Journal of International Technology and Information Management

Volume 15 | Issue 2 Article 1

2006

A Collaborative Project Management Approach and a Framework for Its Supporting Systems

Fang Chen
University of Manitoba

Nicholas C. Romano Jr
Oklahoma State University

Jay F. Nunamaker Jr
The University of Arizona

A Collaborative Project Management Approach and a Framework for Its Supporting Systems

Fang Chen
University of Manitoba

Nicholas C. Romano, Jr.
Oklahoma State University

Jay F. Nunamaker, Jr.
The University of Arizona

ABSTRACT

The project management (PM) paradigm has changed during the past decade due to the globalization of business and advancement of information technology (IT). Increasingly, projects involve members from different geographical locations more than at any other time in history. Traditional PM methods may be inadequate to manage distributed projects, and new information systems are needed to support distributed PM. This article provides the rationale for a collaborative PM approach to manage distributed projects and proposes a conceptual framework for the collaborative PM software development. The objective is to present a collaborative PM framework that can guide further research and development in this challenging area.

INTRODUCTION

The project landscape has changed dramatically during the past decade. One change is that project is becoming a pervasive organizational phenomenon (Lundin and Hartman, 2000), and project management (PM) is becoming more like a profession. Some organizations in U.S.A started to require their project managers to be certified by PMI (Project Management Institute), the well-known non-profit leading professional association in the area of PM (Mahaney and Greer, 2004). Another change is that, increasingly, projects involve professionals from different geographical locations due to globalization of business (Evaristo and Fenema, 1999). Although distributed projects provide benefits in terms of increasing team members' competency, obtaining required resources from different sites, reducing costs and generating synergy among team members, they also impose significant challenges for coordinating and monitoring team performance (Bourgault, Lefebvre, Lefebvre, Pellerin and Elia, 2002; Jonsson, Novosel, Lillieskold and Eriksson, 2001). Traditional PM methods are inadequate to manage distributed projects (Nidiffer and Dolan, 2005).

The PM paradigm has been shifting from a restrictive (traditional) management approach toward a more collaborative approach (Cleetus, Cascaval and Matsuzaki, 1996; Evaristo and Fenema, 1999; Jonsson, Novosel et al., 2001; Maurer, 1996; Romano, Chen and Nunamaker, 2002). In the more restrictive PM approach, the focus was on '*management*' or '*control*', which implied a top-down view of how projects are conducted and controlled (Cleetus, Cascaval et al., 1996; Augustine, Payne, Sencindiver and Woodcock, 2005). A few individuals who were high in the organizational hierarchy maintained project control and were the decision makers and information owners. Others had very limited opportunities to participate in the decision-making process, and they only shared information related to their specific tasks; therefore, what possible impacts their individual work might have on the entire project was not always clear to them. However, this style of PM only works well in repeat product and process environments (Graham and Englundm, 1997). The assumption that projects are conducted within repeat product and process environments is no longer valid for most projects today due to rapid information technology (IT) advancement, business globalization, high personnel turnover, increasing project complexity and distributed team membership (Evaristo and Fenema, 1999; Graham and Englundm, 1997; Jonsson, Novosel et al., 2001; Romano, Chen et al., 2002).

Distribution of project members changes the nature of the roles played by project managers and team members. Project managers now serve more as coordinators rather than information keepers and disseminators or a "benevolent dictator in a top-down hierarchical structure" as they did in the past (Nidiffer and Dolan, 2005) . Distributed PM takes a more flattened "*network view*" of a project (Whittaker, 2000). Project members share project information, decision-making power, and responsibility for project processes and outcomes. Information flow may be in all directions (Jonsson, Novosel et al., 2001; Augustine, Payne et al., 2005) rather than just the top-down and/or the bottom-up.

Managing project workforce and coordinating team efforts relies on information technologies (Nidiffer and Dolan, 2005; Potter and Balthazard, 2000). However, information systems designed to support the traditional approach are becoming inadequate as an increasing number of projects become distributed and PM shifts toward a more collaborative approach. Marttiin, Lehto, and Nyman (2002) pointed out: "*Conventional fact-to-face practices form an essential part of successful project co-working. But this is often impossible in globally-dispersed projects. New information and communication technology solutions are needed for converting collaborative actions into virtual ones.*" Also, Nidiffer and Dolan (2005) stated: "*Most project management techniques were designed for co-located teams. Those techniques may prove ineffective in global, multi-site organizations.*"

When project teams become distributed, "*the communication, coordination, and tracking of ongoing project activity become key issues for project success*" (Ly, 1997). Other researchers have also expressed that these and other functions may need to be supported by PM systems that are more collaborative. Researchers and practitioners in the PM field have indicated that PM systems should support basic PM functions and additional functions such as collaboration (Jonsson, Novosel et al., 2001; Clarke, 1999); information sharing and file management (Weiser and Morrison, 1998; Hefner, 2000; Abramovici and Gerhard, 2000); and project process management (Marttiin, Lehto et al., 2002; Turner, 2000; Abramovici and Gerhard, 2000).

PM systems for co-located projects should support these PM functions; however, the need is more urgent when project team members are distributed. One indication that there is a need for new PM tools is the expanding market for "distributed" or "collaborative" PM systems reported by Collaborative Strategies (Distributed project management, 2004). The report estimates that revenues from distributed PM software will rise from $888 million in 2002 to nearly $7.2 billion in 2007.

The rising need for new PM systems that can support distributed projects has been evidenced by researchers developing distributed PM prototypes in recent years. Lam and Maheshwari (2001) developed a PM tool to conduct task and team management. Abramovici and Gerhard (2000) introduced a product data management system to support virtual engineering cooperation. Lysakowski and Doyle (1998) proposed to develop an information system to act as an electronic lab notebook for pharmacy research. Other researchers have described different systems that facilitate project document management and knowledge management (KM) (Weiser and Morrison, 1998; Baek and Liebowitz, 1999; Hefner, 2000; Katzy, Evaristo and Zigurs, 2000; McManus and Snyder, 2003). These systems were developed with different core PM functions; some focused on task and team management, while others focused on document management. This piecemeal approach of implementing different PM functions in different systems has advantages and disadvantages. There are two major advantages. First, since the system focuses on one or two major PM functions, it is easier to develop and maintain, and may be easy to use. Second, since the system is easy to develop, the cost of developing and purchasing the system is reasonable; therefore, it is realistic to implement such a system in organizations, large or small. However, the major disadvantage of this approach is that project members typically use a number of software packages to deliver PM functions (Jaafari and Manivong, 1998). Jaafari and Manivong (1998) explained the inefficiencies and drawbacks associated with using multiple software packages to deliver PM functions.

"Firstly, there will be multiple data entries; each using their own data modeling and structure. Second, there will be difficulties in co-ordination of information across these software packages. In fact, the probability of a blunder in data entries increases with an increase in the number of systems used. Third, there will be no integrated (compounded effect) analysis; this is not a desireable situation when judged against the fact on most projects one disturbance will have a compound (chain reaction) effect on many aspects of the project" (Jaafari and Manivong, 1998).

Jaafari and Manivong (1998) described an "idealized" system that incorporates a variety of PM functions and suggested two approaches to address the gap between the current PM systems and the "idealized" system. One approach is to extend the existing system to include the required capabilities, and the other approach is to build new PM systems that include all desired capabilities. Instead of explaining how to extend an existing system or how to build a new system, our aim is to establish a conceptual framework that provides a holistic view of what "desired" or "idealized" capabilities a collaborative PM system needs to have, since we think Jaafari and Manivong's (1998) "idealized" system largely ignores collaboration among project team members.

The following two examples illustrate the need for a comprehensive and holistic view of PM functions. Lam and Maheshwari (2001) described their effort to extend an existing PM software prototype to include more functionalities. Their first prototype provided a project repository and configuration management, then a critical analysis of the prototype revealed that the design had two holes: it did not include task or team management. Therefore, they had to extend the first prototype by including these two management functions. Task and team management are very important and salient components of PM functions, and yet, the development team discovered their design inadequacy after the first prototype had already been developed. This example illustrates the need for a more comprehensive PM framework. If the development teams were aided with a comprehensive PM framework, then they might not have made the mistake. They may still choose to implement a project repository and configuration management before they implement team and task management; however, they should have the opportunity to make an informed decision rather than jump into development without realizing some other functions should also be considered.

Balk and Kedia (2000) described a company that built a single PM system by integrating Commercial-off-the-Shelf (COTS) products. The integration worked well, data could flow among different products and be saved in a central database. The advantage of this type of development was that "*deficiencies in individual component products can be compensated for by judicious selection of other component products. Defects or designed issues can be rapidly worked around by adding low level coding to the component interfaces*" (Balk and Kedia, 2000). According to the authors, this approach saved development time and money and increased user buy-in. This company simply integrated products they were using, and may not need a comprehensive PM framework to guide their choice of individual PM products. However, for other companies, if they would like to follow a similar approach to develop a PM system but have few products that are being used, then a comprehensive PM framework would help project managers and software development teams to evaluate and choose appropriate products to integrate into the PM system.

A few researchers perceived the need for a PM framework and proposed different models to identify PM components and constituent functions that are required to design and develop a new type of PM system that can support complex and/or distributed projects.

Maurer (1996) summarized the project coordination literature and proposed a PM framework that incorporated a variety of PM functions; however, his model focused more on management and ignored a large part of collaboration among team members, such as group discussion, negotiation, communication, group writing and group meetings. Jaafari and Manivong (1998) proposed an "idealized" PM framework, analyzed and listed management functions for an idealized PM system, and ranked a sample of PM systems according to the list. Derived from the project life cycle, their list of management functions can help project managers and members to see what they need to do to manage a project in its life cycle. However, there are two limitations of their list. First, the list is mainly from project members' perspective and not from a software development perspective. If a team of software engineers would like to develop a PM system, then the list may not provide much help for them to visualize what functions the PM system needs to implement. Second, the list is not comprehensive; it does not include a project repository or document management, both of which are very important parts of any PM system. Also, it did not include collaboration among project members.

We attempt to fill this gap in the research by proposing a more comprehensive collaborative PM framework, which may help project managers and members to understand what system functions they may need to conduct collaborative PM, help PM developers to visualize major components and constituent functions PM systems need to support, and initiate more research on how to develop PM systems and how to use these systems to achieve more efficient and effective PM results.

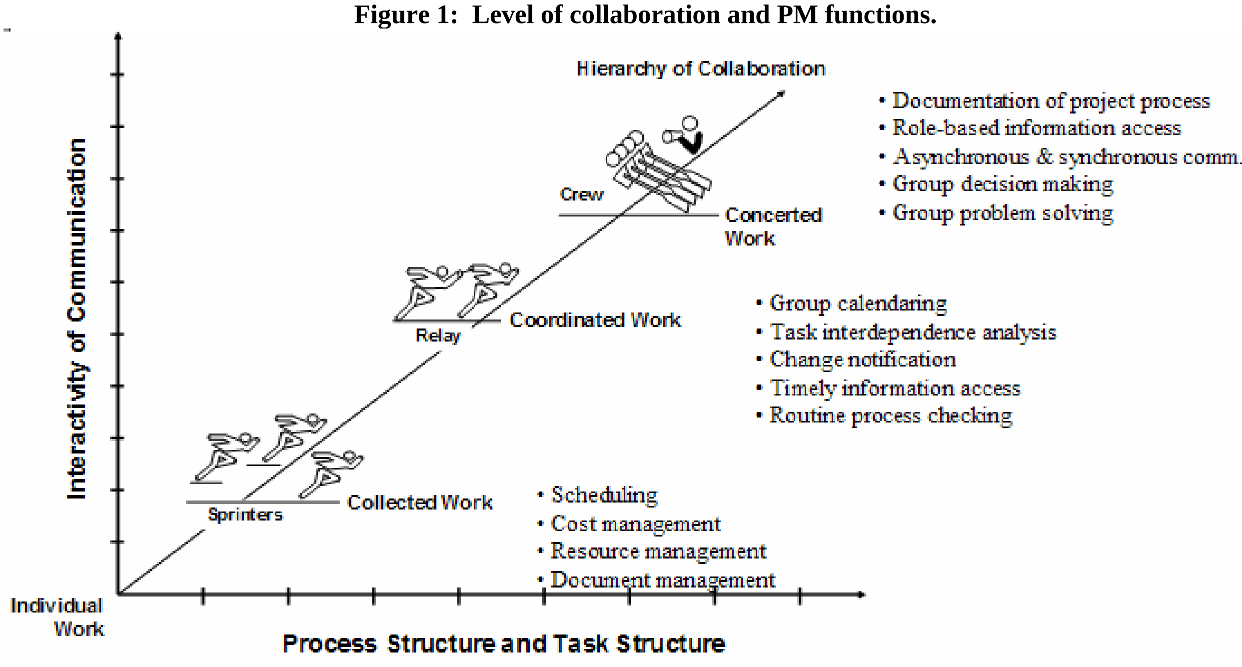
The remainder of this paper is structured as follows: Section 2 explains levels of collaborative support and the implications for PM; Section 3 describes the collaborative PM approach; Section 4 presents the results of a PM software comparison; Section 5 proposes a conceptual framework for collaborative PM system development; and Section 6 concludes with a discussion of implications and future research directions.

COLLABORATIVE SUPPORT LEVELS

PM and team collaboration need to be supported throughout the project life cycle. However, different people have different interpretations of what collaboration means. This section explains our understanding of collaboration in the context of PM.

We define *collaboration* as joint effort toward achieving a mutual goal. Collaboration can be represented hierarchically. As people collaborate, there are at least three modes in which they can work: collected, coordinated and concerted (Nunamaker, Romano and Briggs, 2001). Figure 1 displays this collaboration hierarchy. The higher up the hierarchy a team operates, the higher the requirements for task and process structure, and the higher the requirement for communications interactivity. The PM functions for collaboration levels are listed in Figure 1, as well.

Figure 1: Level of collaboration and PM functions.



(Nunamaker, et al., 2001)

At the collected level, group productivity is the simple sum of all individual efforts. This is analogous to a team of sprinters, each of whom makes the best individual effort possible, and then the accomplishments of all are added to comprise the overall team contribution. Communication and collaboration are minimal at best. Processes are individualistic start-to-finish and usually not integrated until the individuals' outcomes are combined. Process structure and task structure may be low or nonexistent. Typical PM scenarios at this level may be a co-located project and uncoupled or very loosely coupled tasks conducted in a static environment. PM tools at this level should support scheduling, cost, resource, task, and document management.

At the coordinated level, the success of some members depends on the timely receipt of deliverables from others. Therefore, team success depends on the ability to coordinate efforts. This mode of work is like a team of relay runners, each of whom makes their best individual effort but must also execute carefully coordinated hand-offs to the next member in the process. This level of collaboration involves managing activity dependencies (Sena and

Shani, 1999). Coordinated processes are typically ordinal and characterized by hand-offs and progressive integration; thus, this level is more structured in terms of process order, specific milestones and hand-offs than the collective level. The need for interactive communication also increases within the collective level, such that team members can monitor progress toward hand-offs. PM at this level requires coordination among project individuals, and tools should support group calendaring, task dependence analysis, timely change notification, easy access to project information, and routine tracking of project process, in addition to all collected collaboration functions.

At the concerted level, performance of any one member may directly and immediately influence the performance of all members. A crew team is a useful metaphor for this level of work. All rowers must synchronize their efforts and contribute simultaneously and synergistically to achieve a near optimal level of performance. An aggregation of uncoordinated, individual efforts would yield nothing. Task and process structure must be far higher for concerted work than for coordinated work, and the need for interactive communication may be nearly continuous. PM at the concerted level requires tight coordination among project individuals. In addition to the functions supported at the collected and coordinated level, PM must provide some more advanced functions including explicit process documentation, document version control, document co-authoring, role-based information access and support for synchronous and asynchronous communication, decision making and problem solving.

In order to support distributed project team leaders and participants effectively, a collaborative PM system must be designed to support all of the three collaboration levels. The next section discusses our collaborative PM approach and conceptual framework for PM system development.

A COLLABORATIVE PM APPROACH

The discussions in Sections 2 and 3 imply that a variety of management functions should be incorporated into collaborative PM software so that it can successfully support distributed project managers and participants across the project life cycle. Different researchers classify these functions into different categories. Huang, Feng and Mak (2001) classified the computerized systems to support management of collaborative product development projects into three categories: management of project, workflow, and product data. Coleman (1997) asserts that the collaboration functions that organizations support fall into five main categories: document management, group calendaring/scheduling, PM, communication, and KM. Table 1 presents the results of incorporating both perspectives to group major PM functions into four types of support. Each of them is discussed in detail in the sections that follow.

Table 1: Four major components of collaborative PM approach.

Components	Descriptions	Functions
Basic PM Support	Scheduling, Time Management Resource Management Cost Management Task Analysis Task Allocation Status Tracking Reporting	Collaborative Calendaring / Gantt Chart Resource Management Cost Management Work-Breakdown-Structure Task Dependency Management , Pert Chart, Status Tracking Reporting
Knowledge Management	Develop High Levels of Project Awareness Project Dictionary Business Rules & Policies Project Context Info All Other Project-Related Info	Electronic Doc Repository With Functions of Uploading/Downloading Updating Searching (Key Word and Full Text Search) Browsing Document Version Control Role-Based Access
Process Management	Conduct Project Tracking and Increase Project Process Visibility	Work Flow Management Integration Management Change & Risk Management Issues Management

		Action Items Management Collaborative Process Structuring
Communication and Collaboration Support	Facilitate Communication in Synchronous & Asynchronous Mode, Group Decision Making, Problem Solving	Session Management Desktop Sharing Video & Audio Conference Support Idea Generation, Organization Consensus Polling Issue Exploration Group Writing and Modeling Shared Whiteboard

Basic PM Support

Basic PM support is essential to manage all project types and includes scheduling, budget management, task interdependence analysis, milestones, critical path management, human resources and equipment management. Post and Kagan (2005) pointed out that project managers often are reluctant to admit the failure of project being over budget and delayed due to the lack of awareness of cost, overall picture of the project objectives and directions. Used appropriately, basic PM support and knowledge management support will likely increase project managers' awareness of project related activities, obtain more accurate estimate of project progress, and make decisions accordingly.

Knowledge Management

The purpose of KM is to increase project awareness by capturing the key processes of the project and to allow timely access to current and accurate project information. KM is a very important dimension of organizations and PM (Nonaka, 1994; McManus and Snyder, 2003; Wu, Chu, Li, Han and Sculli, 2003). By using KM, successful thinking and effective practicing become visible, manageable, and useful to more than one person. Key know-how knowledge will remain even if employees leave the project team or the project team is disbanded (McManus and Snyder, 2003; Chan, 2004). KM can be practically implemented through an electronic repository. A paper-based repository has several drawbacks including retrieval delays, lost documents, and storage problems (Back and Moreau, 2001; Kingman, Lambert and Steen, 1990; Teicholz and Fisher, 1994). An electronic project repository needs to incorporate the following four components to facilitate information sharing.

- Project Dictionary: to define and clarify key terms, concepts, jargon and methodology.
- Project Business Rules and Policies: explicit specifications of project rules and policies.
- Project Context Information: documentation of project background, boundary, objectives, and project requirements. Project requirements management is one of the critical factors that influence project success (Chen, Jiang, Chen, and Shim, 2004), therefore requirements should be documented, and changes to project requirements should be closely tracked and controlled.
- Comprehensive Knowledge Capture: all other project-related data, information and knowledge that can be captured for later retrieval and use.

Process Management

During the project life cycle, project managers and members usually start by defining the goal for the project, then they make the plan to obtain the goal. The plan involves breaking projects into manageable tasks, analyzing interdependence among tasks, defining milestones for the project, and assigning individuals to tasks. When the project starts, managers and members execute process management to coordinate the team effort, check the project progress against the plan, and make sure the project is moving toward its goal efficiently and effectively. The purpose of process management is to increase process visibility, ensure the smooth handover of tasks among project members, reduce misunderstanding among members, avoid unnecessary rework, discover problems and issues early, make changes accordingly, control change effects, and ensure task quality.

Although process management is critical for project success, the reality is that people often ignore it (Turner, 2000). LaBrosse (2004) stated:

"According to a February 2003 study by The Center for Business Practices (CBP), the largest PM challenge facing companies is implementing a consistent process… According to the Standish Group's CHAOS report that reviewed more than 40, 000 projects in the last 10 years, when there is not a consistent process for doing PM in a company, companies waste up to 20 percent of all project dollars spent" (LaBrosse, 2004).

If people only manage project inputs and outputs, then the process remains a black box and project members typically don't realize something has gone awry until it may be too late to correct the problem without causing large amounts of rework or unanticipated costs and delays. In other words, without effective and consistent process management, the execution of the project plan will be in an ad hoc way rather than a systematic way. Project managers and members respond to issues, problems, changes and risks reactively instead of proactively. This may largely decrease the possibility of project success. Collaborative PM efforts, therefore, must support process management, which can be viewed from two levels of task granularity. At the higher level, process management needs to manage workflow, task integration, change and risks. At the lower level, or a day-to-day operation level, process management needs to manage daily tasks, issues, problems and actions.

Workflow management concerns the core components of the project and may be conducted in repeated patterns for specific project types. Task integration specifies the relationship among individual tasks, as well as procedures of how individual tasks can be integrated to form a cohesive, functional unit. Project change management is to ensure change is appropriate for the project and the effect of change is minimized; to achieve this, project managers and members need to closely monitor changes to the project plan, tasks, resources and documents. Risk management is to predict and analyze design plans to mitigate risks.

In addition to managing the project process at a higher level than they would in traditional projects, project members also need to manage project process at the operational level. Approaching deadlines for tasks need to be monitored and enforced. Issues, and problems that emerge in the project process need to be addressed. Actions may be taken to address these problems or issues, and the actions need to be tracked. If there is a repeated pattern to resolve problems and issues, or to track actions and tasks, then project members can use collaborative process structuring to streamline these process procedures into templates.

Communication and Collaboration Support

In communication and collaboration support, the term "collaboration" is used in a narrower sense here. Communication and collaboration support includes communication, meeting, and group decision-making support in synchronous and asynchronous mode. The software would allow participants to engage in divergent thinking (e.g., idea generation and issue exploration) and convergent thinking (e.g., idea organization and consensus polling) during group meetings. Session management refers to how to transfer data from one meeting session to the next. More advanced software may support shared agenda, document co-authoring and collaborative modeling, desktop sharing, shared whiteboard, or even video and audio conference.

PM SOFTWARE COMPARISON

Our Web search indicated that a variety of PM systems exist that have some of the above-mentioned functions. However, no one system has all of them. The appendix lists a sample of PM systems. Traditional PM software is a stand-alone application, such as Microsoft Project. Current trends reveal that the PM systems are Web-based and provide at least basic PM support. Document management systems are usually Web-based and provide almost all of the document management features in Table 1; more advanced systems may support searches for any words in the documents. Some systems also support document approval signatures and change notification functions such as MatrixOne Document Central.

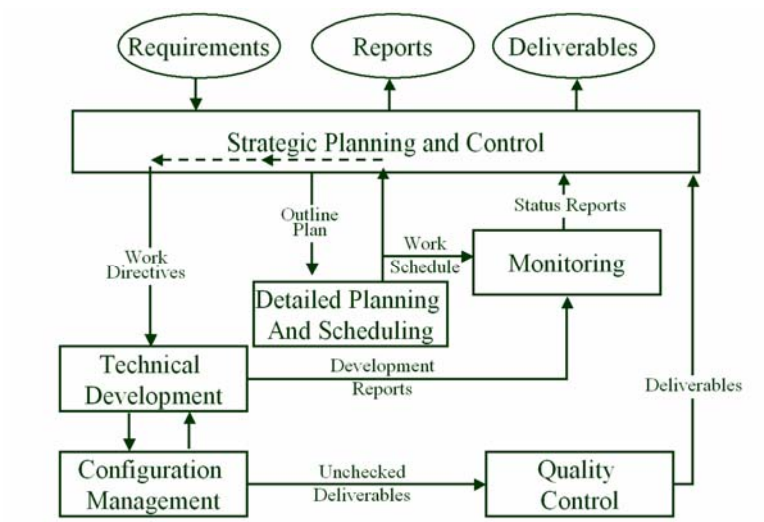
Collaboration support in this article focuses on communication, meeting and group decision-making support. We classified the systems in this category into two types: text communication with process & task structure and process & task support (i.e. Cognito and Facilitate 8.0); as well as voice and video support with minimal text communication and no process or task structure, nor any process or task support (i.e. WebEx and Lotus SameTime). Most likely, both types of systems are required to adequately support collaborative PM for distributed projects.

Other systems listed in the appendix support two or more of the major functional areas. It seems that none of the systems are able to find support for all four components: PM, KM, process management, and collaboration. In practice, team members may need to use different systems for different functions to support their project activities. However, it may not be a trivial issue to administer, maintain, and train users for different systems. It also may be difficult to transfer the outputs from one system to another. A suite of tools sharing the same platform and user interface may provide more efficient and effective IT support for PM. In the next section, we propose a conceptual collaborative PM framework within which such a suite of tools could be built.

A COLLABORATIVE PM FRAMEWORK

A few researchers have proposed general PM frameworks. Figures 2 and 3 present two that influenced the development of our collaborative PM framework. Figure 2 is a software development PM tool framework developed by Dixon (1988). While useful, Dixon's model has some limitations that make it less than optimal for today's distributed projects. First, it lacks a project repository and has no collaborative aspects. Second, the management process is sequential in nature and the influence of one module on the next is one-way. In real PM situations, different management considerations may influence one another in parallel and cyclic ways, and there is seldom a sequential or one-way influence. This model may be applicable to well-defined and repeat project environments; however, it may underestimate the complexity of distributed projects and the collaborative support required.

Figure 2 : Dixon's project management model.



(Dixon, 1988)

Figure 3 presents a generic framework of a project coordination system discussed by Maurer (1996). This framework goes further than the one presented by Dixon, but it still lacks the integration of some important aspects of collaborative support that we think are necessary for successful distributed PM.

Figure 3: Maurer's project coordination model. (Maurer, 1996)

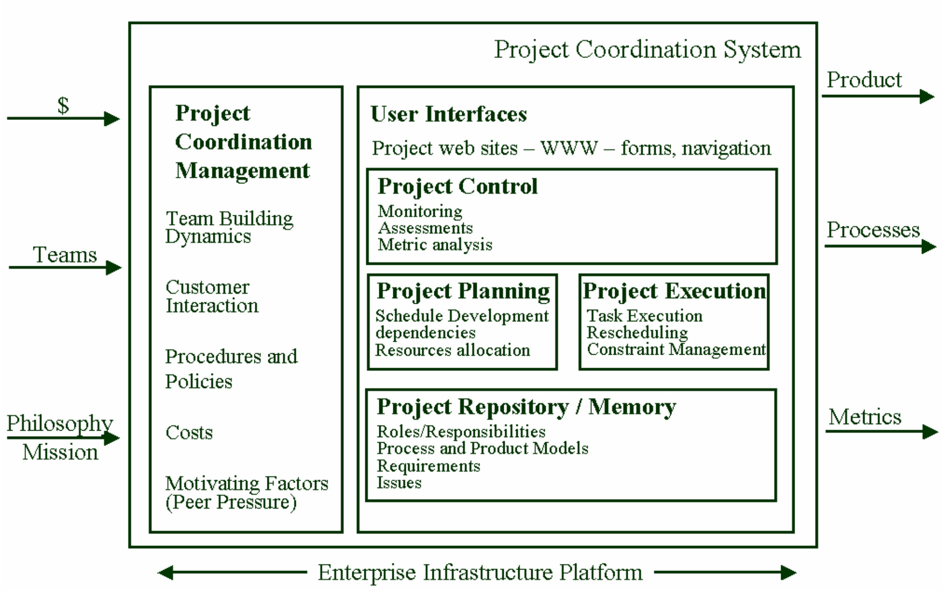
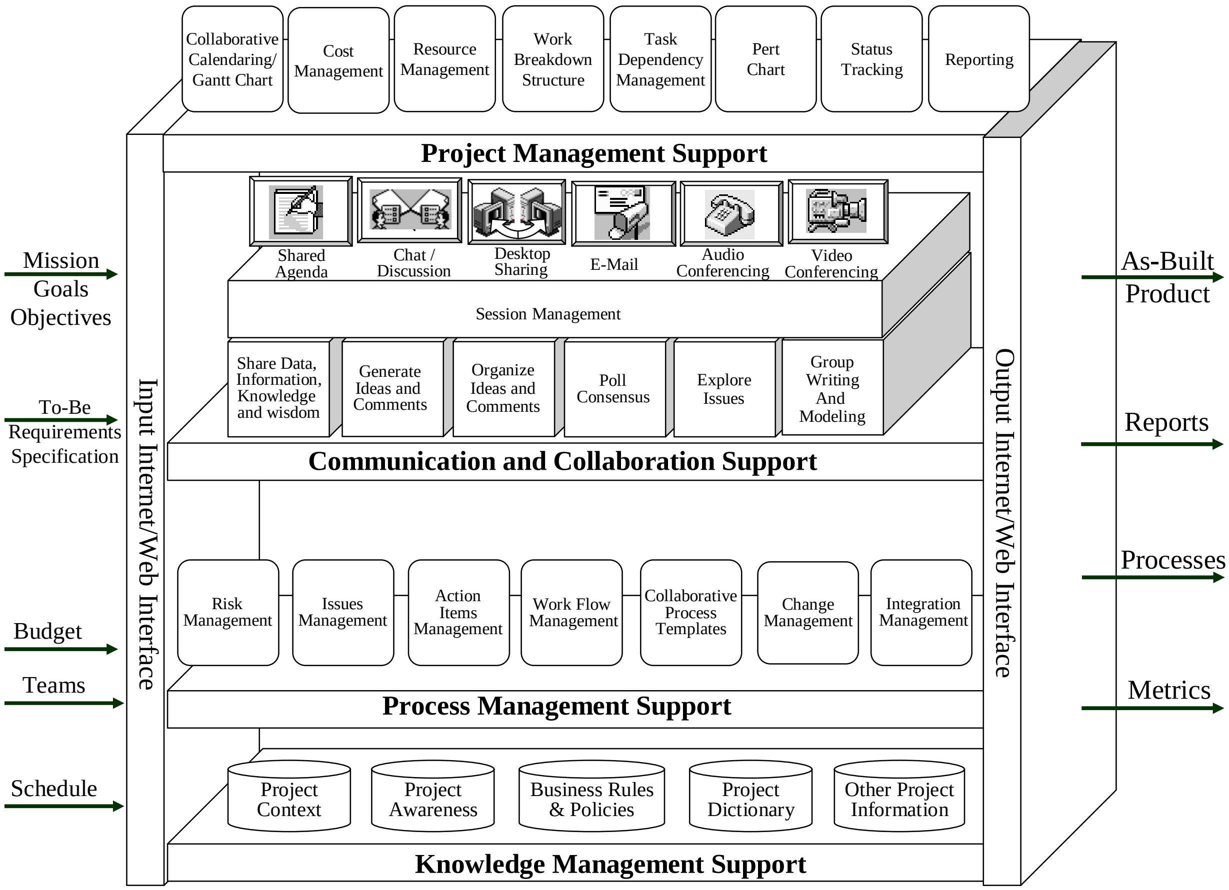


Figure 4 illustrates our extended framework based upon the earlier ones of Dixon and Maurer. The diagram only specifies the major components that need to be incorporated into the systems, and the implementation level architecture is not presented in this paper. The diagram does not mean that a single tool has to implement all of these features.

One development option is to form a product family by implementing these major components in separate tools on a common platform, providing central data storage and interfaces among the tools so that data can flow seamlessly among tools. A middleware approach to implement the system may be appropriate in this case. The middleware is "*a variety of technologies that links a client application with one or more server applications*" (Gregory and Briggs, 2002). Gregory and Briggs (2002) proposed a system architecture that incorporated several types of middleware, communication control, objected oriented data store, and relational data store. The detailed discussion of the middleware approach is beyond the scope of this paper. We think that Gregory's architecture may have implications for other collaborative systems, even though the original architecture was developed for a Web-based Group Support System.

There are two major limitations of our framework. One limitation is that it is only a conceptual framework; we do not provide a detailed implementation system architecture, even though we suggest a middleware approach may be used. The other limitation of the framework is that it only considers how to manage a single project at a time. In practice, many organizations may have multiple projects running at the same time, and these projects may involve multiple organizations and teams. In this case, "meta-project" management function may need to be considered, such as program management, or profile management. Program/profile management may facilitate managers with prioritizing different projects, resolving resource allocation conflicts, conducting scheduling and budgeting across multiple projects/organizations, analyzing change effects of one major decision across multiple projects, and understanding individual projects in an organization-wide framework. Despite these limitations, this framework has useful implications for software development, project managers, and researchers.

Figure 4: A collaborative project management Framework.



IMPLICATIONS OF THE FRAMEWORK AND FUTURE RESEARCH

Implications for Software Development

The framework provides some guidance for developing collaborative PM systems. A software development team may develop an integrated, comprehensive, and collaborative PM based upon our framework. Even if the software development team decides to implement only some of the PM functions, this framework will still be useful in terms of giving the team a holistic view of the PM so it can prioritize PM functions for development. The team members can decide which functions are basic functions, which functions are good to have, and which functions are rarely used. If they determine not to implement certain PM functions, then it is because they have good reasons to do so, not because they are not aware of these functions. If a software development team needs to integrate different commercial software to support PM effort, then it can also use this framework to select the proper software with desired functions.

Implications for Project Managers and Practitioners

First, PM practice can be a consistent, organized, or even standardized approach. Our framework helps project managers and members to approach PM in a more systematic way. With the help of the framework, project managers can visualize what aspects of projects need to be considered and what kind of consistent practice can be created and followed for these aspects. If the project teams uses an information system with all of the four major PM functions: basic PM, KM, process management, and communication and collaboration, then project members can use PM tool to schedule tasks and milestones, can use process management tool to enforce management of task dependency, workflow, change and day to day operations, and can use communication and collaboration tool for group meeting, discussion and problem solving. All documents, artifacts, meeting minutes and communication text produced by these tools can be archived in the repository - organized, associated, and indexed for fast retrieval and update. Project members can search the repository to retrieve all documents related to a certain topic, such as a milestone. The documents may include project budget and schedule, its constituent tasks, resources and personnel assigned to the tasks, routine reports of the tasks, changes to the tasks and resources, decision rationales, and meeting minutes. By using such a collaborative PM system, project members can do a more efficient and effective job of seeking information, writing reports, collecting project performance matrices, analyzing projects' well being, and detecting mistakes early for correction.

Second, using a system developed with this framework will increase project learning or organizational learning. Project information is organized and new members can quickly get themselves familiar with the project by searching and browsing information about the project. Turnover of project members may not be a big issue as it used to be. Project members may also save time by re-using some project items (e.g., template for report and meeting minutes) archived in the project repository. Additionally, members of future projects can learn best practices from past projects and avoid repeating similar mistakes. Using a number of PM systems to deliver PM functions can also facilitate PM and project learning to a certain degree; however, they lack the functionality of collaborative software in terms of accessibility, accuracy, flexibility and efficiency. Project managers may have to search different places for information about a project item; the search can be time-consuming, and members may not be aware of all of the places that store project information. Project members may also get duplicate, outdated information or conflicting information if it is stored in different places and by using different software/system.

Using Web-based project collaboration system is on the rising trend. However, adoption of the collaborative PM system is not a common practice; this type of system is very useful for the high-value, long-duration project work that stresses collaboration (Sawyer, 2004). Some construction companies started to use collaborative PM system to manage their projects, be able to increase their teamwork efficiency and effectiveness, and require their subcontractors to use the system as well (Sawyer, 2004). Project practitioners should actively implement consistent process to manage their projects and utilize information systems to support their effort. Our framework helps practitioners to get a holistic view of the PM effort and select the PM systems accordingly.

Future Research

This framework may initiate more research in PM. For researchers that are more interested in technology, they may want to address the following questions: What might be the appropriate system architecture under which to build collaborative PM systems? What is the right method to provide interfaces among different functional features? What would be the format for the system output so that the output could be utilized by other popular software packages? For researchers that are more interested in the impact of IT on organizations and teams, they may want to know answers to the following questions: How can project processes be streamlined and standardized through collaborative PM systems? What types of PM systems are most frequently used by project teams? How "collaborative" are these systems when compared with our framework? What are the issues when project teams use these systems? If these systems need to be improved, then does our framework provide some insights into the system improvement? Usually project teams use a number of software packages to deliver PM functions; what usage rules or norms can project members adopt to increase the data and information flow across packages? Lastly, how is the quality of information controlled and ensured? We believe that answers to these and similar questions will help project researchers and practitioners further PM research and improve PM practice.

APPENDIX

List of Example PM Systems

Product Name	Web site	Software type	Application /Industry
ActiveProduct	Frametech.com	PM	General
Asta Powerproject	Astadev.com	PM	Construction
Bizwall	Bizall.com	PM & process mgmt.	General
BPMS	Bpms.com	PM	General
Cognito	GroupSystems.com	Collaboration	General
Concurrent Version Systems (CVS)	Open source cvshome.org	Doc. Mgmt.	General
Critical Path Suites	Cpts.com	PM & process mgmt.	General
Defect Manager	Tierasoft.com	Process mgmt.	Software
Documentum Enterprise Document Management (EDM)	Documentum.com	Doc. Mgmt.	General
EPM.Ensemble	Inventx.com	PM	Software
Eroom	Documentum.com	Some PM, Collaboration	General
Facilitate.com 8.0	facilitate.com	Collaboration	General
GigaPlan	Gigaplan.com	PM & Doc. mgmt.	General
Hummingbird Document Management	hummingbird.com	Doc. mgmt.	General
IBM Lotus Domino Doc Manager	Lotus.com	Doc. Mgmt.	General
IBM Lotus SameTime	Lotus.com	Collaboration	General
MatrixOne Document Central	MatrixOne.com	Doc. mgmt.	General
Microsoft Visual SourceSafe (VSS)	Microsoft.com	Doc. Mgmt	Software
MS Project 2002	Microsoft.com	PM	General
OnProject	onproject.com	PM & process mgmt.	General
Project Center	Bricsnet.com	PM & process mgmt.	Construction
Projistics	nagarro.com	PM, process support & KM	General
Samepage eStudio	Samepage.com	Mainly PM, some communication and process support	General
Teamcenter project	Eds.com	Mainly PM, and some doc. Mgmt., collaboration support	General
Teamspace	Teamspace.com	PM, collaboration, some doc. Mgmt.	
WebEx	webex.com	Collaboration	General
Windchill ProjectLink	PTC.com	Doc. Mgmt. & work flow	General

REFERENCES

Distributed Project Management: A Marketplace and Software Vendor Analysis. (2004). Retrieved January 15, 2005, from http://www.collaborate.com/announcements/announce_3.html

Abramovici, M., & Gerhard, D. (2000). A Flexible Web-Based PDM Approach to Support Virtual Engineering Cooperation. Proceedings of the 31st Annual Hawaii International Conference on Systems Sciences, Hawaii, IEEE Computer Society Press.

Augustine, S., Payne, B., Sencindiver, F. & Woodcock, S. (2005). Agile Project Management: Steering from the Edges. *Communications of the ACM 48*(12): 85-89.

Back, W. E. & Moreau, K. A. (2001). Information Management Strategies for Project Management. *Project Management Journal 32*(1), 10-20.

Baek, S. & Liebowitz, J. (1999). Designing a Web-Based Knowledge Repository in a Virtual Team and Exploring Its Usefulness. Proceedings of the Fifth Americas Conference on Information Systems, Milwaukee, Wisconsin.

Balk, L. D. & Kedia, A. (2000) PPT: A COTS Integration Case Study. Proceedings of the 2000 International Conference on Software Engineering (ICSE), p.42-49, Limerick, Ireland.

Bourgault, M., Lefebvre, L., Lefebvre, L. A., Pellerin, R. & Elia, E. (2002). Discussion of Metrics for Distributed Project Management: Preliminary Findings. Proceedings of the 35th Haiwaii International Conference on System Sciences, Hawaii, IEEE Computer Society.

Chan, S.L. (2004). Prototype Web-Based Construction Project Management System. *Journal of Construction Engineering and Management 130*(6): 935-944.

Chen, H.G., Jiang, J. J., Chen, J.C. & Shim, J. T. (2004). The Impacts of Conflicts on Requirements Uncertainty and Project Performance. *Journal of International Technology and Information Management 13*(3): 157-167.

Clarke, A. (1999). A Practical Use of Key Success Factors to Improve the Effectiveness of Project Management. *International Journal of Project Management 17*(3), 139-145.

Cleetus, K. J., Cascaval, G. C. & Matsuzaki, K. (1996). PACT - A Software Package to Manage Projects and Coordinate People. Proceedings of the 5th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE'96), Stanford University, CA, IEEE Computer Society.

Coleman, D. (1997). Collaborating on the Internet and Intranets. Proceedings of the 30th Hawaii International Conference on System Sciences, Hawaii, IEEE Computer Society Press.

Dixon, D. (1988). Integrated Support for Project Management. Proceedings of the 10th International Conference on Software Engineering, Singapore.

Evaristo, R., & Fenema, P. C. V. (1999). A Typology of Project Management: Emergence and Evolution of New Forms. *International Journal of Project Management 17*(5), 275-281.

Graham, R. J. & Englundm, R. L. (1997). Creating an Environment for Successful Projects. San Francisco, CA, Jossey-Bass Inc.

Gregory, T. & Briggs, R. O. (2002). An Approach to Middleware for Repeatable Collaborative Processes. Proceedings of the 35th Annual Hawaii International Conference on Systems Sciences, Hawaii, IEEE Computer Society Press.

Hefner, R. (2000). Managing Projects through a Corporate Repository. Proceedings of the 33rd Annual Hawaii International Conference on Systems Sciences, Hawaii, IEEE Computer Society Press.

Huang, G. Q., Feng, X. B. & Mak, K. L. (2001). POPIM: Pragmatic Online Project Information Management for Collaborative Product Development. Proceedings of the Sixth International Conference on Computer Supported Cooperative Work in Design, London, Ontario, Canada

Jaafari, A. & Manivong, K. (1998). Towards a Smart Project Management Information System. *International Journal of Project Management 16*(4), 249-265.

Jonsson, N., Novosel, D., Lillieskold, J. & Eriksson, M. (2001). Successful Management of Complex, Multinational R&D Projects. Proceedings of the 34th Hawaii International Conference on Systems Sciences, Hawaii, IEEE Computer Press.

Katzy, B., Evaristo, R. & Zigurs, I. (2000). Knowledge Management in Virtual Projects: A Research Agenda. Proceedings of the 33rd Hawaii International Conference on System Sciences, Hawaii, IEEE Computer Society.

Kingman, L. C., Lambert, R. E. & Steen, R. P. (1990). Operational Image Systems: A New Opportunity. *IBM Systems Journal 29*(3), 304-312.

LaBrosse, M. (2004). Project Management in the Real World. *Plant Engineering 58*(11): 29 - 31.

Lam, H. E. & Maheshwari, P. (2001). Task and Team Management in the Distributed Software Project Management Tool. Proceedings of the 25th Annual International Computer Software and Applications Conference (COMPSAC.01), IEEE Computer Society.

Lundin, R. A. and F. Hartman (2000). Pervasiveness of Projects in Business. In P*rojects as business constituents and guiding motives*. R. A. Lundin and F. Hartman (Eds.), Norwell, Massachusetts: Kluwer Academic Publishers**,** 1-10.

Ly, E. (1997). Distributed Java Applets for Project Management on the Web. *IEEE Internet Computing*: May - June 1997.

Lysakowski, R. & Doyle, L. (1998). Electronic Lab Notebooks: Paving the Way of the Future in R&D. *Records Management Quarterly 32*(2), 23-30.

Mahaney, R. C. & Greer, B. M. (2004). Examining the Benefits of Project Management Professional (PMP) Certification for IS Project Managers and Organizations. *Journal of International Technology and Information Management 13*(4): 263-273.

Marttiin, P., Lehto, J. A. & Nyman, G. (2002). Understanding and Evaluating Collaborative Work in Multi-Site Software Projects - A Framework Proposal and Preliminary Results. Proceedings of the 35th Annual Hawaii International Conference on Systems Sciences, Hawaii, IEEE Computer Society Press.

Maurer, F. (1996). Working Group Report on Computer Support in Project Coordination. Proceedings of the Project Coordination Workshop of the IEEE Fifth Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise (WET ICE), Stanford University, CA, IEEE Computer Press.

McManus, D. J. & Snyder, C. A. (2003). Knowledge Management: The Role of EPSS. *Journal of International Technology and Information Management 12*(2): 17-28.

Nidiffer, K. E. & Dolan, D. (2005). Evolving Distributed Project Management. *IEEE software 22*(5): 63-72.

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

Nunamaker, J. F., Jr., Romano, N. C., Jr. & Briggs, R. O. (2001). Increasing Intellectual Bandwidth: An Integrated Framework of KMST and CST. Group Decision and Negotiation Conference 2001, La Rochelle France, Faculty of Technology, Policy and Management. Delft University of Technology.

Post, G. V. & Kagan, A. (2005). Systems Development Tools and the Relationship to Project Design: Cost and Budget Implications. *Journal of International Technology and Information Management 14*(1): 1-13.

Potter, R. E. & Balthazard, P. A. (2000). Cross-Cultural Issues in Virtual Team Support: Communication Characteristics and Task/Technology Perceptions from Mexican and U. S. Team Members. *Journal of International Information Management.9*(2): 1-15.

Romano, N. C., Jr., Chen, F. & Nunamaker, J. F., Jr. (2002). Collaborative Project Management Software. Proceedings of the 35th Hawaii International Conference on Systems Sciences, Hawaii, IEEE Computer Press.

Sawyer, T. (2004). Online Management Tools Excel at Empowering Project Teams. *ENR 253*(14): 24-27.

Sena, J. A. & Shani, A. B. (1999). Intellectual Capital and Knowledge Creation: Towards an Alternative Framework. In *Knowledge management handbook*, J. Liebowitz (Ed.). CRC Press, Boca Raton, FL., 8-1 - 8-16.

Teicholz, P. & Fisher, M. (1994). Strategy for Computer-Integrated Construction Technology. *Journal of Construction Engineering and Management, ASCE 120*(1): 117-131.

Turner, J. R. (2000). Do You Manage Work, Deliverables or Resources? *International Journal of Project Management 18*(2), 83-84.

Weiser, M. & Morrison, J. (1998). Project Memory: Information Management for Project Teams. *Journal of Management Information Systems 14*(4), 149-166.

Whittaker, J. (2000). Reflections on the Changing Nature of Projects. In *Projects as business constituents and guiding motives*, R. A. Lundin and F. Hartman (Eds.), Norwell, Massachusetts: Kluwer Academic Publishers.

Wu, F., Chu, L. K., Li, H. Z., Han, X. M. & Sculli, D. (2003). The Virtual Organizing Process – A Critical Tool for Enterprise Competitiveness in the Information Era. *Journal of International Technology and Information Management 12*(1): 25-42.