

Knowledge Sharing Tools for IT Project Management

Stacie Petter

Georgia State University, USA

Vijay Vaishnavi

Georgia State University, USA

Lars Mathiassen

Georgia State University, USA

INTRODUCTION

Information technology (IT) project disasters make worldwide headlines, and billions of dollars have been lost due to poor project implementations. The Standish Group, a research advisory firm, reports that only one-third of the over 13,500 IT projects evaluated in 2003 were successful, and half of the reported IT projects were classified as challenged, meaning they experienced cost and budget overruns (Larkowski, 2003). While the state of IT project management is improving, organizations must explore ways to reduce unnecessary spending that occurs because of failures, cost and schedule overruns on IT projects. One possibility is to improve knowledge sharing to avoid repeating mistakes and to build on successes from the past.

BACKGROUND

IT project management is demanding because of time pressure, restricted capital, and high degrees of uncertainty during projects and is comprised of complicated and ill-structured problems (Grupe, Urwiler, Ramarapu, & Owrang, 1998). However, valuable knowledge gained before, during, and after the completion of projects is rarely captured, shared, and utilized in future projects. As a result, projects suffer from reinvention of solutions, repetition of mistakes, and loss of process knowledge after project completion. These problems are further exacerbated by the turnover of project managers and the lack of technologies that effectively integrate relevant knowledge with existing project management software (Tiwana & Ramesh, 2001).

Knowledge is a combination of experience, values, contextual information, and insight used to create, absorb, and evaluate new experiences and information (Davenport & Prusak, 2000). Project managers rely on past experiences to make decisions that keep the project within schedule, budget, functionality, and quality targets; however, these

experiences are rarely shared among project managers (Schindler & Eppler, 2003). Furthermore, a problem faced by many IT project managers is their own lack of experience as an IT project manager. Individuals are often promoted to an IT project manager position because of their superior programming skills. However, these experiences alone are not enough to guarantee success as an IT project manager (Standish Group, 2001). Fortunately, a variety of knowledge sharing tools can help inexperienced project managers acquire relevant knowledge. Even for IT project managers with extensive knowledge, such tools provide good opportunities to learn from others when confronted with a unique problem (Newell, 2004).

TOOLS FOR SHARING KNOWLEDGE

Many tools have been developed to assist IT project managers in avoiding project failures, including post-mortem analysis, knowledge management systems, and networking. Rather than focusing on specific tools for sharing knowledge, this section describes generic classes of knowledge sharing tools available to IT project managers. Each tool is described in terms of *what* type of knowledge is shared (i.e., available in documented form or emergent through interaction), *who* is the primary user of the tool (i.e., project manager or entire project team), *where* knowledge is shared within the organization (i.e., between individuals or organization-wide), and *why* the tool is used for sharing knowledge (i.e., exploitation of existing knowledge or a basis for exploration of new knowledge).

Post-Mortem Analysis

Post-mortem analysis is supported by a process and a series of documents to identify successes and failures for a given project (Sinofsky & Thomke, 1999). Good post-mortem analyses not only record the history of the project itself,

but also provide information on what went wrong during specific phases of the project's life cycle (Thomke & Fujimoto, 2000). Often, organizations only conduct post-mortem analyses on projects that have been abandoned or have failed (Esusi-Mensah & Przasnyski, 1995); however, there are benefits to conducting post-mortem analyses on successful IT projects. Organizations that perform post-mortems state that the knowledge gained is useful in avoiding repetition of past mistakes, improving processes on future projects, providing historical accounts of what went wrong, and enhancing performance on future projects (Esusi-Mensah & Przasnyski, 1995).

The output of a post-mortem analysis is a series of documents. These documents can be shared across the entire project team as they articulate the aspects of a project that were successful and the areas needing further improvement. Although the results of post-mortems can benefit the entire project team, a survey of post-mortem analyses found that IS managers, system developers, and new IT project managers were more likely to consult these documents than other groups such as programmers, senior management, and other functional managers (Esusi-Mensah & Przasnyski, 1995). Post-mortem analyses are shared across the organization, frequently through the use of new processes or management practices. The knowledge gained through post-mortem analyses is exploited throughout the organization to minimize the repetition of problems, better plan or manage new projects, and guide the development of new management procedures (Esusi-Mensah & Przasnyski, 1995).

KNOWLEDGE MANAGEMENT SYSTEMS

Within the knowledge management literature, there are three types of systems to create and share knowledge: codified, personalized, and collaborative systems. While codified knowledge systems transform knowledge into documented form, a personalized system helps people interact to create and communicate knowledge (Hansen, Nohria, & Tierney, 1999). Codified knowledge systems are basically shared databases; personalized knowledge systems encourage interaction. Collaboration systems combine personalized and codified knowledge and focus both on supporting interaction among colleagues and providing a repository to share knowledge that emerges through the collaboration.

Codified knowledge systems store knowledge in databases where it can be accessed and used easily by anyone in the organization, making knowledge available to all members of a project team and across project teams (Hansen et al., 1999). As a result, all project team members, from developer to project manager, can benefit from the codified knowledge. The purpose of developing and maintaining a codified knowledge system and populating it with knowledge is to disseminate knowledge throughout the organization (Hansen et al., 1999). By investing in codified knowledge and a tool

to effectively share the stored knowledge, organizations can exploit knowledge obtained and captured during prior IT projects on future projects.

Organizations use codified knowledge systems in an effort to prevent knowledge from leaving the company by relying on technology for the storage of knowledge. Because IT projects are vulnerable to personnel changes, codified knowledge systems are created to exploit knowledge from past projects. Because knowledge is documented and stored within a shared system, temporal and geographical differences between the knowledge-seeker and knowledge-holder are irrelevant. The primary problem with these knowledge sharing systems is that the knowledge captured and stored often goes unused. The reason is that IT project managers must acknowledge that there is a need to seek out additional knowledge and this knowledge must be clearly communicated across project teams (Newell, 2004). Successful utilization of codified knowledge systems requires individuals and teams to seek out available knowledge as part of their project management activities.

Personalized knowledge systems help those seeking knowledge find people who have the needed knowledge or who can help create that knowledge (Davenport & Prusak, 2000). These systems are useful when the necessary knowledge is not easily documented; therefore, brainstorming sessions or conversations between an expert and the knowledge seeker are needed to provide the necessary knowledge (Hansen et al., 1999). The complex nature of IT project management suggests that some problems are better suited to this interactive method of knowledge sharing. A key limitation to this approach is that knowledge does not remain behind to benefit others when experts leave the company (Hansen et al., 1999).

Personalized knowledge systems provide a map or listing of individuals in the organization who possess different types of knowledge (Davenport & Prusak, 2000). Knowledge sharing is based on personal interaction rather than on explication of the knowledge to be shared (Hansen et al., 1999). All members of a project team, including the IT project manager, developers, and others, are able to solicit advice from those with expertise in a given subject matter. The purpose of developing and maintaining a personalized knowledge system, or knowledge map, is to communicate expertise across the organization. Personalized knowledge systems are often used for brainstorming and addressing unique problems that arise during a project (Hansen et al., 1999), and the reason for adopting these systems is to create new knowledge through the combination of existing personal insights.

Collaboration systems enable groups of people to share information, communicate, coordinate, and work together as a team (Lamont, 2004). These systems enable team members that may be geographically dispersed to work and share knowledge within the team. Incorporating collaboration sys-

4 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/knowledge-sharing-tools-project-management/13915

Related Content

E-Logistics: The Slowly Evolving Platform Undrepinning E-Business

Kim Hassall (2009). *Encyclopedia of Information Science and Technology, Second Edition* (pp. 1354-1360). www.irma-international.org/chapter/logistics-slowly-evolving-platform-undrepinning/13752

Organizational Knowledge Management

David B. Paradise and James F. Courtney (1989). *Information Resources Management Journal* (pp. 1-14). www.irma-international.org/article/organizational-knowledge-management/50916

Configuring Software-as-a-Service (SaaS) Enterprise Systems to Mitigate Idiosyncratic Risk

Yuanyuan Chen, Annie Tian and Matthew L. Jensen (2025). *Journal of Information Technology Research* (pp. 1-27). www.irma-international.org/article/configuring-software-as-a-service-saas-enterprise-systems-to-mitigate-idiosyncratic-risk/393632

Intelligent Information Systems

John Fulcher (2009). *Encyclopedia of Information Science and Technology, Second Edition* (pp. 2118-2125). www.irma-international.org/chapter/intelligent-information-systems/13871

Mobile Technology

Paul Cragg and Prue Chapman (2001). *Annals of Cases on Information Technology: Applications and Management in Organizations* (pp. 169-178). www.irma-international.org/article/mobile-technology/44614