test of the 24-hour knowledge factory concept. An ideal study would use a task that is handed off from shift to shift to shift across three time zones on a daily basis.

Since the distributed and collocated teams worked on similar projects and on identical timelines, the current design did not enable a complete test of product completion speed. We were able to observe the speed of each team's response to approaching deadlines and to resolution of Software Problem Reports (SPR's), but it was beyond the scope of the present

study to evaluate whether the distributed team would have been able to deliver the end product in shorter overall time. In future tests of project completion speed, investigators need to control for total number of man-hours. The present study controlled for both team size and project duration (hence also total man-hours), which allowed us to test key behavioral aspects of the 24-hour knowledge factory. However, if these constraints are relaxed, the 24-hour knowledge factory needs to complete a project at least three times faster than a collocated team in order to demonstrate superiority, since three distributed work shifts in 24 hours utilize three times as many man-hours as one collocated shift per day. Thus, testing project completion time involves a number of design trade-offs. For example, an experimental design that tests for overall project completion speed essentially distorts the time-scales of the two teams, which does not allow for direct comparisons of periodic behaviors, such as weekly emails, threads, meetings, and SPR's, or actions related to mid-project milestones, that generated important insights in the present study.

Future Research Directions

The findings of the present study suggest a number of future research directions to further test the viability and value of the 24-hour knowledge factory.

Larger Number of Time Zones. Future studies should test the effect of incorporating additional time zones on the performance of a 24-hour knowledge factory. The original model (Gupta and Seshasai 2004) proposed three sequential 8-hour work shifts distributed across time zones around the globe. While additional work shifts are generally predicted to accelerate productivity relative to fewer shifts or relative to a collocated team (essentially, one shift per day), a larger number of shifts necessitates a larger number of hand-offs between shifts, which will require additional transition time between shifts in order to document progress and update

the next shift on developments during the previous shift. Adding time zones to the design also necessarily adds new logistical and cultural differences among team members distributed across three or four countries rather than two, which may further complicate communication and affect team productivity.

Larger Number of Teams. Future investigations should examine a larger number of teams. Maintaining the same controls on team size, scope and product, it would be very useful to collect data on a larger number of teams to validate our initial findings regarding the effect of the geographical structure of the team.

Results of Weekly Feedback to a Team. If teams received weekly feedback related to the data which were collected that week, the behavior of the team may change to adapt their work to what they learn. The interviews conducted in this study were done before and after the quantitative data were collected, and in discussing the data and results with team members, it was clear that some insights were not obvious to team members and would cause definite changes in interaction. Providing feedback to one team while not providing it to another could also be an interesting experimental manipulation and could validate the need for better technology to make teams more aware of their own knowledge-sharing practices.

Collection of Other Data. Figure 1 showing the inputs, processes and outputs of a software development team included a number of other potential sources of data which would provide a more complete picture of knowledge sharing on distributed and collocated product development teams. Other possible measures of communication processes include logs of instant messaging usage, phone usage, development schedules and logging of informal interactions. Collection of measures of personality characteristics of each team member could also be used to better understand the human resource requirements of a 24-hour knowledge factory and how

individual personality differences enhance or inhibit trust and inter-personal interaction within product development teams (distributed and collocated).

Varying Other Team Characteristics. Teams of different size, in a different technical domain, responsible for a product in a different stage of the lifecycle, or in different companies would all provide an interesting contrast to the present study. There are also different types of software development teams beyond those producing new product releases as in the present study. For example, software development teams can act in a consulting role, producing short-term solutions for one customer, or in a support role, solely producing fixes for an existing product in the market. Each of these different types of teams has a different type of knowledge to share and different methods of sharing that knowledge. Applying the data collection techniques developed in this study to those teams would provide additional insights into how geographic distribution affects forms of knowledge sharing by task type.

Other Knowledge-Based Industries. Many other industries beyond software development feature distributed team work, and similar archival data from such teams could help to better understand the impact of geographic distribution or collocation. Directly comparable industries such as mechanical engineering may provide insight that can be incorporated back into the findings from the software industry. Additionally, industries where work occurs in one location but on a time-shifted basis may also benefit from similar study. One developer on the collocated team compared software development on a distributed team to the practice of nursing: when a nurse completes a shift, she or he must transfer the knowledge obtained during that shift to the next shift of nurses, and this is often done in a hybrid form of structured and unstructured communication mechanisms. Industries such as nursing are not directly related to the product

design and engineering domain of software engineering, but may still provide fertile ground for future research.

Technology Prototypes. A next step in applying these findings is to build technology prototypes which would improve access to the type of data collected in this study. Such prototypes could be useful in acting as a "dashboard" for team managers to monitor social and technical collaboration on their teams, as well as providing useful indicators of the team's progress towards goals or milestones. The technology prototypes could also be used to train individual team members on their team's collaborative processes, and also on the processes of other teams. Finally, technology could be built to collect even more data than was possible in this study. Much of the data collected in this study was in inconsistent formats and required a significant degree of manual cleaning before being incorporated into the analysis. If technology could be built to parse through much more diverse sets of data such as web discussion forums, document revisions and comment lines within source code, an even richer picture of the knowledge sharing on global product development teams could be obtained.

In sum, this paper highlights the concept of the 24-hour knowledge factory and tests the model in a controlled field experiment that directly compares the use of information systems and subsequent performance in collocated and globally distributed software development teams. Geographic distance introduces new challenges but these can be overcome – and even leveraged for strategic advantage. Our findings suggest that firms can apply the 24-hour knowledge factory model to transition from a service provider framework in which offshoring is a short-term and unilateral cost-saving tactic to a strategic partnership in which offshoring becomes a core component of a global corporate strategy.

REFERENCES

Andre, D. "Case Study: Technology Offshoring at Upromise". Presentation to Massachusetts Software Council, Apr. 13, 2004.

Asprey, W., Mayadas, F., Vardi, M. "Globalization and the Offshoring of Services: A Report of the ACM Job Migration Task Force." *Association of Computing Machinery*, 2006.

Audretsch, D.B. and Stephan, P.E. "Company-Scientist Location Links: The Case of Biotechnology," *American Economic Review* (86:June), 1996, pp. 641-652.

Barley, S.R. "Technology as an occasion for structuring: Evidence from observations of CT scanners and the social order of radiology departments." *Administrative Science Quarterly* (31:1), 1986, pp. 78-108.

Baxter, S.P. "A Business Primer: How to Expand into China". Presentation to MIT Special Seminar on International Management - Offshoring, Apr. 10, 2004.

Chandler, H.E. "The Complexity of Online Groups: A Case Study of Asynchronous Distributed Collaboration." *ACM Journal of Computer Documentation* (25:1), 2001, pp. 17-25.

Clark, K., Wheelwright, S. "Organizing and Leading 'Heavyweight' Development Teams." *California Management Review* (34:3), 1992, pp. 9-28.

Conrath, David W. "Communication Environment and its Relationship to Organizational Structure," *Management Science* (20), 1973, pp. 586-603.

Cook, T.D. and Campbell, D.T. *Quasi-Experimentation: Design & Analysis Issues for Field Settings,* Houghton-Mifflin, Boston, 1979.

Cummings, J.N. "Work Groups, Structural Diversity, and Knowledge Sharing in a Global Organization," *Management Science* (50:3), 2004, pp. 352-364.

Dennis, A.R., and Valacich, J.S. "Computer Brainstorms: More Heads Are Better Than One," *Journal of Applied Psychology* (78:4), 1993, pp. 531-537.

Diehl, M., and Stroebe, W. "Productivity Loss in Brainstorming Groups: Toward the Solution of a Riddle," *Journal of Personality and Social Psychology* (53:3), 1987, pp. 497-509.

Diehl, M., and Stroebe, W. "Productivity Loss in Idea-Generating Groups: Tracking Down the Blocking Effect," *Journal of Personality and Social Psychology* (61:3), 1991, pp. 393-403.

Dunnette, M.D. "Are Meetings Any Good for Solving Problems?" *Personnel Administration* (27:(March-April), 1964, pp. 12-16, 29.

Dunnette, M.D., Campbell, J. and Jaastad, K. "The Effect of Group Participation on Brainstorming Effectiveness for Two Industrial Samples," *Journal of Applied Psychology* (47:1), 1963, 30-37.

Ellis, J. "Building an offshore contract engineering business." Presentation to MIT Special Seminar on International Management – Offshoring, Apr. 10, 2004.

Eppinger, S.D. and Chitkara, A.R. "The New Practice of Global Product Development," *MIT Sloan Management Review* (47:4), 2006, pp. 22-30.

Gallupe, R.B., Dennis, A.R., Cooper, W.H., Valacich, J.S., Bastianutti, L.M., and Nunamaker, J.F. Jr. "Electronic Brainstorming and Group Size," *Academy of Management Journal* (35:2), 1992, pp. 350-369.

Ganesan, S., Malter, A.J., and Rindfleisch, A. "Does Distance Still Matter? Geographic Proximity and New Product Development," *Journal of Marketing* (69:4), 2005, pp. 44-60.

Godart, C, et. al. "Asynchronous coordination of virtual teams in creative applications(co-design or co-engineering): requirements and design criteria." *Workshop on Information Technology for Virtual Enterprises*, 2001, pp 135-142.

Gupta, A. and Seshasai, S. "Toward the 24-Hour Knowledge Factory," *MIT Center for e-Business working paper Number 203*, 2004.

IEEE "IEEE Position Statement on Offshore Outsourcing." *IEEE-USA Career and Workforce Policy Committee*, 2004.

Ishizaki, Y. "Supporting the Next Generation of Mission-Critical Systems". *FUJITSU Scientific Technology. Journal*, (41:3), 2005, pp. 334-340.

Kogut, B. and Zander, U. "Knowledge of the Firm, Combinative Capabilities, and the Replication of Technology," *Organization Science* (3:August), 1992, pp. 383-397.

Lorge, I., Fox, D., Davitz, J., and Brenner, M. "A Survey of Studies Contrasting the Quality of Group Performance and Individual Performance, 1920-1957," *Psychological Bulletin* (55:6), 1958, pp. 337-372.

Majchrzak, A., Rice, R.E., Malhotra, A., King, N., and Ba, S. "Technology Adaptation: The Case of a Computer-Supported Inter-Organizational Virtual Team," *MIS Quarterly*, (24:4), 2000, pp. 569-600.

Malhotra, R. "Unique Business Model." Presentation to MIT Special Seminar on International Management, Apr. 10, 2004.

Maznevski, M.L., and Chudoba, K.M. "Bridging Space Over Time: Global Virtual Team Dynamics and Effectiveness," *Organization Science* (11:5), 2000, pp. 473-492.

Mullen, B., Johnson, C., and Salas, E. "Productivity Loss in Brainstorming Groups: A Meta-Analytic Integration," *Basic and Applied Social Psychology* (12:1), 1991, pp. 3-23.

Nonaka, I., Toyama, R., and Konno, N. "SECI, Ba, and Leadership: A Unified Model of Dynamic Knowledge Creation," *Long Range Planning* (33:February), 2000, pp. 5-34.

Orlikowski, W., and Yates, J. "It's About Time: Temporal Structuring in Organizations." *Organization Science* (13:6), 2002, pp. 684-700.

Osborn, A. F. *Applied Imagination,* (Rev. ed.), New York: Scribner, 1957.

Paulus, P.B., Larey, T.S., Putnam, V.L., Leggett, K.L., and Roland, E.J. "Social Influence Processes in Computer Brainstorming," *Basic and Applied Social Psychology* (18:1), 1996, pp. 3-14.

Porter, M.E. "Clusters and the New Economy of Competition," *Harvard Business Review* (76:November-December), 1998, pp. 77-90.

Ramesh, V. and Dennis, A.R. "The object-oriented team: Lessons for virtual teams from global software development." *HICSS. Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, 2002, pp 212- 221.

Rosenfeld, S.A. "Bringing Business Clusters into the Mainstream of Economic Development," *European Planning Studies* (5:1), 1997, pp. 3-23.

Saini, S. "Offshoring Experiment in Radiology". Presentation to MIT Special Seminar on International Management - Offshoring, Feb. 25, 2004.

Sambamurthy, V., Bharadwhaj, A., Grover, V. "Shaping Agility through Digital Options: Reconceptualizing the Role of Information Technology in Contemporary Firms," *MIS Quarterly* (27:2), 2003, pp. 237-263.

Sepulveda, C. "Agile development and remote teams: learning to love the phone." *Proceedings of the Agile Development Conference,* 2003, pp 140-145.

Shah, S. "Offshore Development." Presentation to Massachusetts Software Council, Apr. 13, 2004.

Srinivasan, V. "Outsourcing and Offshoring" Presentation to MIT Special Seminar on International Management - Offshoring, Feb. 25, 2004

Steiner, I.D. *Group Process and Productivity*, Academic Press, New York, 1972.

Sternberg, E. "The Sectoral Cluster in Economic Development Policy: Lessons from Rochester and Buffalo, New York," *Economic Development Quarterly* (5:November), 1991, pp. 342-356.

Sternberg, R. J., Forsythe, G.B., Hedlund, J., Horvath, J.A., Wagner, R.K., Williams, W.M., Snook, S.A., and Grigorenko, E.L. *Practical Intelligence in Everyday Life*. New York: Cambridge University Press, 2000.

Suh, B. "Perspectives on Global Delivery". Presentation to MIT Special Seminar on International Management - Offshoring, Feb. 25, 2004

Swadia, S. "Increasing Software Development Efficiencies." Presentation to Massachusetts Software Council, Apr. 13, 2004.

Tallman, S., Jenkins, M., Henry, N. and Pinch, S. "Knowledge, Clusters, and Competitive Advantage," *Academy of Management Review* (29:2), 2004, pp. 258-271.

Taylor, D. W., Berry, P.C., and Block, C.H. "Does Group Participation When Using Brainstorming Facilitate or Inhibit Creative Thinking?" *Administrative Science Quarterly* (3), 1958, pp. 23-47.

Treinin, J.J. and Miller-Frost, S.L. "Following the Sun: Case Studies in Global Software Development," *IBM Systems Journal* (45:4), 2006, forthcoming.

Valacich, J.S., Dennis, A.R., and Connolly, T. "Idea Generation in Computer-Based Groups: A New Ending to an Old Story," *Organizational Behavior and Human Decision Processes* (57), 1994, pp. 448-467.

Venkatraman, N. "Offshoring Without Guilt," *Sloan Management Review* (45:3), 2004, pp. 14-16.

Venkatraman, N., and Subramaniam, M. "Theorizing the Future of Strategy: Questions for Shaping Strategy Research in the Knowledge Economy." *The Handbook of Strategy and Management*, Sage Publications, 2002, pp. 461-474.

Weber, R. "Editor's comments. Some implications of the Year-2000 era, Dot.com era, and Offshoring for information systems pedagogy." *MIS Quarterly* (28:2), 2004, pp. iii-xi.

White, L. "NSPE Issue Brief: Offshore Outsourcing of Engineering Work." *National Society of Professional Engineers*, Publication no. 4065, 2004.

Xiaohu, Y., Bin, X., et al. "Extreme Programming in Global Software Development", *Canadian Conference on Electrical and Computer Engineering* (4), 2004, 1845-1848.

Yap, M. "Follow the Sun: Distributed Extreme Programming Development," *Proceedings of the Agile Conference* 2005, pp. 218-224.
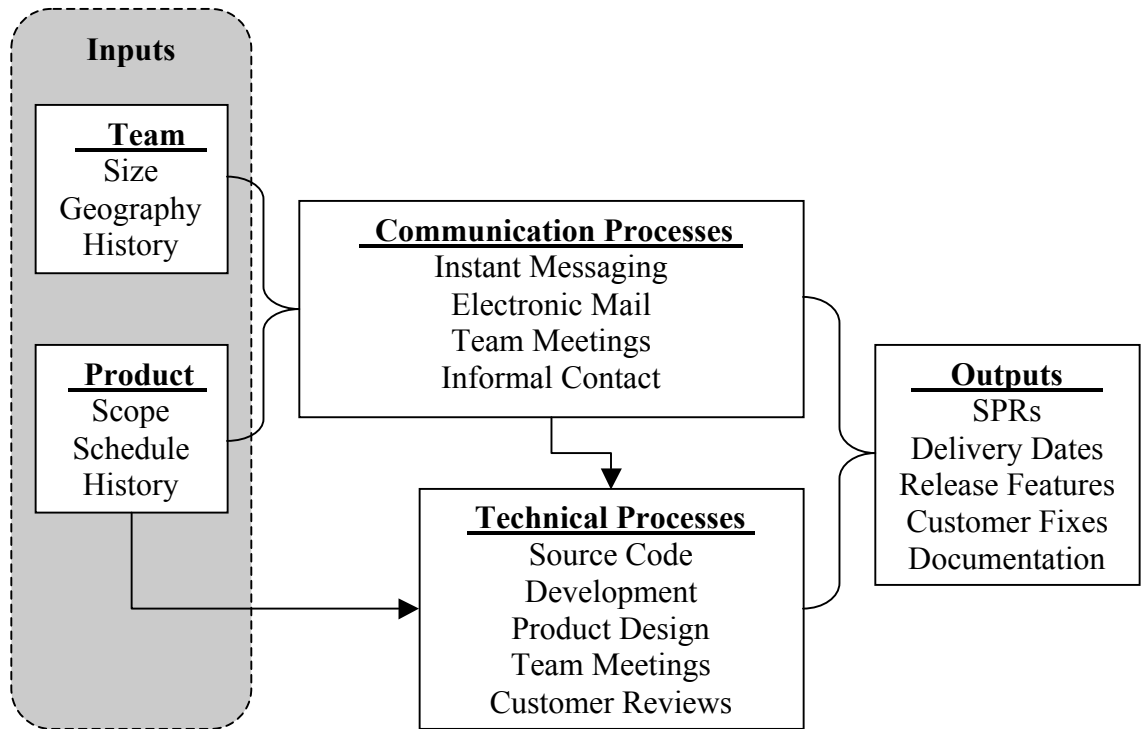
Figure 1: The Inputs, Processes, and Outputs involved in the system of software development are shown. The team and product characteristics are put through a series of communication processes that generate both the input into the technical process and the outputs. The product inputs can also be directly applied to the technical processes.
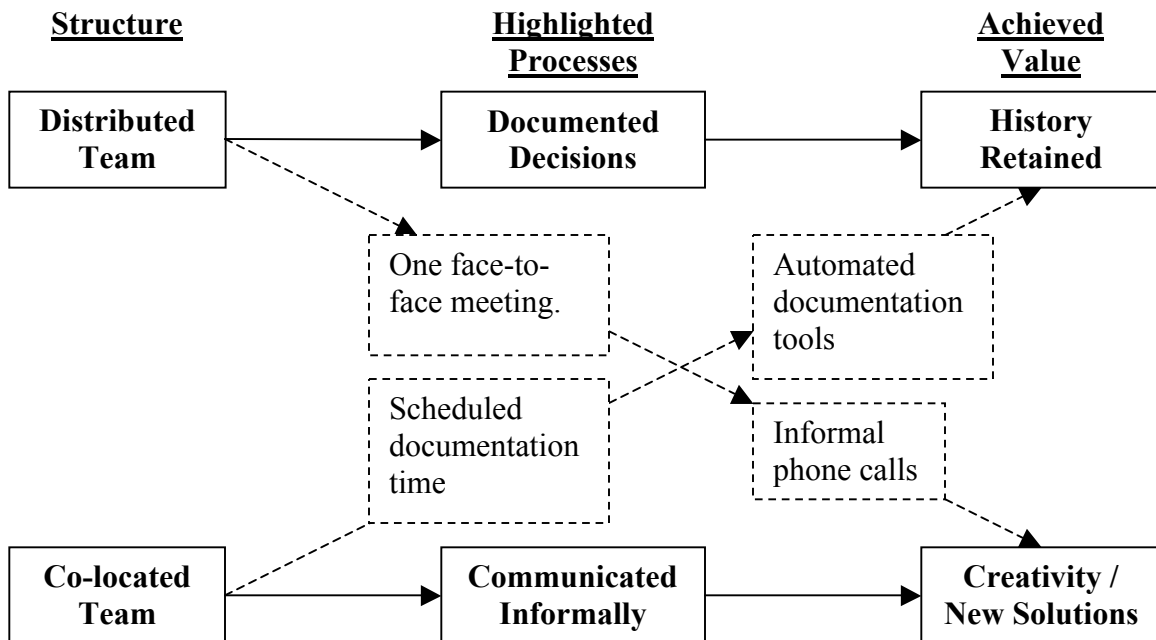
Figure 2: Example scenarios where the geographic structure of a team leads to a process that achieves a desirable outcome which can also be achieved in a different manner by the team whose structure does not naturally allow for the highlighted process. The alternative path to the desired outcome for each team is indicated by dashed lines and boxes. In sum, the tight link between team structure, group process and outcomes does not need to exist.

Table 1. Case Examples of the 24-Hour Knowledge Factory

Reference	Summary	Relevance to the 24-hr Knowledge Factory
Ramesh and Dennis (2002)	"Object oriented team" – decoupled virtual teams, communicating only inputs and outputs.	Knowledge sharing has potential to break barriers to decoupling teams.
Xiaohu, Bin, et al. (2004)	Global units owned their own module, no discussion except for inputs/outputs.	Knowledge sharing has potential to break barriers to separating modules.
Ishizaki (2005)	Fujitsu offers 24/7 support by combining local teams, continental teams (NA, EMEA, AP) and global team to handle mission-critical systems.	Global teams can act as local teams by having complete knowledge shared daily.
Chandler (2001)	Roles on a virtual team were defined without regard for remote vs. local.	Roles can be shared globally, or distributed globally without concept of remote and local.
Clark and Wheelwright (1992)	Motorola's virtual team project completed in half time with higher quality, focus on lesson reuse from previous efforts.	Lesson reuse can be applied to global teams of multiple sites.
Yap (2005)	A global, round-the-clock development project with collective ownership of individual pieces of code.	Collective ownership is the key to passing work items around the clock daily.
Sepulveda (2003)	Pair programming with remote and local developers cut out bureaucracy but negatively impacted culture.	Pair programming concept can be extended to three or more sites and individuals.
Godart, et al. (2001)	Implicit (voluntary) versus explicit (forced) coordination is debated. Benefit of implicit is that it has minimal impact on work methods, benefit of explicit is it allowed better tracking and reuse.	Implicit coordination becomes explicit coordination because it's part of the daily schedule.

Table 2. Comparison of Outcomes for Key Process Variables

Process Variable	Distributed Team	Collocated Team	Effect on Distributed (24-Hour Knowledge Factory) Team
Contributors per email thread	1.73	1.50	Distributed team used more written communication for group discussion.
Average weekly email threads	10.42	19.85	Distributed team used less broadcast style email messages.
Average emails per thread	2.32	1.75	Distributed team conducted longer discussions in written form.
Average weekly emails	17.06	29.91	Distributed team sent fewer logistical messages to the group.
Source code modifications prior to deadline	84.10	14.10	Distributed team used source code modification process to resolve issues in place of informal collaboration before 'feature freeze' date.
Average number of developers per code element	1.10	1.63	Distributed team's socio-technical system was less interconnected.
Percent of tactical (vs. strategic) meeting agenda items	81.36	39.23	Distributed team relied more on meetings for short term issues.
Percent of task assignment (versus status) meeting agenda items	35.08	24.53	Distributed team assigned tasks formally in meeting format.
Average Software Problem Report (SPR) actions per week	76.49	70.13	Distributed team had similar level of output in terms of quality as collocated team.
Average # of individuals modifying SPR state	3.25	1.74	Distributed team relied more on formal knowledge capture systems.
Average SPR time to resolution	113.80	120.72	Distributed team had similar level of output in terms of productivity as collocated team.

Table 3. Major Advantages of Each Type of Team

Distributed (24-Hour Knowledge Factory) Team	Collocated Team
Formal logging of knowledge	Issues resolved informally, in a timely manner
Structured use of processes	Incidental interaction led to productivity
Explicit role definition via tasks	Meetings focused on strategic discussion
Exploiting technology for collaboration	Technical system more collaborative

Table 4. Recommendations for Improving Team Performance

Major Findings from Field Study	Implication for Information Systems
Same Technologies and Processes can be Applied Differently (and both effectively)	Design information systems for different teams differently based on the structural characteristics and processes of each team.
Social Relationships and Technical Productivity are Intertwined	Information systems can play major role in facilitating social communication among distributed teams.
Teams React Differently to Different Stages of the Project	Project milestones cause different levels of usage in the supporting information system; as such, one needs to design the system to adapt, e.g., anticipate spikes before milestones for globally distributed teams.
Structure Does Not Define Destiny	A team is not destined for success or failure by the geographic constitution of the team.
Individuals can Heavily Influence Coordinated Development	Open access and availability within information systems can allow members of coordinated team to view and participate in any aspect of the endeavor.
Geographic Distribution can be an Asset	Information systems that facilitate structured knowledge capture can exploit spatial distribution by leveraging continuous engagement on tasks.