

A Case Study of Wiki-based Experience Repository at a Medium-sized Software Company

Thomas Chau

Department of Computer Science
University of Calgary
Calgary, Alberta, Canada
tsychau@gmail.com

Frank Maurer

Department of Computer Science
University of Calgary
Calgary, Alberta, Canada
maurer@cpsc.ucalgary.ca

ABSTRACT

The paper reports on an exploratory case study where an organizational learning methodology was used to support knowledge sharing in a medium-sized distributed software development company. The method is based not only on a codification approach but also on a personalization strategy. To facilitate its use in a dynamic environment with minimal overhead, our MASE system utilizes both informal and formal knowledge representation mechanisms. The results provide evidences for the need of knowledge sharing tools (1) to incorporate not only codification-oriented repository technologies but also those that facilitate communication and collaboration among people, and (2) to support not only structured but also unstructured knowledge representation mechanism. It reveals that an informal knowledge authoring tool such as the Wiki-based MASE is used for sharing content for problem understanding, instrumental, projective, social, expertise location, and content navigation purposes. We also observe self-organized maintenance of the repository content among the ordinary repository users as a result of the open-edit nature of MASE.

Keywords

Organizational Learning, Authoring Tools, Experience Reports.

INTRODUCTION

This paper presents results of an exploratory case study conducted at a medium-sized software consulting company on the use of MASE—a knowledge sharing tool based on Wiki [7] technology—for facilitating organizational learning.

The objective of our work on MASE is to address the need for an improved mechanism of sharing experiences among multiple software development teams in an organization. Such need is echoed by recent studies indicating that software development teams are often prone to repeat past mistakes [4] or duplicate effort in “re-inventing the wheel” as they are not aware of experiences acquired or successful solutions adopted from other teams within the same organization [5]. The significance of this organizational learning problem is that an organization’s inability to leverage existing experiences and generate new knowledge will have a

negative impact on its competency, productivity, and product quality.

In software engineering, an approach to address this organizational learning challenge is the Experience Factory framework [3]. In essence, this framework only describes what knowledge engineering tasks—experience elicitation, analysis, generalization, formalization, packaging, customization, and dissemination—need to be done, but it does not prescribe how they are to be done. Hence, different software organizations adopt the Experience Factory framework differently depending on their business needs and organizational constraints.

Based on reported practitioners’ experience, two general strategies to organizational learning have been identified [12]. The codification strategy is characterized by processes that transform employees’ tacit experiences to an explicit and structured form like a document, and by tools that connect employees to these documents. Recently, semantic net initiatives follow that strategy. In contrast, the personalization strategy is characterized by processes and tools that promote knowledge sharing via person-to-person contacts; connecting employees to one another rather than to knowledge in its explicit forms. The codification strategy has been found to increase knowledge re-use achieving a leverage effect, while the personalization strategy has been found to help generate new knowledge achieving an innovation effect [22].

In software engineering, most of the existing literature on knowledge management primarily focuses on implementations of the Experience Factory concept in the form of knowledge repositories [8]. These repositories vary in the technologies used for experience elicitation and analysis [17], formalization [18], and dissemination [1], but most of them share in common in that they implement the Experience Factory concept based on the codification strategy.

This is usually achieved by establishing a centralized experience factory group that collects development, technology and product knowledge, generalizes it, and packages it for reuse by other teams. Access to these packaged experiences is often provided via knowledge repositories that are maintained by the experience factory group. Since the experience factory group aims to generalize as much project experiences as possible for re-use, to bootstrap or to update

the content in an experience repository need to undergo a controlled and often slow process.

In addition, most reports on experience factories stem from large organizations that can afford to create centralized organizational units for organizational learning [2,20,14]. However, such resources may not be available to small- and medium-sized software companies.

Hence, it is our opinion that small- and medium-sized development organizations need more lightweight means of creating these knowledge bases—ways that allow all developers to create, update and reuse experiences with minimal overhead—especially in a dynamic environment where customer requirements change frequently and unpredictably, and new technologies constantly emerge.

There also exist implementations of the Experience Factory concept based on the personalization strategy but they are, in comparison to the codification approach, few and far in between. At Ericsson [16], some employees act as “experience brokers” to mediate referrals to individuals with the expertise needed by the knowledge seekers. Although experience repositories are also consulted by the individuals, more often it is during the personal interactions that expertise is elicited, analyzed, generalized, customized, and shared. Likewise, personal contacts are heavily relied upon and are made possible by post-mortem review workshops in [9] to facilitate expertise sharing.

Despite the substantial number of Experience Factory implementations cited in the literature, there are few evaluations on the use of such knowledge repositories. Of the few experience reports and case studies, the study contexts are often in large organizations such as NASA [2], Daimler Chrysler [20,14], and Ericsson [16], or the knowledge repositories studied are only based on either the codification strategy [2,20,14] or the personalization strategy [16] but not both.

In comparison, the context in our case study is in a medium-sized software organization where teams are geographically distributed. Furthermore, our case study evaluates the use of a knowledge repository (MASE) that supports the Experience Factory concept by using both the codification and personalization strategies.

TOOL UNDER STUDY - MASE

The decision that both strategies are used in MASE to implement the Experience Factory framework is based on the empirical findings suggesting that various processes and tools supporting these two strategies can [22] and ought to [15] co-exist in an organization.

To support the use of the codification strategy, MASE makes use of Wiki technology to allow users to annotate information in an *informal* and *unstructured* fashion. Wiki technology enables any users to access, create, organize, and update content on web pages in real-time using only a web browser. To change these editable web pages (“wiki

pages”), users apply the Wiki markup language, which is much simpler than HTML.

MASE also supports the use of *structured* information content by providing a plug-in API that allows tool developers to easily develop plug-ins. A MASE plug-in contains additional functionalities that extend the capability of the original Wiki implementation. It either presents (1) an input form that allow users to submit data following specific schemas or (2) a table that displays information retrieved from databases in a structured fashion. Users can include any MASE plug-ins on any wiki page simply by referencing the names of the plug-ins in the wiki pages. This capability and the fact that any content on any wiki pages can be modified by anyone gives users the flexibility to design not only the information content but also the navigational structure of MASE.

To facilitate increased re-use of task-specific knowledge, MASE provides a set of project management plug-ins which project members can use to define project tasks and store information about them according to specific formats. Since these project and task information can be retrieved via plug-ins and can be stored in any wiki pages, project members are free to annotate additional information, in free-formatted text, about the specific tasks that they are working on if they find the default set of input fields provided by the plug-ins inadequate.

To retrieve the posted information content, MASE provides full-text search capabilities on any unstructured and structured data. This allows users to use natural language to specify their search queries.

To support the use of personalization strategy and to realize its intended benefit of facilitating knowledge creation, we concur with Computer Supported Cooperative Work (CSCW) experts [11] that knowledge sharing and creation is inherent in day-to-day collaboration and tools that enhance collaboration will naturally support knowledge sharing and creation. To enhance collaboration, MASE supports various styles of collaborative work.

First, MASE supports asynchronous collaboration by letting users to store the state of any wiki pages at any time.

Second and more importantly, MASE also enables users to collaborate synchronously. This is made possible by a MASE plug-in that integrates with Microsoft NetMeeting and another plug-in that provides awareness information by displaying all users who are currently connected to MASE. Whenever a user connects to MASE, the IP address of that user’s computer is recorded. By viewing that online user’s personal profile, one can start a NetMeeting session with that user. These two plug-ins made it possible for users to be aware of each other’s presence and facilitate informal and spontaneous interaction, which is critical for sharing tacit knowledge [10].

Besides providing a collective space for organizational knowledge, MASE also provides each user with a wiki page that can serve as a personal information space for

him/her. Like any other wiki pages, any content can be stored in these personal wiki pages. As such, this allows users to have complete control over the structure and the type of information that they deem relevant to themselves only.

More details about MASE can be found in [6]. We now will describe our exploratory case study in more detail.

STUDY OBJECTIVES

The objectives of this case study are both exploratory and descriptive in nature with the following research goals:

1. To understand how developers use an informal knowledge capture and sharing tool for sharing their experiences;
2. To elicit the challenges, enabling factors, and perceived benefits of using an informal knowledge sharing tool for knowledge sharing; and,
3. To gather feedback on our tool's usability and functionalities.

For the first goal in particular, the specific research questions that we attempt to address via the case study are:

1. How do users collaborate with each other, given the tool's flexibility in supporting real-time and asynchronous work?
2. What types of knowledge (e.g., task-related, team-related, personal, social, etc.) are being shared?
3. What are the characteristics of the users with respect to the information that they use?
4. How do users use the information content, given the tool's flexibility in supporting the use of structured and unstructured information?
5. To what extent do users self-organize among themselves in maintaining the information content given the open-edit nature of the tool?

STUDY CONTEXT

The software company involved in this case study is empolis, which specializes in producing software for knowledge management support. As such, its domain expertise gives empolis unusually keen insight into its own organizational learning requirements. Empolis consists of employees who are distributed across four offices in Germany, as well as offices in the UK, Poland, Hungary and Dubai. This distribution of labor and resources presents unique challenges in how knowledge is represented and shared.

At empolis, several small teams work on three historically grown product lines of software that share some commonalities. As such, these teams operate relatively independently of one another on different but often related projects and core products. For example, one team may be working on a new product release, while another team works on customizing the previous release for a customer. Yet another team may be engaged in research and new product development. Depending on the project, teams consists of varying stakeholders—customer-specific project teams

include marketing personnel, while new development teams consists mostly of programmers.

Initially, there was no way for these teams to share experiences in a consistent and recognized way. Occasions for face-to-face interactions had been set up as much as possible but the benefits of these efforts were limited due to the distributed team environment. To overcome this constraint, some teams had taken matters into their own hands and used personal web sites, web logs, and even a wiki for representing artifacts and information among members of single teams. Knowledge contained in these ad-hoc knowledge repositories include best practices, lessons learned, technical data, artifacts used and created by the teams, and even some personal information shared among more social developers. These existing mechanisms were not applied across project or team boundaries and were not recognized between empolis offices in different cities. In addition, the user base of these repositories often was entirely made up of software developers, and did largely not include other stakeholders or employees. Teams with better knowledge management tools were recognized at empolis to produce better results. Aware of these practices and their effects, the upper management at empolis deemed a more widely adopted organizational learning solution as a necessity.

Empolis' interest in MASE was chiefly in its lightweight, flexible approach to the Experience Factory concept, and in its ability to support the personalization strategy that is lacking in other knowledge management tools, including those of empolis', which focus only on the codification strategy. Goals at empolis included both gaining from the benefits of knowledge management in practice and gathering experience with such lightweight tools to help with future development of their own knowledge management solutions.

PARTICIPANTS & APPARATUS

As for the demographics of MASE users at empolis, 80 employees from 8 teams self-registered for MASE accounts. About 10% of the total registered users are managers and technical writers with the remaining 90% of users being developers.

Design and installation of the MASE system at empolis was largely driven by a specific vision communicated by the CEO, based on specific examples he had observed and elicited from existing teams and some of his own ideas. The primary objective was to create a company-wide wiki for internal use, allowing teams to self-organize new knowledge after some resources and time were expended entering existing important information into the system. Bottom-up support was initially varied, with some teams and individuals being highly enthusiastic and participatory, and others unsure of the value that any top-down system would bring them. To advertise the MASE presence, the empolis CEO offered a reward to users who contributed content to MASE in the early days. As of mid August

2004, no more explicit incentives were used to motivate contribution to MASE.

After MASE was installed and while it was being used, we kept a wiki page of bugs and feature requests from users. One critical feature added to MASE at the request of empolis was a private messaging system to allow users to communicate privately within MASE rather than using email, as well as to receive in-browser notification of wiki page changes. The private messaging mechanism works very similar to the text-chat facilities in online chat software such as MSN Messenger. While the private messaging feature was fairly simple to implement, informal feedback from users indicate that this feature was critical to making many of the users feel as though they were communicating and collaborating rather than simply generating documents.

Another major feature requested by the CEO was a way to examine the users' participation level in MASE. This was in order to see how quickly MASE was being adopted, from what offices or projects, and with what kinds of data.

DATA COLLECTION & ANALYSIS

A partnership between empolis and the University of Calgary allowed a researcher (not the authors) to work on-site at an empolis office over two three-week periods, installing and customizing MASE to meet their knowledge management needs (e.g., configuring MASE to store content in a file system instead of in a source code configuration management server). Installation of MASE was done at a single office in Kaiserslautern, Germany, however the presence of an organization-wide intranet means that teams and individuals from all empolis offices have access to MASE.

Data for this case study has been collected over a three-month period, starting at the end of May 2004 and ending in mid August 2004. The observations come mainly from static analysis of logs generated by MASE and from informal feedback given by the empolis representatives.

For data analysis, we constructed a database with information from the system logs and from responses to the questionnaires. We inspected the logs manually and applied a simple categorization of tool access into: read, write, search, and synchronous interactions; and of the content in each wiki page into: problem understanding, instrumental, projective, social, and content navigation aid. The categorization scheme of the wiki page content is based on work done in the information science field [21]. In his work, Taylor defines problem understanding as using information to increase comprehension of a problem; instrumental as using information to follow guidelines or procedures; projective as using information to make forecasts or scenarios; and, social as using information to develop relationship that can be personal and/or political in nature. Unlike the other four categories, the category of content navigation aid is specific to MASE. This is because MASE allows users to collaboratively author content freely; as such, some wiki

pages may be authored specifically to guide users to find particular types of content.

OBSERVATIONS

To answer the research question of how project teams at empolis make use of MASE to collaborate with each other, we analyzed the various ways that users accessed information. The result (Table 1) suggests that the knowledge sharing tool is primarily used for asynchronous collaboration. It also indicates that project teams at empolis use MASE primarily for retrieving information content and not as much for collaboratively authoring information content. With respect to information retrieval patterns, the result indicates that users prefer to browse to the desired content rather than using the search engine.

Table 1. How users use MASE to collaborate with each other

| Types of Access | # of Use | % of Use |
|-------------------------|----------|----------|
| Read | 10,400 | 80.88 |
| Write | 4,757 | 13.66 |
| Search | 509 | 3.96 |
| Synchronous Interaction | 193 | 1.50 |

To answer the question of what types of knowledge can be found in a knowledge repository in a software development setting, we analyzed and categorized all the wiki page content in MASE. The analysis reveals that users make use of wiki pages in MASE to:

- Exchange ideas for solution development;
- Record decisions made and their rationales during meetings;
- Following instructions from answers to FAQs;
- Share social information like the internal dart throwing competitions;
- Identify experts in particular parts of the products or development processes;
- Self-coordinate themselves for collaborative work; and,
- Track progresses on project tasks.

These results are summarized in Table 2. It shows that nearly 50% of the use of MASE was on exchanging ideas on domain and technical problems. In fact, Figure 1 shows that 80% of all the read-accesses to information content were concentrated to just over 20% of the pages (170 out of 815 pages) and about 25% of the read-accesses were to the top 10 most visited pages.

Table 2. Types of knowledge found in an informal software development knowledge repository

| Types of Content | Pages | Read Access | Write Access | Overall Access |
|-----------------------|-------|-------------|--------------|----------------|
| Problem Understanding | 46 % | 40 % | 8 % | 49 % |
| Content Navigation | 16 % | 39 % | 3 % | 42 % |

| Aid | | | | |
|--------------------|------|------|-----|------|
| Instrumental | 43 % | 30 % | 6 % | 36 % |
| Social | 28 % | 32 % | 2 % | 35 % |
| Projective | 29 % | 18 % | 3 % | 21 % |
| Expertise Location | 18 % | 19 % | 1 % | 20 % |

Distribution of Read-Access to Information Content

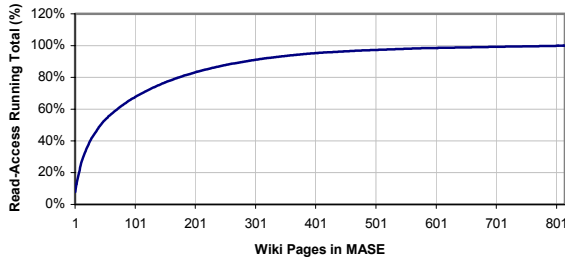


Figure 1. How read-accesses are distributed in a software development knowledge repository

Of these 10 most visited wiki pages, 8 of them were pages not bundled with the initial MASE installation but were created by project teams at empolis spontaneously to share knowledge unique to their project or team. Of the other 2 pages, the “RecentChanges” wiki page lists all pages that have been updated over a certain period of time as specified by users; the “ListUsers” wiki page lists contact information for other MASE users, and provides the means to contact and interactively message those users.

To answer the question of to what extent users would formalize or structure the information content in a software development knowledge repository¹, the analysis in Table 3 shows that over half of the content is annotated in purely unstructured form with the remaining half annotated in either semi-structured or totally structured format. In particular, 80% of the structured content was generated by 6 of the 34 plug-ins and these 6 plug-ins were used to:

- Aid navigation by finding the contexts of the currently viewed page in relation to other pages;
- Embed or retrieve content from other online resources such as the Java API;
- Rate wiki pages on the quality of the posted content;
- Receive e-mail notifications of changes made to any subscribed wiki pages; and,
- List the number of times a particular wiki page has been read.

¹ The knowledge structuring mechanisms in MASE are very limited compared with ontological approaches. MASE provides a fixed set of concepts (including tasks, users, process types, process decompositions and others). This set is not extensible (without writing code) by the users of the system. Our goal was not to evaluate strong ontological mechanisms (as being used by semantic net approaches) but to determine how far lightweight approaches can reach in our application domain.

Table 3. Users' preference for formalizing repository content

| Formalization of Content | Pages (%) |
|--------------------------|-----------|
| Unstructured | 58 |
| Both | 23 |
| Structured | 19 |

With respect to content contribution, employees at empolis created about 12 times (761/54) as much new wiki pages as the default wiki pages that were bundled in MASE when the case study began. As for collaborative authoring of information content in MASE, about 7% of the pages (60/815) were contributed by 2 or more authors.

To answer the question of whether there exists any self-organized maintenance of information content in the knowledge repository given the open-edit nature of MASE, we analyzed the distribution of write-access to MASE. Of the 80 registered users, 49 of them contributed content. In total, 75% of contributions were by the top 10 users, of which 8 were developers and 2 were technical writers, and 55% were by the top 5 users who were all developers. While managers account for 10% of the user base, they were noticeably absent from the aforementioned list of top contributors.

The analysis also reveals that nearly half (47%) of the edits were to a subset of 20 pages, and roughly over one-third (34%) of edits were to a subset of 10 pages. This is similar to the trend depicted in Figure 1. Moreover, only about 15% of pages were created but never accessed again. Thus, the amount of effort wasted in write once/read never content is limited.

When asked of the challenges, the enabling factors, and the perceived benefits of using an informal knowledge sharing tool for knowledge sharing and organizational learning, the users indicated that the largest motivating factors were (in decreasing importance):

- Presence of required information;
- Contributing information to MASE is very easy
- Desire to help others; and,
- Encouragement from management

In particular based on informal feedback from the empolis users, many empolis users found that with MASE, “contributing is very easy, nothing before did the job that easy”.

With respect to users’ perceived benefits of the approach, analysis of the weekly usage provides some insights to this question. As shown in Figure 2, employees at empolis make use of MASE consistently every week. The large increase in overall access in the 4th week coincided with the completion of the final customization of the system and the bootstrapping of content into the repository. The sudden drop in overall access in the 11th week coincided with the German school holiday when many of the empolis employees were not at work. Our analysis also reveals that about 80% of all read-access were made by the top 20 users, of

which 14 were developers and 6 were either managers or technical writers, and close to 60% of all read-accesses were by the top 10 users of which 7 were developers.

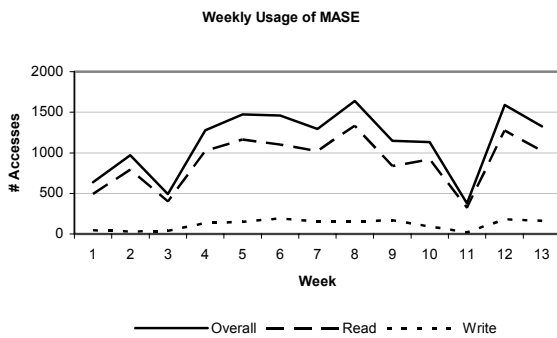


Figure 2. Weekly usage of a knowledge repository

INTERPRETATION

Based on the observations and analysis of the usage patterns, the following interpretations are made with respect to the research objectives and questioned stated at the beginning of the paper.

First, observations from the case study indicate that MASE is primarily used for asynchronous collaboration.² The comparatively low number of synchronous interactions is more an indication that synchronous communication is primarily done outside of MASE. It shows our inability to capture the amount of synchronous communication; it does not indicate that synchronous communication is not happening. Furthermore, empolis staff's urgent demand for the private messaging system and the intense use of features such as the "ListUser" plug-in in the "ListUser" wiki page provide strong evidence that it is critical for knowledge sharing tools to support not just the codification strategy but also the personalization strategy by providing means for easy communication and collaboration. This reinforces the views in the CSCW [11] and learning theory [23, 13] literature that tools which integrate codification mechanisms together with collaboration and communication features are critical for enhancing knowledge sharing and creation.

Second, observations from the case study show that much of the content reflects Taylor's description of information use accurately [22].

Third, given MASE's flexibility in support for both structured and unstructured information, observations from the case study indicate that roughly over 80% of the information content was annotated in either unstructured or in a hybrid form that consists of both structured and unstructured form. Similar to the finding from Taylor's case study, the finding from our case study demonstrates that there is a

² The main effort in the development of MASE was provide means for asynchronous knowledge sharing. Thus, tool is being used for what is being developed.

greater need for unstructured than structured knowledge.³ It provides clear evidence that it is insufficient for knowledge sharing tools to support only structured/formalized information content; there needs to be built-in support for both structured and unstructured content.

Fourth, despite the fact that less than a handful of users had contributed the vast majority of content, it should be highlighted that none of the top 10 contributors in MASE are management staff. Most of them are developers with a few of them being technical writers. In a sense, they formed a self-organized experience factory unit while still working on their normal development tasks. This self-organization process of running the experience factory may have the benefit of avoiding the ivory tower syndrome that is sometimes attributed to centralized experience factory units and process groups ("process police"). Members of such a typical centralized experience factory units are often not involved in the day-to-day software development tasks and the knowledge content that these members put in an experience repository is often perceived as out-of-touch with reality and therefore not used. This leads some users of existing experience bases to raise concerns about the usefulness of such experience bases—sometimes polemically calling them "write-only memories" or "experience cemeteries" [19]. This was certainly not the case with the use of MASE at empolis given that only 15% of wiki pages are created and not accessed again. MASE remained in active use to this date even after our researchers left empolis.

Finally, with respect to the perceived advantages of using a lightweight approach for knowledge sharing, many empolis users reported an unparallel ease in contributing information. This may be attributed to the simplicity of the wiki markup language and we believe it is this simplicity that facilitates efficient collaboration between information contributors and readers. This is because information content on any wiki pages are nearly free-formatted text. In typical project web sites, information content is often embedded among HTML markup elements and this can make it time consuming for updating content. While one can argue that even the wiki markup language may be too complex for typical computer users and that a WYSIWYG (What-You-See-Is-What-You-Get) web page editor should be used instead, there are two counterarguments. First, software developers are no typical computer users and they are unlikely to have problems learning and using the wiki markup syntax, as supported by the empolis staff's feedback. Second, the use of WYSIWYG web page editors requires the users to use another tool. In contrast, updating information content in wiki pages is nearly as easy as read-

³ One could also argue that the knowledge structuring mechanisms provided by MASE are too weak to be used and, thus, the users revert to informal contents. The counterargument is that we did not find any evidence in our surveys that the users felt limited by the knowledge structuring means provided by our tool nor did they request enhancements regarding formal knowledge representation (although empolis employees are very familiar with these).

ing them. In addition, end-users can simply use the web browser to update information. They do not have to learn another separate tool just for editing information on a web-based knowledge repository. At empolis, one user even stated that Wikis are great to “*get people involved and make them having fun while writing*”.

On the other hand, the ease of contribution has resulted in a proliferation of wiki pages. In fact, one user even stated that “*retrieving becomes harder every day, because of the growing amount of content and the lack of structure*”. To address this concern, a future challenge is to provide users a way to categorize wiki pages with as much ease as that for making updates to wiki pages.

LIMITATIONS

While this case study shed lights on how multiple software development teams use an informal knowledge repository, MASE in particular, there are limitations and threats to the validity of the case study.

First, this case study provides insights only on how software development teams make use of an informal knowledge repository. While such data is useful, the case study does not reveal the learning impact on individual software developers when they make use of the repository.

Second, the case study is conducted within a short time period during which the tool’s long term effectiveness (or ineffectiveness) for facilitating organizational learning cannot be accurately observed.

Third, the content analysis of the MASE repository in this case study is conducted by only one investigator, whose judgment may be subjective.

Fourth, the number of case studies conducted is low. It is insufficient for revealing accurately the tool’s effectiveness and the derived findings cannot be generalized, as those are findings from just one case study alone.

Fifth, the positive participant feedbacks in this case study are collected informally and hence may not be representative of other participants who have not provided feedback.

CONCLUSION & FUTURE WORK

To facilitate organizational learning, the complementary nature of the personalization and codification strategies and their benefits underlie the need for knowledge sharing tools to support these two strategies.

To help facilitate knowledge creation, MASE supports the personalization strategy by enhancing communication and collaboration through its support for synchronous work. To help realize increased knowledge re-use, the tool implements the codification strategy through its use of Wiki technology to provide support for asynchronous work. To enhance its use in a dynamic environment with minimal overhead, it also supports the use of both unstructured and structured information content.

To obtain feedback on how MASE facilitates knowledge sharing, a three-month long exploratory case study was conducted at empolis, a medium-sized global software organization with a distributed team environment.

In practice, the tool is used heavily. In particular, users found unparalleled ease in contributing knowledge. The approach was seen to help identification of team members’ expertise; self-coordination for collaborative work; sharing of product, project, and social knowledge; and generation of ideas for solution development.

The empirical results provide evidence for the need of knowledge sharing tools (1) to incorporate not only codification-oriented repository technologies but also those that facilitate communication and collaboration among people, and (2) to support not only structured but also unstructured information content. It also reveals that an informal knowledge authoring tool like MASE is used for sharing content for problem understanding, instrumental, projective, social, expertise location, and content navigation purposes. We also observe self-organized maintenance of the repository content among the ordinary repository users as a result of wikis’ open-edit nature. Feedbacks from the end-users provide encouraging support for the continuing use of MASE for knowledge sharing purposes. Critical success factors cited for tool adoption include a critical mass of relevant information, users’ willingness to share experience, as well as support and participation from upper management.

As illustrated in this case study, when many minds collaborate together in a Wiki repository, content albeit useful will be put in the wrong place. This problem will increase as users contribute more and more content into the Wiki repository over time. One area of future work is tool support for analyzing Wiki page content and their interrelationship and providing knowledge refactoring support.

In terms of empirical evaluation, this case study is limited in terms of its observation timeframe; a longitudinal study will be beneficial for more accurately determining the long-term effectiveness of the proposed approach and of MASE in facilitating knowledge sharing.

While a substantial amount of data has been collected in this case study, data such as the number of read- and write-accesses are not direct indicators of the amount of learning individual users may derive as a result of using the repository. In light of this, it is desirable to perform a direct observational study of the use of the proposed approach in order to access their learning impacts on software teams.

Lastly, given information content that are similar in nature, a study that compares and contrasts the use of MASE against another knowledge sharing tool that uses a more formal knowledge representation mechanism can provide insights into the values and limitations of an informal knowledge repository like MASE.

ACKNOWLEDGMENTS

We are grateful to all employees whom we have worked with at empolis for providing us their opinions on knowledge sharing using MASE.

REFERENCES

- [1] Althoff, K.D., et al., "CBR for Experimental Software Engineering", *Case-Based Reasoning Technology – From Foundations to Application*, Springer (1998)
- [2] Basili, V., et al., "The Software Engineering Laboratory – An Operational Software Experience Factory", *Proc. 14th International Conf. on Soft. Eng.* (1992).
- [3] Basili, V., et al., "Experience Factory", *Encyclopedia of Soft. Eng. vol. 1*, John Wiley Sons, pp. 469-476 (1994)
- [4] Basili, V., et al., "Implementing the Experience Factory Concepts as a Set of Experience Bases", *Proc. 13th Int. Conf. on Software Engineering & Knowledge Engineering*, pp. 102-109 (2001)
- [5] Brössler, P., "Knowledge Management at a Software Engineering Company – An Experience Report", *Proc. 1st Int. Workshop on Learning Software Organization*, Kaiserslautern, Germany, pp. 77-85 (1999)
- [6] Chau, T., "Inter-Team Learning for Agile Software Processes", MSc Thesis, University of Calgary, Canada (2005)
- [7] Cunningham, W., and Leuf, B., *The Wiki Way Quick Collaboration on the Web*, Addison Wesley, Reading, MA (2001)
- [8] Dingsøy, T., "An Evaluation of Research on Experience Factory", *Proc. 2nd Int. Workshop on Learning Software Organization*, Oulu, Finland (2000)
- [9] Dingsøy, T. and Hanssen, G., "Extending Agile Methods: Postmortem Reviews as Extended Feedback", *Proc. 4th Int. Workshop on Learning Software Organization*, Chicago, US, Springer Verlag, pp. 4-12 (2002)
- [10] Fitzpatrick, G., "Emergent Expertise Sharing in a New Community", *Expertise Sharing – Beyond Knowledge Management*, MIT Press, pp. 81-109 (2003)
- [11] Greenburg, S. and Roseman, M., "Using a Room Metaphor to Ease Transitions in Groupware", *Expertise Sharing – Beyond Knowledge Management*, MIT Press, pp. 203-255 (2003)
- [12] Hansen, M.T., et al., "What's your Strategy for Managing Knowledge?", *Harvard Business Review*, 77(2), HBS Press, pp. 106-116 (1999)
- [13] Hildreth, P., *Going Virtual: Distributed Communities of Practice*, IDEA Group Publishing (2004)
- [14] Houdek, F., et al., "Establishing Experience Factories at Daimler-Benz. An Experience Report", *Proc. 20th Int. Conf. on Soft. Eng.*, Kyoto, Japan (1998)
- [15] Huysman, M. and de Wit, D., "A Critical Evaluation of Knowledge Management Practices", *Expertise Sharing – Beyond Knowledge Management*, MIT Press, pp. 27-55 (2003)
- [16] Johansson, C. et al., "Talk to Paula and Peter – They are Experienced", *Proc. 1st Int. Workshop on Learning Software Organization*, Kaiserslautern, Germany, Springer Verlag, pp. 69-77 (1999)
- [17] Krishnaswamy, S., et al., "Knowledge Elicitation through Web-Based Data Mining Services", *Proc. 3rd Int. Workshop on Learning Software Organization* (2001)
- [18] Liao, M., et al., "Ontologies for Knowledge Retrieval in Organizational Memories", *Proc. 1st Int. Workshop on Learning Software Organization*, Kaiserslautern, Germany, pp. 11-26 (1999)
- [19] Melnik, G. and Holz, H. (Eds.), *Advances in Learning Software Organizations*, 6th Int. Workshop, LSO 2002, LNCS, vol. 3096, Springer Verlag (2004)
- [20] Schneider, K., et al., "Experience in Implementing a Learning Software Organization", *IEEE Soft. May/June*, IEEE, pp. 46-49 (2002)
- [21] Taylor, R., *Information Use Environments*, V. Dervin & M. Voight (Ed.), *Progress in Communication Sciences 10*, Ablex, Norwood, NJ (1991)
- [22] Trittman, R., "The Organic and Mechanistic Form of Managing Knowledge in Software Development", *Proc. 3rd Int. Workshop on Learning Software Organization*, Kaiserslautern, Germany, Springer Verlag, pp. 22-37 (2001)
- [23] Wenger, E., et al., *Cultivating Communities of Practice: A Guide to Managing Knowledge*, HBS Press, (2002)