December 2006

Case Research in Global Software Projects: Coordinating Through Knowledge

Julia Kotlarsky
Warwick Business School

Paul van Fenema
Netherlands Defense Academy

Leslie Willcocks
London School of Economics

CASE RESEARCH IN GLOBAL SOFTWARE PROJECTS: COORDINATING THROUGH KNOWLEDGE

Knowledge Management

Julia Kotlarsky
Warwick Business School
The University of Warwick
Julia.kotlarsky@wbs.ac.uk

Paul C. van Fenema
Netherlands Defense Academy
pfenema@gmail.com

Leslie P. Willcocks
London School of Economics
l.p.willcocks@lse.ac.uk

Abstract

This paper brings together two challenging areas for both information systems (IS) research and practice: forms of coordination, and the efficacy of global, virtual software development projects. The contribution of the present paper is to develop a more comprehensive, knowledge-based model of how coordination can be achieved, and to illustrate the heuristic and explanatory power of the model when applied to global software projects experiencing different degrees of success. We first detail the literature on coordination, and show also what is known about coordination experiences in global software projects. We then develop, through a literature critique, a new, distinctive knowledge-based model of coordination. This is then employed to analyze two case studies of global software projects, at SAP and Baan, which illustrate the utility of the model. Finally, the paper discusses implications of the findings for future IS research.

Keywords: Knowledge, coordination, global software projects

Introduction

Coordination, defined as the achievement of concerted action (Goodhue and Thompson 1995), underpins the development and delivery of products and services, and continues to attract attention in current research (Bechky 2003; Hoffer Gittell et al. 2004). Current coordination theory is commonly built on a contingent information processing perspective (Galbraith 1973). According to this approach, differentiation of work translates into task dependencies that are resolved through coordination mechanisms (Crowston 1997), each with varying information processing capacity (Mintzberg 1979). Examples include standards (low information capacity) and mutual adjustment (high information capacity). Matching information processing needs and capacity is required for effective coordination (Van de Ven et al. 1976). While this information-based perspective on coordination has been dominant and useful, its focus on design and tasks (Malone et al. 1999) leaves room for additional theorizing.

Since the mid-1990s organizational economists (Grant 1996; Spender and Grant 1996) and knowledge management scholars (e.g. Nonaka 1994) have worked on a knowledge-based perspective. Here the previous assumption that humans are information processing entities is extended to one in which they are considered as intelligent, learning, reflexive, creative and communicative knowledge workers (Giddens 1991). This shift becomes particularly pressing as organizations tend to become more knowledge-intense and globally distributed. Coordination, from this emerging knowledge angle, is perceived as a problem of sharing, integrating (Grant 1996), creating (Kogut and Zander 1996), transforming (Bechky 2003), and transferring knowledge (Szulanski 1996; von Hippel 1994).

The knowledge-based perspective has received most attention from, first, organizational economists who revised the theory of the firm (Kogut and Zander 1996). These researchers tended to focus on knowledge-based coordination as something that differentiates firms from other governing modes such as the market. Second, knowledge management researchers have paid most attention to knowledge as a dependent variable in their theorizing, leaving coordination as a peripheral spin-off effect. And third, researchers have proposed that transactive memory, mental models, and frames support coordination (Faraj and Sproull 2000; Levesque et al. 2001). However, the mechanics of how these cognitive similarities and linkages lead to coordination remain unclear and therefore deserve more attention for two specific reasons. First, from a micro-organizational point of view, knowledge-based theorizing about coordination remains immature (Brown and Duguid 2001). Second, the emerging knowledge-based perspective (such as information processing) tends to remain focused on mechanisms. These two concerns imply that little is known about how coordination mechanisms impact on knowledge processes and thus enable coordination (i.e.,, a process-oriented perspective). The objective of this paper is to conceptualize this process and thus contribute to a knowledge-based perspective on coordination. Our empirical research concerns a modern organizational form: globally distributed software development projects, which has become increasingly common nowadays.

Research Context: Globally Distributed Software Development Projects

These projects consist of two or more teams working together to accomplish project goals from different geographical locations (Carmel 1999). Difficulties recorded include distance, time-zone, and cultural differences that may include but are not limited to different language, values and traditions, norms, and values of behaviour (Kumar et al. 2005; Walsham 2002).

Most researchers agree that global distribution of knowledge work impacts on coordination practices. A growing number of studies has investigated specific areas and reported problems such as coordination breakdowns (Carmel 1999; Cheng et al. 2004; van Fenema 2002), lack of understanding of a counterpart's context (Orlikowski 2002), and different competencies in language (Sarker and Sahay 2004). Other studies show global work distribution exacerbating the chance of misunderstandings (Battin et al. 2001; Olson and Olson 2004), lack of trust (Jarvenpaa et al. 1998), asymmetry in distribution of information among sites (Carmel 1999), and difficulty in collaborating due to different skills and training, and mismatches in Information Technology (IT) infrastructure (Sarker and Sahay 2004). The practices recommended to overcome these difficulties mainly focus on (i) inter-site coordination through division of work that minimizes cross-site communication and synchronization (Ebert and De Neve 2001; Mockus and Weiss 2001) and (ii) technologies that support collaboration in a distributed environment (Majchrzak et al. 2000; Smith and Blanck 2002).

Studies focusing on global software team performance point to the importance of knowledge sharing in building trust and improving effectiveness, while recognising the additional complexity in sharing knowledge across geographically dispersed sites, not least because of the tacit dimension of so much knowledge (Kotlarsky and Oshri 2005; Orlikowski 2002). Following Wegner (1987), Faraj and Sproull (2000) found that transactive memory is important in globally distributed software teams – instead of sharing specialized knowledge, individuals should focus on knowing where expertise is located and needed. A need to know 'whom to contact about what' in global teams has been reported in several studies in this area (e.g. Kotlarsky and Oshri 2005; Majchrzak and Malhotra 2004; Yoo and Kanawattanachai 2001).

Despite the fact that most of the problems reported in global software projects are fundamentally to do with information and knowledge, overall, past research has stressed the importance of coordination mechanisms and technologies, and has focused much less on the role of knowledge sharing and social aspects in global software projects. One has to go to a different literature, such as organization studies, to work up strong links with such issues. These considerations lead to the need to understand how coordination, knowledge, technical, social, work-based, and organization design factors can relate and form a more comprehensive heuristic and explanatory model. This is accomplished in the next section, grounded in a critique of a more extensive, relevant literature.

Background and Research Model

Coordination Mechanisms

In situations of task interdependence, coordination mechanisms are supposed to help achieve coordination in the sense of some form and level of coherence of individual contributions. Building on various categorizations in past

and current literature (Goodhue and Thompson 1995; Hoffer Gittell et al. 2004), we regroup the mechanisms into four categories: *organization design mechanisms, work-based mechanisms, technology-based mechanisms,* and *social mechanisms* (the external circle in Figure 1).

The selection and use of these mechanisms depends on contingencies such as diversity, work unit size, and task uncertainty (Van de Ven et al. 1976; van Fenema 2002). While connections may exist between the categories, current literature follows the information processing tradition and generally considers the mechanisms as substitutable alternatives (Mintzberg 1979). As an extension of the current literature, we make explicit how each category of mechanisms contribute to coordination. This facilitates the transition towards a knowledge-based perspective on the relationship between coordination mechanisms, knowledge processes, and achieving a coordinated outcome.

First, *organization design mechanisms* encompass formal role structures such as hierarchies, linking pins, teams, and direct contacts (Galbraith 1973; Hoffer Gittell et al. 2004). These structures can be considered 'mental traces' that are enacted and modified in practice (Orlikowski 1992). The objective of organization design is to accomplish the integration of differentiated tasks. Professionals and units with unique tasks must work together to integrate knowledge and achieve a common output (Grant 1996). How their accomplishments are linked at a general level is determined by the role to which people and units are assigned, and how these roles are linked. Information processing theorists suggest that direct forms of design (direct contacts, teams) have higher information processing capacity than indirect forms (hierarchy, liaisons) (Galbraith 1973). The former is more suited to complex, uncertain tasks that may involve diversely skilled professionals.

Second, *work-based mechanisms* concern the specific structuring of tasks to be accomplished by an organization. Examples include plans, specifications, standards, categorization systems (Bowker and Starr 1999), and representations of work-in-progress, such as prototypes (von Hippel 1994) and design documents. Research suggests that people tend to rely more on work-based practices if tasks are complex (e.g. discussing a complex prototype of a new car), if communication opportunities are limited (e.g. remote communications), and if achieving common understanding is highly important.

Third, *technology-based mechanisms* are defined here in terms of IS (Hoffer Gittell et al. 2004). These support coordination by enabling information capturing, processing, storage, and exchange (e.g. electronic media, groupware, shared databases). As coordination practice, technologies automate and informate organizations (Zuboff 1988). On one hand, automating implies that technology replaces humans for accomplishing coordinating tasks (e.g. a traffic light system). In the project management this may include automated scheduling, automated file version control, and automated notification when tasks are finished. Informating, on the other hand, relies more on human involvement in the coordination process. In this role, technologies enable people to communicate asynchronously and possibly remotely.

And fourth, *social (inter-personal) mechanisms* relate to human functioning in situations of interdependence (Hoffer Gittell et al. 2004). Human engagement in coordination practices involves communication activities, working relationships, and social cognition: (1) Communication has been traditionally recognized as a mode for adaptive coordination (Van de Ven et al. 1976). When people encounter novel circumstances or counterparts, they must communicate in order to make sense and establish a shared understanding (Donnellon et al. 1986). (2) Working relationships enhance the accuracy of expectations concerning a counterpart's thoughts, activities, and expectations. This promotes coordination and communication efficiency (Gabarro 1990). And (3) social cognition refers to the frames and mental models people have in common because of similar experiences (Cramton 2001; Krauss and Fussell 1990). The coherence of individuals' functioning becomes more likely when these social practices occur. Organizations select and deploy them particularly when they work on novel or tightly linked tasks, or when people from different functional areas must cooperate (Dougherty 1992).

A Knowledge-Based Perspective on Achieving Coordination

When coordination is considered from a knowledge point of view, the four mechanisms discussed above gain further depth in the sense of their role in knowledge processes. Key questions thus become, firstly, *how do these mechanisms contribute to knowledge processes?* and, secondly, *what must happen to knowledge in order to achieve coordination?* Organizations must do something with their (potential) knowledge resources in order to coordinate activities that are performed in different time-space configurations and across a variety of units, teams, and communities (Brown and Duguid 2001). If they do not, their performance suffers from knowledge asymmetries, knowledge that remains 'stuck' to particular sites (von Hippel 1994), and unrealized potential of knowledge creation

and collective creativity (Grant 1996; Kogut and Zander 1996). Coordination mechanisms thus become knowledge management instruments, with a focus on their contribution to the coherence of knowledge processes and activities, i.e.,, achieving a coordinated outcome. As shown in our research model (Figure 1), we conceptualize for each category of coordination mechanisms their impact on knowledge processes and thus ultimately on a coordinated outcome.

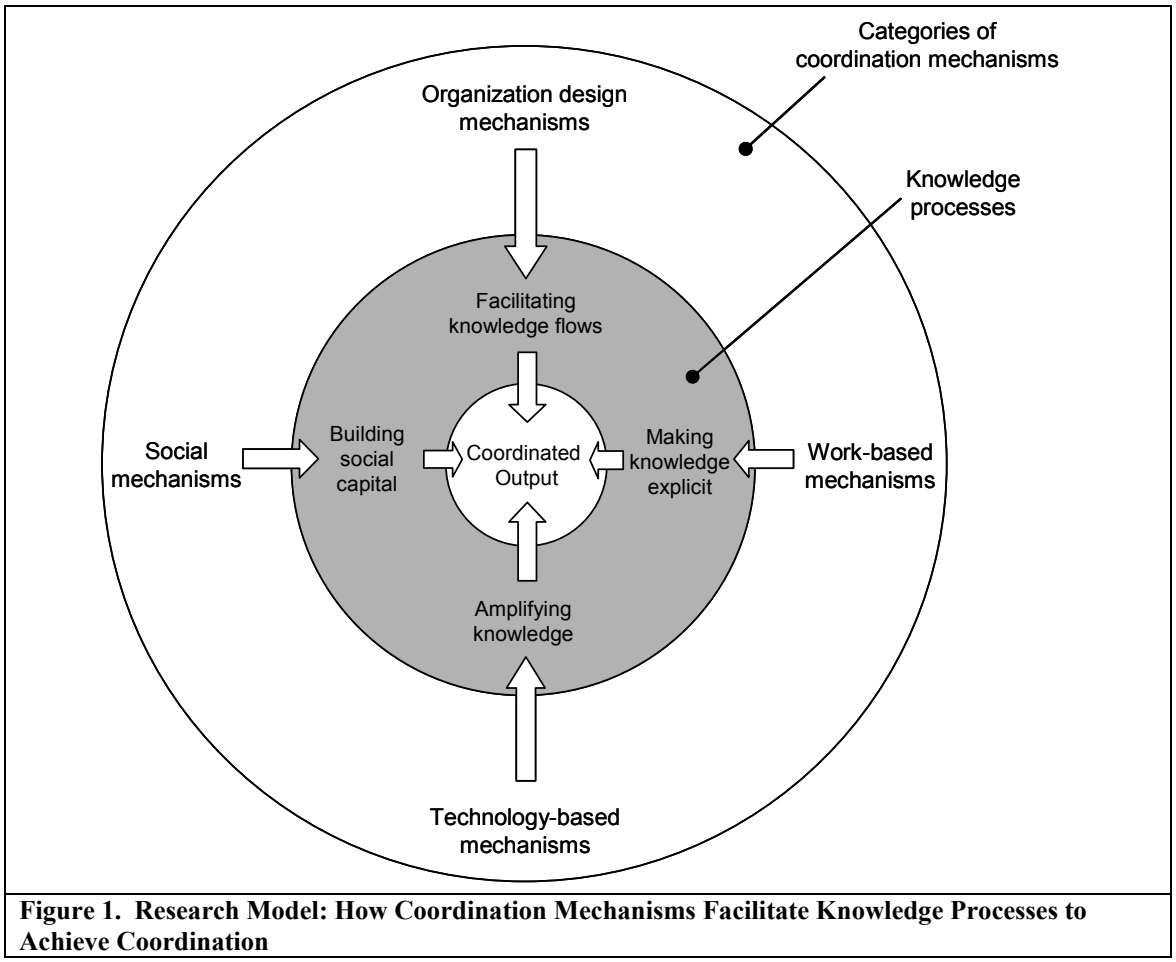

Figure 1. Research Model: How Coordination Mechanisms Facilitate Knowledge Processes to Achieve Coordination

First, organization design mechanisms *facilitate knowledge flows* by providing a structure through which knowledge workers can channel their expertise. To achieve coordination, knowledge must flow, be connected, and different perspectives must be confronted (Boland and Tenkasi 1995). Organization design clarifies who is supposed to know what and who is supposed to communicate with whom. It therefore economizes knowledge flows.

Second, work-based mechanisms capture knowledge are important for *making knowledge explicit*, as they enable activity replication and commonality (Adler 1995). Hence, we reconsider the explicitation-internalization cycles proposed in Nonaka et al.'s SECI (Socialization, Externalization, Combination, Internalization) model in the light of achieving coordination. The use of work-based mechanisms implies that knowledge and expectations are made explicit and thus are known and useful to other people working at different sites or at different times (i.e.,, with limited communication opportunities).

Third, in dispersed organizations, knowledge must be rapidly disseminated by means of technology-based mechanisms. Knowledge-intense multinationals and service firms *amplify their knowledge management processes* using intranets, knowledge databases, and groupware (Ciborra et al. 1996; Majchrzak et al. 2000). While IS processes data and information, to knowledge workers within the same community and organization these constitute pieces of knowledge that trigger new thoughts and enable coordinated action (Ngwenyama and Lee 1997).

And fourth, social mechanisms *establish social capital* in the sense of relationships (Gabarro 1990), and of knowledge of who knows and does what, i.e.,, transactive memory (Faraj and Sproull 2000; Moreland 1999). Individuals are not just information processors, but knowledge workers who negotiate points of view (Brown and

Duguid 2001), and transform their understandings (Bechky 2003) to generate innovative outputs. In this context, they have relational needs that are relevant for coordinating their work.

Our research framework presented in Figure 1 suggests that four types of coordination mechanisms impact on knowledge processes. By considering this knowledge dimension of the mechanisms, we can better understand how coordination is achieved. The model, grounded in the predominantly mechanisms-based literature on coordination, attempts to extend this approach to a dynamic, knowledge-based perspective on coordination. Next, we examine how this perspective could improve our understanding of achieving coordination in practice.

Research Design and Methods

Design and Case Selection

In line with much past research (Eisenhardt 1989; Yin 1994), a case study method was selected for this research. An in-depth case study of globally distributed software development projects is provided. A qualitative, interpretive approach is adopted.

Selection of case studies was driven by the main research question: *how do coordination mechanisms contribute to knowledge processes so that coordination is achieved?* To analyze the role of coordination mechanisms in supporting knowledge processes we compare two cases, one of which is successful (i.e.,, a coordinated outcome is achieved) and another one unsuccessful (i.e.,, failed to achieve a coordinated outcome). The nature of success and failure is therefore assessed based on whether an organization succeeded or failed producing a coordinated outcome through the use of (one or more) coordination mechanisms (Nidumolu 1995; Nidumolu 1996). Based on this criterion, we selected one successful project at SAP and one project that failed at Baan. By comparing coordination mechanisms used to facilitate knowledge processes in the successful and unsuccessful cases, we could draw conclusions about the role of knowledge processes in achieving coordination.

Data Collection

Evidence was collected from interviews, documentation and observation, as suggested by Yin (1994) and Eisenhardt (1989). Interviews were conducted at two remote sites per company: in India and Germany for SAP; in India and The Netherlands (NL) for Baan. Interviewees were chosen to include (1) counterparts working closely at remote locations, and (2) diverse roles such as managers and developers. In total, 19 interviews in two companies were conducted. Interviews lasted on average 1.5 hours; they were recorded and fully transcribed. A semi-structured interview protocol was applied, to allow the researchers to clarify specific issues and follow up with questions.

Data Analysis

Data analysis followed several steps. It relied on iterative reading of the data using open-coding techniques (Strauss and Corbin 1998), and sorting and refining themes emerging from the data with some degree of diversity (Miles and Huberman 1994). In particular, four themes that represent the concept *coordination* were carefully studied: *coordination by organization design*, *work-based coordination, technology-based coordination,* and *social coordination*. Statements that were found to correspond with mechanisms that support the these four types of coordination were selected, coded and analysed using Atlas.ti - Qualitative Data Analysis software (Miles and Huberman 1994).

The first step involved reading through the interview transcripts and collected documents and (i) creating a list of coordination mechanisms that were employed or lacking in the projects under study, and (ii) marking evidence of the existence or lack of knowledge processes, according to the four types of knowledge processes: *designing knowledge flows*, *amplifying knowledge management processes*, *making knowledge explicit,* and *building social capital*. During this stage chunks of text (paragraphs or sentences) (Strauss and Corbin 1998) describing (i) coordination mechanisms and (ii) evidence or lack of knowledge processes were coded.

Next, statements (i.e., codes) illustrating coordination mechanisms were grouped into the four above mentioned categories that represent four types of coordination mechanisms.

Finally, we analyzed statements in each of these four categories to identify the impact of coordination mechanisms on knowledge processes and, through knowledge processes, on achieving a coordinated outcome. At this stage statements grouped into each of the four categories were analyzed in the light of knowledge processes that were

supported by the coordination mechanisms (or those lacking as a result of not having appropriate coordination mechanisms in place).

Analysis and Results

In this section the results of two case studies carried out at SAP and Baan are presented. Based on the empirical evidence presented below, we explore how the use of coordination mechanisms facilitates knowledge processes between globally dispersed teams and enables remote counterparts to share and integrate their knowledge, which results in achieving coordination when they work towards knowledge-intense innovative outcomes. In order to support the above claim, empirical evidence illustrated by statements made by interviewees will be outlined in the following section. A presentation of the cases follows the analysis process described above and aims to illustrate the knowledge dimension of the coordination mechanisms (i.e., how these impact on knowledge processes and on achieving a coordinated outcome).

SAP Case

The Project under Study

This case study focuses on the SAP Collaboration tools project developed by the Knowledge Management (KM) Collaboration group, which is part of the Enterprise Portal Division. The goal of the SAP Collaboration tools project was to develop a comprehensive collaborative platform that would enable both individuals and teams in different locations to communicate in real-time and asynchronously, and to support the teamwork of any distributed project teams. The SAP Collaboration tools were developed to be part of the next generation application and integration platform (that is, SAP NetWeaver), and to allow integration with various tools of different providers.

The development of SAP Collaboration tools started in September 2001. By June 2002, the first version of SAP Collaboration tools was released and the group was working on the second release.

Software Team

The KM Collaboration group, where the case study was conducted, is part of SAP Portal. From a geographical perspective, the software team was distributed between three locations and consisted of four teams: two teams in Walldorf, Germany (ten people in each team), one team in Bangalore, India (six people) and one team in Palo Alto, USA (five people). Each team worked on a different part of the Collaboration tools (see Figure 2).

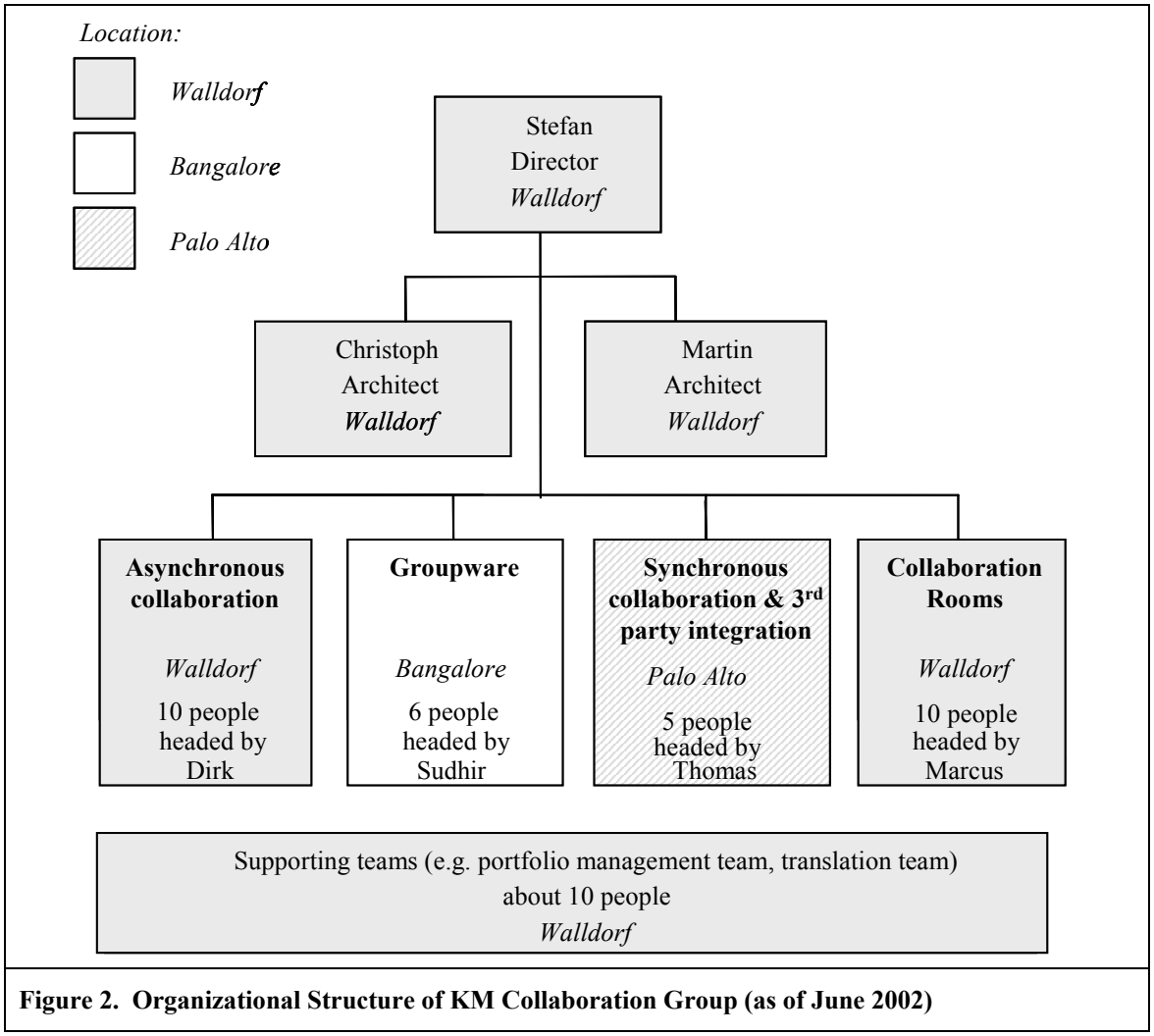

Figure 2. Organizational Structure of KM Collaboration Group (as of June 2002)

The development managers of each team report directly to Stefan, who is the director of the group. Two development architects, Christoph and Martin, work on the conceptual design of the architecture. Their responsibility is to drive the architectural design and ensure that everything fits together.

SAP Case: Analysis

In September 2001, when the Collaborative tools project started, key players (managers and architects) and team members from remote locations did not know each other. Some of the team members had previous experience of working in a globally distributed environment, but not necessarily with Indian, German, or American cultures. For the majority of key players and team members this cross-cultural setting was new. Furthermore, at the beginning of the project, there was a knowledge gap between individuals involved in the project:

> People have different profiles: here [in Bangalore], the maximum experience is 5 years. But if you take these three colleagues travelling to the team-building exercise [Stefan, Christoph and Thomas], the two of them have about 12-15 years of experience, and the minim um experience here [in Bangalore] is about 2½ years, so that's a huge experience gap that they have to bridge (Sudhir).

From the very beginning managers of the KM Collaboration group realized the importance of sharing and coordination of knowledge across dispersed locations and put a lot of effort in setting up and facilitating the knowledge process.

In this section four types of coordination practices employed to facilitate knowledge processes between globally dispersed team members are discussed and illustrated using quotations made by interviewees.

Coordination by Organization Design

The organization design of KM Collaboration group aimed to facilitate knowledge flows in order to reduce existing knowledge gaps and prevent knowledge and information gaps in the future. In particular, a clear division of technical versus 'social' supervision (i.e., management of local teams) between the technical architects located in Walldorf and the local development manager aimed to ensure the quality of the product and effective team management. The local development manager of each team was responsible for team management: he divided specific assignments (tasks) between team members and resolved social issues. The development manager and team members belonged to the same culture. This made it easier for the development manager to understand and deal with the team members. Furthermore, mini-teams were created and reporting channels across the globe were established. For example, Christoph and Martin (development architects located in Walldorf) served as technical contact persons for the remote teams: Christoph was a contact person for the Bangalore team (Cristoph and the Bangalore team constituted one mini-team), and Martin was a contact person for the Palo Alto team (another mini-team). The architects provided technical supervision for the assigned remote team, and were responsible for technical issues and the quality of software developed by this team. Creating cross-continental mini-teams was helpful in shaping communication patterns, providing clarity and, as a result, facilitating knowledge-sharing processes between the Head Office in Walldorf and remote sites.

Moreover, direct communications were encouraged in the KM Collaboration group to facilitate knowledge sharing. After the key players visited the Bangalore site and got to know remote team members personally, centralized communications (via Sudhir) were replaced by direct communications. Christoph explained:

> From a code perspective, what I did before I met all of them [the team in Bangalore] in person was to send all things to Sudhir and he was the one to distribute it within the team, and this has changed now. I address most of the things directly to the team members […]. Quick and direct communications as far as possible, is the most important thing. 'Direct' means: do not communicate through other people but with the people directly. If you have one contact person who distributes all the information, you lose some amount of information, just because you do not reach the right people.

Work-based Coordination

Work-based coordination aimed to capture knowledge and make it explicit and accessible for all team members despite their geographical location. It was achieved through division of work and creating common knowledge about software development tools and procedures.

First, work was divided feature-wise, providing dispersed teams with full ownership of and responsibility for an entire block of functionality: 'you are responsible for what you have taken up' (Stefan). This approach aimed to reduce knowledge dependencies in the newly formed global team, reducing possibilities for misunderstandings and conflicts. Moreover, it was important, in particular for offshore teams, to have full ownership of their work. It gave them a feeling of being valuable and the motivation to collaborate and share knowledge in the future.

Second, to ensure consistency in the methods and tools used by dispersed teams and to facilitate common understanding of the evolving product, the managers of KM Collaboration group decided to standardize tools and methods across dispersed locations:

> We use all the same tools, so there is no difference. We even use the same Word templates [templates with project activities and related documents], so even the specifications look more or less the same (Christoph).

A sharing of knowledge being embedded into the standards facilitates coordination across dispersed locations, as people from remote locations perform interrelated tasks coherently.

Technology-based Coordination

A variety of technologies were used to communicate, coordinate and share knowledge over distance, amplifying the knowledge sharing of the SAP global team. Technologies enabled remote team members to share explicit knowledge resources and increased the speed and flexibility of knowledge sharing by making it independent of

place and time (remote/asynchronous collaboration). Therefore, technologies were very important for the integration of knowledge across dispersed locations: for achieving a common understanding between remote counterparts, ensuring consistency in the evolving product, and coordinating of tasks between teams and individual team members.

Furthermore, technologies were used to facilitate the reuse of knowledge and software components across locations, which could reduce time-to-market of new product versions. For example, videoconferences (VC) that involved members from all remote teams were used to identify opportunities for reuse:

> The team in Walldorf should be aware of what is being developed in Bangalore or Palo Alto, so that we don't reinvent the wheel again and again. So we communicate about things that are being done, and is there something reusable which we are developing, or have they developed something which somebody else can use. Then you are not rewriting the whole product again and again. Maybe they can just use our package available, make some changes according to what they need, and use it. For things like that we need to interact with each other (Akhilesh).

The quote above illustrates how technologies can increase the efficiency of knowledge processes (by avoiding re-inventing the wheel).

Moreover, Internet and Web-technologies enabled the centralization of technologies under a single environment accessible from all remote locations, was important to ensure that everybody was working with the same, most updated versions. For example, SAP Intranet (called SAPNet) served as a central place with links to all updated information. This way, technologies allowed remote counterparts to keep updating their knowledge about what is going on; the plans and progress.

A variety of collaborative technologies were used in different situations. A phone was used for urgent matters, for regular updates between managers, and to resolve misunderstandings. For situations that required knowledge sharing between remote counterparts Application Sharing Tool (AST) or VC were used. For example, typically an ACT was used remotely (i) for discussions that involved showing slides (usually, in such situations remote counterparts use AST to show presentation, and simultaneously they use the phone to explain the slides and to discuss issues) and (ii) for discussing technical issues (e.g. code reviews, debugging); in this case the AST was used for taking control of a computer remotely.

Twice a month VC sessions that involved managers and developers from all three locations were organized to discuss progress and other issues, for example:

> Whenever a new colleague joins our team or any of the teams in the other locations, in the next VC we will have an introduction round like 'these are new colleagues that have joined'. So though you have not met them physically, you get to know that this is the person, he exists there, things like that (Akhilesh).

This way counterparts from dispersed locations got to know the composition of a remote team and could know whom to contact. This helped to streamline knowledge sharing between remote counterparts.

Finally, email was typically used for low priority tasks and issues, and tasks that could not be completed in real-time because of time-zone differences.

Social Coordination

Social coordination mechanisms aimed to create the social capital of the global team. A particular effort was put into building up shared experiences, and creating transactive memory among dispersed team members. Social mechanisms included team-building activities and mutual adjustment which aimed to reduce knowledge gaps, build relationships and maintain team atmosphere between dispersed team members. Furthermore, frequent interactions and systematic communications between remote counterparts were considered important to ensure effective coordination over distance.

The remote counterparts did not have a history of working together before they were merged into one group. Therefore transactive memory in this group had been developing since the project started. Having transactive memory was important as it influenced the amount of information that needed to be shared, and had an impact on the efficiency of communications, as illustrated by the following quote:

> A simple one-line question can result in a 10-page answer. It can be a very lengthy answer a one-line reply. The level of detail you get in the answer depends on how well you know that person. Because if the person knows me very well and knows in what areas I am working, then he can decide how much information I will need. Is one line good enough for him or should I explain to him over three pages so that he knows what is happening? (Sudhir).

To bridge the knowledge gap and facilitate knowledge sharing between the teams in the early stages of the project Sudhir (manager of the Bangalore team) organized a team-building exercise in Bangalore in which key members of Waldorf and Palo Alto teams participated, together with the local team. The team-building exercise gave an opportunity for key members to meet in person, learn about areas of expertise of remote counterparts and their working experience, learn about cultural differences and create space for social interactions. This exercise helped to reduce the possibility of conflicts and misunderstandings in the future:

> The team-building exercise improved relationships among the KM Collaboration group, because earlier communications were only in a formal way, and after the team-building activity we really knew people much better, it became easier to communicate and communications became more informal (Jyothi).

This way, the team-building exercise was used to promote knowledge sharing processes. It was the first major step towards bridging knowledge gaps between dispersed team members and towards developing trust:

> The end result of that exercise was that the entire team feels more comfortable to work together. Now they know each other and trust each other better (Stefan).

Mutual adjustment included setting up rules of communications which helped people to adjust to communication styles and reduce misunderstandings and confusions that typically happen as a result of different cultural backgrounds. For example, agreement was reached that Indian team members would not take it personally when Germans are too direct. Compared to Indians, Germans usually are very direct and 'brutally precise' in communicating what they have in mind and, typically, this is one of the biggest challenges in German-Indian teams.

Facilitating interactions between remote counterparts included facilitating personal face-to-face interactions and organizing frequent interactions over distance. For example, regular teleconferences between software managers in Walldorf, Bangalore and Palo Alto, and transatlantic VCs with all team members every couple of months helped to keep knowledge of all parties up to date.

Baan Case

The Project under Study

This case study focuses on the development of an E-Enterprise Suite. The case study was conducted in early 2002, when two globally distributed locations, Hyderabad (India) and Barneveld (NL), were involved in developing the E-Enterprise Suite. The E-Enterprise Suite was designed to let users extend their Baan manufacturing, financial, and distribution software on the Web to allow them to collaborate better with customers, suppliers, and partners. In March 2002 the E-Enterprise Suite consisted of seven products that were all based on one platform called E-Enterprise Server. Products included in the E-Enterprise Suite were developed to be stand-alone as well as to be integrated with the ERP package developed by Baan.

Software Team

Development of the E-Enterprise suite was organized by feature/product function. From a geographical perspective, the E-Enterprise group was distributed between two locations: Hyderabad (about 60 people working on five products of the E-Enterprise Suite) and Barneveld (about 35 people working on two products and the common platform of the E-Enterprise Suite) (see Figure 3).

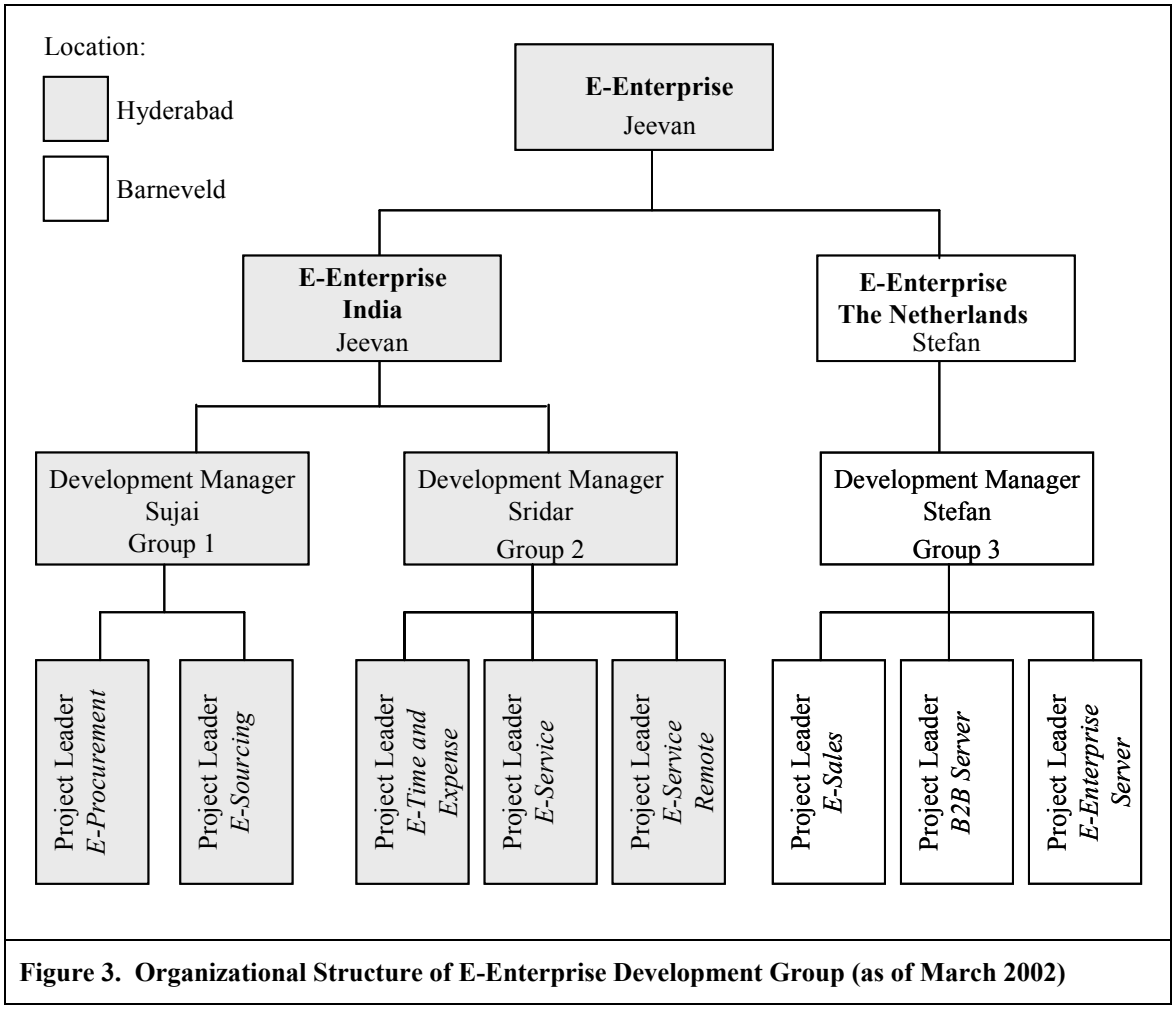

Figure 3. Organizational Structure of E-Enterprise Development Group (as of March 2002)

In addition to the E-Enterprise group, Marketing&Alliances group and the Project&Process office were involved in the management of the E-Enterprise suite,.

Baan Case: Analysis

The E-Enterprise group was relatively young: the first E-Enterprise products were released in 1999. Some people in Hyderabad had been working in a globally distributed environment before joining the E-Enterprise group. However, because of a general Baan policy to reduce travel expenses, and because the E-Enterprise organization structure had changed several times since the group was established, team members did not have a history of working together: the majority of them did not know each other and did not know the composition of the dispersed team. Therefore, in Baan transactive memory among dispersed team members was not developed.

Furthermore, team members in NL and India had different cultural backgrounds in terms of national culture (Dutch and Indian) and organizational culture (newcomers and people from Baan ERP group), and did not have a common technical background. Therefore, there was a gap in common understanding of the technology and the processes team members were supposed to follow. Moreover, it was reported that often people in the Hyderabad office were not aware of what was happening in the Barneveld office: they were not updated about changes in requirements and dependencies between the products, and not aware of product and technology roadmaps.

In this section four types of coordination practices that were (or were not) in place in the E-Enterprise group are presented. The implications of (the lack of) coordination practices for knowledge processes between globally dispersed team members are discussed and illustrated using quotations made by interviewees.

Coordination by Organization Design

The organization design of the E-Enterprise group was continuously changing: people, their roles, products, product requirements, processes, ownership, and physical location of tasks – all was changing very fast. Moreover, too many people in different roles were involved in the management of each product included in the E-Enterprise Suite, so that some responsibilities were overlapping. Combined with other circumstances (e.g. the sequentially changing ownership discussed below under 'Work-based coordination'), a situation emerged where everybody was involved but nobody was responsible. Interviewees (in different roles) were asked to list the people involved in the management of different E-Enterprise products and to describe their roles and responsibilities. From the descriptions provided by the interviewees it followed that sometimes people had different views on what they or their colleagues were supposed to do.

Lack of stable organization design and clearly defined roles created flaws in knowledge flows between globally distributed teams: some information was lost because there were no clearly defined communication channels, team members had very limited knowledge about their remote counterparts and often did not know whom to contact at remote site if issues emerged.

Work-based Coordination

Work-based coordination in the E-Enterprise group was very limited. First, there was no clear division of work between the Indian and Dutch teams. For example, between 1999 (when the first version of E-Enterprise Suite was developed) and 2002 (when the interviews were curried out) ownership of the common platform – E-Enterprise Server – was transferred from India to NL and then back to India.

Changing ownership had implications for knowledge processes: there was always a need to understand the product developed by another team (which is often more difficult than to develop a product from scratch), and there was never a complete knowledge of the product and the logic behind it. For example, as Sujai explained:

> It's difficult to visualise the idea when it is not yours. If we have the knowledge of the existing product then we're building on top of it, it's easy. But sometimes it happens that the understanding of the existing architecture is not very good because we are not there from the beginning: the initial product has been transferred from there [NL] to here [India].

Furthermore, there was no feeling of 'our' product, because the product was inherited from another team: 'I expect one of the important things that should happen within E-Enterprise Baan or anywhere is that more ownership must be felt by everybody' (Vijaya). Everything seemed to be in transition and unstable. This situation reduced morale in the E-Enterprise group and increased tensions between Indian and Dutch group members. This tense relationship, in turn, reduced motivation of remote team members to share knowledge.

Moreover, there was a strong technical dependency between the E-Enterprise Server and the seven products comprising the E-Enterprise Suite. These dependencies existed because the combinations of products had to work together. Technical dependencies on the E-Enterprise Server caused two problems: (1) specifications and (2) schedules across products needed to be synchronized. Technical dependencies on the E-Enterprise Server caused knowledge and information dependencies between the dispersed teams. Vijaya explained:

> The dependency on NL is causing problems. Dependency on information, dependency on knowledge (even in terms of simple design documents, for example, functional designs, or technical designs, they are not complete), dependency on requirements, because everything is centralised in Holland and then that has to be shared with us so that we can proceed.

Taking into account the numerous dependencies discussed above, there was no structured approach to identify and coordinate these dependencies:

> One thing that is missing right now in E-Enterprise is that at any time you can't look into any document to see what are the exact dependencies involved. Right now they're coming with something like a dependency matrix. But so far we didn't have that. So it's generally like if you want to know tomorrow

> whatever dependency with another product, you have to actually talk to the team members or the Architect or the Consultant. There is no central store or central repository (Satish).

However, it is important to note that there were some attempts in the E-Enterprise group to establish work-based coordination. For example, Baan tried to standardize development methods and processes:

> We want to have common processes across the locations. We try to achieve a uniform standard for all these. So that is a basic aim of this. Though we have not reached it in all the areas, but in certain areas we are making steps (Jeevan).

As work-based coordination in the E-Enterprise group was very limited, the remote team members suffered from information flaws and lack of knowledge sharing. In particular, the lack of common knowledge about the evolving product was mentioned as critical:

> We have an existing architecture and we need to build future products based on this architecture, so understanding the existing architecture is most important to be able to build on top of it. We have completed realisation from our side [E-Procurement and E-Sourcing] and E-Enterprise Server has also completed their realisation. But now we need to integrate E-Enterprise Server into our applications. So for that we need a lot of knowledge about E-Enterprise Server (Sujai).

Technology-based Coordination

The E-Enterprise group was well equipped with the technologies required to enable working in a globally distributed environment. Technologies were considered very important: 'this is actually one of the most important things: technology comes to our rescue in working in a distributed environment' (Venkat). Different technologies were used to save on travel costs between NL and India, as Venkat explained:

> Quite some time back, before all of these tools came into practice, we used to travel to NL and they used to travel here in order to meet us, especially at the start of a new release or to share some important needs that stretch over a long time. Even for small purposes people used to travel. That was becoming expensive and they [Baan] had to think of alternatives, then all of these media came in the picture. Then the VC was immediately applied. We started using VC, and we don't have to go to NL: we are saving a lot of dollars.

A variety of collaborative technologies were used in different situations. Typically email was used for quick queries and describing a problem prior to a phone-call. Phone was used in situations when an urgent response was required and to resolve conflicting situations:

> Telephone usually involved when a lot of emails have exchanged and certainly we feel that everyone is talking differently and it is taking too much time and no one is coming to any conclusions, then we start organising a telephone call (Srinnivas).

Overall, there was also a tendency to minimise the use of the phone because of the costs involved.

ASTs – in particular NetMeeting and Webex – were often used for knowledge sharing activities during meetings between sites and with customers.

VC was used occasionally for updates between managers from dispersed locations.

In the E-Enterprise group collaborative tools were mainly used for coordination and knowledge sharing purposes between individual team members. However, there was no configuration management tool in place to coordinate technical dependencies between products included in the E-Enterprise Suite (problems caused by lack of compatibility between versions of different products, discussed under 'Work-based coordination' earlier).

Social Coordination

In the E-Enterprise group social coordination between dispersed team members was very limited. Visiting the other country was difficult for people involved in the E-Enterprise group because Baan was 'trying to make cost-cutting measures, and they tried to shift everything to one location to reduce the communication costs' (Sridar). Therefore, the majority of team members had never been to the remote location and did not know who their remote counterparts were.

Lack of social coordination caused a number of problems. In particular, there was a lack of team atmosphere between teams in Hyderabad and Barneveld: from interviews of members of both teams, tensions between the teams became evident:

> The major issue is that people don't perceive that on the other side, **they**'re not reciprocating **our** needs: what **we** want, during which time, what priority **we** have. **They** don't see the same priority as **our** people see, and vice versa. So there is always a gap (Jeevan).

Interviewees were convinced that meeting their remote counterparts would give them a possibility to develop rapport, learn about cultural differences and share views on different issues. This would help to bridge cultural and knowledge gaps:

> Personally I feel meeting the people would help you resolve the tasks more quickly, because you can really think and feel the person when you are actually talking. For example, assume two people, one has never come to India and the other has never gone to Holland. If they are interacting, there would be some gaps. But if they had an interaction at a personal level at some point in time, then the interaction would really be better, the response will be generally quicker (Phani).

It was mentioned that understanding of cultural differences could help to bridge knowledge gaps and improve working relationships. For example, Ganesh (Process Manager for Baan Hyderabad) explained that understanding of cultural differences helps to define better processes that would be acceptable for Dutch and Indian cultures:

> When we write the process plan there are a lot of cultural issues that come into the picture. How to deal with this particular area? I can give you an example on quality assurance - a critical area. In the Indian culture, quality assurance is an important topic - people don't mind someone checking the work they do, but if you compare with our counterpart: in NL sometimes people don't like this. Because the counterpart, the NL team have a different culture - individualistic. So there will be some resistance on that front sometimes. Once we understand this and appreciate the cultural factors, then we can define better plan.

Comparing the Cases

In this section we compare results from the successful and unsuccessful cases. As an initial step, we compare the existence or absence of coordination mechanisms. Next, we assess how the role of coordination mechanisms in these organizations impacted on knowledge processes and ultimately coordination outcomes.

Coordination Mechanisms at SAP and Baan

In contrast to the E-Enterprise group of Baan, managers of the SAP KM Collaboration group implemented a number of coordination practices which aimed to facilitate knowledge processes between dispersed teams. Table I lists the coordination mechanisms identified in the SAP case, grouped according to the four categories of coordination mechanisms. The sign '+' is used to illustrate existence of a coordination mechanism, '-' indicates lack of a mechanism.

Table 1. Coordination Mechanisms: Comparison of Results across Cases		
Coordination mechanisms	SAP case (successful)	Baan case (unsuccessful)
Coordination by organization design		
Contact person / liaison	+	-
Mini-teams	+	-
Direct contact	+	-
Work-based coordination		
Enabling flexible PM techniques, planning by milestones	+	-
Making efficient division of work	+	-
Using specifications to guide the work	+	-
Using standard tools, SOP and methodologies	+	+
Technology-based coordination		
Shared Software Development tools	+	+
Internet enabled ICT infrastructure	+	+
Wide range of media and collaborative technology	+	+
Shared databases	+	-
Technology-enabled representation / visibility	+	-
Social coordination		
Team-building	+	-
Mutual adjustment	+	-
Facilitating interactions	+	-
Designing systematic communications	+	-

Coordination Mechanisms and Knowledge Processes

Organization design. At SAP, a number of coordination mechanisms were employed to facilitate sharing and integration of knowledge between the dispersed teams. Various forms of organization design enabled handling knowledge processes differently. For example, direct contacts were used to promote unbiased and efficient knowledge transfer. This established and maintained transactive memory (who knows what and who is doing what) and enabled knowledge integration. By contrast, at Baan, organization design did not define clear communication channels to support knowledge and information flows between the teams in NL and India. This caused breakdowns in the coordination of work done at remote sites. Often, people in the Hyderabad office were not aware of plans and changes in products and technology originated by the Barneveld office. This limited knowledge processes between dispersed teams which, in turn, did not help the teams to achieve a coordinated outcome.

Work-based coordination. Both companies, SAP and Baan, used project plans and product specifications to coordinate work between locations. However, the companies used these coordination mechanisms differently. For example, in SAP (i) updated specifications were available on the intranet, and (ii) remote teams were informed about any modifications through their contact persons (technical architects). Through these mechanisms SAP ensured that the new knowledge is captured, and made explicit and accessible for all team members despite their geographical

location. However, in Baan there was no central point of access to updated documents and, often, employees used outdated specifications to design their products. Both companies put efforts into standardization of tools and methods across locations to reduce knowledge gaps and create common knowledge about these tools and procedures. At the time of data collection Baan was implementing standard development methods and processes across locations. SAP standard tools and methods were implemented at the very early stages of the project, which helped to facilitate knowledge processes between remote teams through shared (similar) understanding of standards and made easier integration of deliverables of remote teams into the single (joint) outcome.

Technology-based coordination. Various technologies deployed at SAP helped to amplify knowledge processes across dispersed locations: Web access and synchronization of data ensured that all teams had access to the latest files. At Baan there was an attempt to have a central requirements database. However, requirements were changing so quickly that the database was not up to date. Code was synchronized via the synchronization of databases at two locations. At SAP, various collaborative technologies were available for dispersed team members: for example, internal phone lines (a 5 digit number) between Bangalore and Walldorf made it easy to contact remote counterparts. These collaborative technologies were used in a *proactive* manner, for example, for updates and knowledge sharing during various meetings between managers and developers, organized on a regular basis. In Baan, team members at both locations had a variety of collaborative technologies as well. However, different from SAP, at Baan these collaborative technologies were used largely in a *reactive* manner to fill in knowledge and information gaps; for instance, for clarifications and to resolve problems.

Social coordination. At SAP, social coordination played an important role in building social capital at a global team level. As three dispersed teams were merged into one group in the beginning of the project, members of these teams had to build relationships such as trust and rapport from scratch. The team-building exercise and short visits were organized to give developers and key players an opportunity to meet in person in an informal environment and get to know each other. This helped to create transactive memory and build relationships among the team members. By contrast, Baan did not have coordination mechanisms in place aimed at building social capital between dispersed team members. Furthermore, many of the people interviewed did not know their remote counterparts in person. Baan tried to reduce project costs by avoiding traveling, thus reducing the opportunity of remote team members to meet in person. As a result, there were tensions between team members in Hyderabad and Barneveld. Tensions reduced the motivation of remote team members to communicate and share knowledge, unless this was considered absolutely necessary. Therefore, communications between the teams were often postponed to the integration stages and were initiated by emerging problems.

Knowledge Processes and Coordinated Outcomes

In SAP, the managers of the KM Collaboration group realized the importance of the sharing and integration of knowledge across dispersed locations at the early stages of the project and put a lot of effort into setting up and facilitating knowledge processes across dispersed locations. It was particularly important to create transactive memory between dispersed team members, to share knowledge about the cultures of the remote counterparts, and to achieve a common understanding of the evolving product. Interviewees from SAP said that knowing who knows what at a remote location enabled them to reduce development lifecycle because team members knew whom to contact for a specific problem and the response was quicker. As a result, they were able to achieve a successful coordinated outcome:

> We just went through a merger, so setting up a global project was not an easy task. Despite all the difficulties we managed to have a successful second software release in 8 months (Stefan).

The success of the project outcome is also supported by external evidence. According to JupiterResearch, a leading research and consulting company in emerging technologies, SAP Enterprise Portal was recognized as the third largest software solution, with 17% of the USA market in 2002. The studied project developed SAP Collaboration tools as one of the main features of the SAP Enterprise Portal.

At Baan, managers of the E-Enterprise group had limited coordination mechanisms in place that would facilitate knowledge processes. Subsequently, there were numerous knowledge gaps between team members in India and NL. Moreover, the E-Enterprise global team did not develop transactive memory as team members did not know who their remote counterparts were. Often interdependencies between products developed at remote locations were discovered at the last moment and this resulted in integration problems (instead of careful advance planning). As a result, working in a globally distributed environment proved to be problematic for the E-Enterprise group, and in summer 2002 the development office in NL was shut down.

Conclusions

This paper has developed a knowledge-based perspective on coordination and demonstrated its applicability in the context of globally distributed software projects. The increasing move toward globally dispersed work, and the resulting implications for knowledge creation, sharing and integrating, we suggest, make our model both heuristically and as an explanatory device eminently applicable as a basis for research in many other contexts where coordination is a key concern.

In terms of practical implications, the paper illustrates micro coordination practices in relation to four types of knowledge processes. We compared a successful project (SAP) with an unsuccessful project (Baan). But an even more important finding emerges. Instead of focusing on coordination mechanisms per se, and (as found in many studies) over-relying on the efficacy of tools and technologies as coordination mechanisms (e.g. Cheng et al. 2004; Ebert and De Neve 2001; Majchrzak et al. 2000; Mockus and Weiss 2001; Smith and Blanck 2002), to ensure successful outcomes, managers should consider their organization in terms of knowledge processes, not just information flows. They must then focus on how coordination mechanisms facilitate knowledge processes in all four areas. Here our research suggests that technologies are most useful for *amplifying knowledge management processes*, that is, for allowing knowledge sharing. Organization design *facilitates knowledge flows* across organizations and teams. Work-based mechanisms *make knowledge explicit and accessible*, while social mechanisms are needed to *build social capital* and exchange knowledge and ideas. While these mechanisms can and do have multiple uses, it is clear that unless management practices take on board their various knowledge pay-offs, coordination in increasingly knowledge-intensive work environments will become much more problematic.

The framework developed in this paper proposes a one-directional, singular relationship between coordination mechanisms and knowledge processes. However it is possible that one coordination mechanism (or combination of them) affects more than one knowledge process. While it is possible that each coordination mechanism affects mainly one knowledge process, the effect on other knowledge processes cannot be totally ignored. Therefore in our further research we will investigate the interrelationships between coordination mechanisms and knowledge processes.

References

Adler, P.S. "Interdepartmental interdependence and coordination: The case of the design / manufacturing interface," *Organization Science* (6:2) 1995, pp 147-167.

Battin, R.D., Crocker, R., and Kreidler, J. "Leveraging Resources in Global Software Development," *IEEE Software* (18:2) 2001, pp 70-77.

Bechky, B.A. "Sharing Meaning Across Occupational Communities: The Transformation of Understanding on a Production Floor," *Organization Science* (14:3) 2003, pp 312-330.

Boland, R.J., and Tenkasi, R.V. "Perspective making and perspective taking in communities of knowing," *Organization Science* (6:4) 1995, pp 350-372.

Bowker, G., and Starr, S. *Sorting Things Out: Classification and Its Consequences* MIT Press, Cambridge MA, 1999.

Brown, J.S., and Duguid, P. "Knowledge and Organization," *Organization Science* (12:2) 2001, pp 198-213.

Carmel, E. *Global Software Teams: Collaborating Across Borders and Time Zones*, (1st ed.) Prentice-Hall P T R, Upper Saddle River, NJ, 1999.

Cheng, L., De Souza, C.R.B., Hupfer, S., Patterson, J., and Ross, S. "Building Collaboration into IDEs," *Queue* (1:9) 2004, pp 40-50.

Ciborra, C.U., Orlikowski, W.K., Failla, A., Patriotta, G., Bikson, T.K., Suetens, N.T., and Wynn, E. *Groupware & Teamwork: Invisible Aid or Technical Hindrance?* John Wiley, Chichester, UK, 1996.

Cramton, C.D. "The Mutual Knowledge Problem and Its Concequences for Dispersed Collaboration," *Organization Science* (12:3) 2001, pp 346-371.

Crowston, K. "A coordination theory approach to organizational process design," *Organization Science* (8:2) 1997, pp 157-175.

Donnellon, A., Gray, B., and Bougon, M.G. "Communication, Meaning, and Organized Action," *Administrative Science Quarterly* (31:1) 1986, pp 43-55.

Dougherty, D. "Interpretive barriers to successful product innovation in large firms," *Organization Science* (3:2) 1992, pp 179-202.

Ebert, C., and De Neve, P. "Surviving Global Software Development," *IEEE Software* (18:2) 2001, pp 62-69.

Eisenhardt, K.M. "Building Theories from Case Study Research," *Academy of Management Review* (14:4) 1989, pp 532-550.

Faraj, S., and Sproull, L. "Coordinating Expertise in Software Development Teams," *Management Science* (46:12) 2000, pp 1554-1568.

Gabarro, J.J. "The development of working relationships," in: *Intellectual teamwork: Social and technological foundations of cooperative work,* J. Galegher, R.E. Kraut and C. Egido (eds.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1990, pp. 70-110.

Galbraith, J.R. *Designing complex organizations* Addison-Wesley, Reading, Massachusetts, 1973.

Giddens, A. *The consequences of modernity* Polity Press, Cambridge, 1991.

Goodhue, D.L., and Thompson, R.L. "Task-Technology Fit and Individual Performance," *MIS Quarterly* (19:4) 1995, pp 213-235.

Grant, R.M. "Toward a knowledge-based theory of the firm," *Strategic Management Journal* (17:Winter) 1996, pp 109-122.

Hoffer Gittell, J., and Weiss, L. "Coordinating Networks Within and Across Organizations: A Multi-level Framework. ," *Journal of Management Studies* (41:1) 2004, pp 127-153.

Jarvenpaa, S.L., Knoll, K., and Leidner, D.E. "Is anybody out there? Antecedents of trust in global virtual teams," *Journal of MIS* (14:4) 1998, pp 29-64.

Kogut, B., and Zander, U. "What Firms Do? Coordination, Identity, and Learning," *Organization Science* (7:5) 1996, pp 502-518.

Kotlarsky, J., and Oshri, I. "Social Ties, Knowledge Sharing and Successful Collaboration in Globally Distributed System Development Projects," *European Journal of Information Systems* (14:1) 2005, pp 37-48.

Krauss, R.M., and Fussell, S.R. "Mutual knowledge and communicative effectiveness," in: *Intellectual teamwork: Social and technological foundations of cooperative work,* J. Galegher, R.E. Kraut and C. Egido (eds.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1990.

Kumar, K., van Fenema, P.C., and Von Glinow, M.A. "Intense Collaboration in Globally Distributed Work Teams: Evolving Patterns of Dependencies and Coordination " in: *Managing multinational teams: Global perspectives.,* D.L. Shapiro, M.A. Von Glinow and J.L.C. Cheng (eds.), Elsevier/ JAI., Oxford, 2005.

Levesque, L.L., Wilson, J.M., and Wholey, D.R. "Cognitive Divergence and Shared Mental Models in Software Development Project Teams," *Journal of Organizational Behavior* (22) 2001, pp 135-144.

Majchrzak, A., and Malhotra, A. "Virtual Workspace Technology Use and Knowledge-Sharing Effectiveness in Distributed Teams: The Influence of a Team's Transactive Memory," in: *Knowledge Management Knowledge Base* 2004.

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.

Malone, T.W., Crowston, K., Lee, J., and Pentland, B. "Tools for Inventing Organizations: Towards a Handbook of Organizational Processes," *Management Science* (45:3) 1999, pp 425-443.

Miles, M.B., and Huberman, A.M. *Qualitative Data Analysis: an expanded sourcebook*, (2nd ed.) Sage, CA, 1994.

Mintzberg, H. *The Structuring of Organizations* Prentice-Hall, Englewood Cliffs, N.J., 1979.

Mockus, A., and Weiss, D.M. "Globalization by Chunking: A Quantitative Approach," *IEEE Software* (18:2) 2001, pp 30-37.

Moreland, R.L. "Transactive Memory: Learning Who Knows What in Work Groups and Organizations," in: *Shared Cognition in Organizations: The Management of Knowledge,* L. Thompson, D. Messick and J. Levine (eds.), Lawrence Erlbaum, Mahwah, NJ, 1999.

Ngwenyama, O.K., and Lee, A.S. "Communication Richness in Electronic Mail: A Critical Social Theory and the Contextuality of Meaning," *MIS Quarterly* (21:2) 1997, pp 145-167.

Nidumolu, S.R. "The Effect of Coordination and Uncertainty on Software Project Performance," *Information Systems Research* (6:3) 1995, pp 191-219.

Nidumolu, S.R. "A comparison of the structural contingency and risk-Based perspectives on coordination in software-development projects," *Journal of Management Information Systems* (13:2) 1996, pp 77-113.

Nonaka, I. "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science* (5:1) 1994, pp 14-37.

Olson, J.S., and Olson, G.M. "Culture surprises in remote software development teams," *Queue* (1:9) 2004, pp 52-59.

Orlikowski, W.J. "The Duality of Technology: Rethinking the Concept of Technology in Organizations," *Organization Science* (3:3) 1992, pp 398-427.

Orlikowski, W.J. "Knowing in Practice: Enacting a Collective Capability in Distributed Organizing," *Organization Science* (13:3) 2002, pp 249-273.

Sarker, S., and Sahay, S. "Implications of space and time for distributed work: an interpretive study of US-Norwegian system development teams," *European Journal of Information Systems* (13:1) 2004, pp 3-20.

Smith, P.G., and Blanck, E.L. "From Experience: Leading Dispersed Teams," *The Journal of Product Innovation Management* (19) 2002, pp 294-304.

Spender, J.-C., and Grant, R.M. "Knowledge and the Firm: Overview," *Strategic Management Journal* (17:Winter) 1996, pp 77-91.

Strauss, A.L., and Corbin, J.M. *Basics of Qualitative Research*, (2nd ed.) Sage Publications, Thousand Oaks, CA, 1998.

Szulanski, G. "Exploring Internal Stickiness: Impediments to the Transfer of Best Practice within the Firm," *Strategic Management Journal* (17:Winter) 1996, pp 77-91.

Van de Ven, A.H., Delbecq, A.L., and Koenig Jr, R. "Determinants of Coordination Modes Within Organizations," *American Sociological Review* (41:April) 1976, pp 322-338.

van Fenema, P.C. "Coordination and Control of Globally Distributed Software Projects," in: *Department of Decision and Information Sciences*, Erasmus University, Rotterdam School of Management, Rotterdam, The Netherlands, 2002, p. 572.

von Hippel, E. ""Sticky information" and the locus of problem solving: Implications for innovation," *Management Science* (40:4) 1994, pp 429-439.

Walsham, G. "Cross-Cultural Software Production and Use: A Strucructurational Analysis," *MIS Quarterly* (26:4) 2002, pp 359-380.

Wegner, D.M. "Transactive Memory: A Contemporary Analysis of the Group Mind," in: *Theories of Group Behavior,* G. Mullen and G. Goethals (eds.), Springer Verlag, New York, 1987.

Yin, R.K. *Case Study Research: Design and Methods* Sage, Newbury Park, CA, 1994.

Yoo, Y., and Kanawattanachai, P. "Developments of Transactive Memory Systems and Collective Mind in Virtual Teams," *International Journal of Organizational Analysis* (9:2) 2001, pp 187-208.

Zuboff, S. *In the Age of the Smart Machine:The Future of Work and Power* Basic Books, New York, 1988.