

Experiences in Distributed Software Development with Wiki

Khalid Rames Al-asmari, Liguu Yu
Computer Science and Informatics
Indianan University South Bend
South Bend, IN 46634, USA
{kalasmar, ligyu}@iusb.edu

Abstract

Software Development as a global enterprise is current a reality for many large corporations and it is one of the rapidly growing trends in the software industry. Because in global software development, developers are located in different cities, different countries, or different hemisphere, communication and coordination is the major concern in the management of distributed teams. The traditional communication mechanisms such as face-to-face meeting is no longer appropriate for distributed software development. Therefore, there is a demand for new communication and coordination tools that can support the distributed development. Wiki is invented as a tool for writing in the Web. It contains freely expandable collections of interlinked pages which are easily editable by any users with a Web browser. In this study, we form a distributed team to work on a software project, in which, wiki is used as the major communication and coordination tool. The goal of this study is to understand how wiki can facilitate communication, coordination, and documentation in distributed software development.

1. Introduction

Global software development (GSD) is defined as "software work undertaken at geographically separated locations across national boundaries in a coordinated fashion involving real time and asynchronous interaction" [11]. GSD has become extremely important for organizations in the climate of increasing tendency towards globalization and global outsourcing [6]. Many companies have begun to adopt GSD in their software products. For Example, IBM, British Airways, Alcatel, British Telecom and General Electric have moved parts of their software development to other countries, such as Ireland and India [9, 1].

The potential advantage of distributed development is that talent engineers around the world can be gathered to

work on the same project without relocating. On the other hand, distributed teams also face new challenges, particularly in communication, documentation, and coordination [7, 10]. For example, how could developers communicate effectively without face-to-face meeting? How could a team leader manage a project with all the members located in different places? How should the various documents be produced and verified in an efficient way that won't compromise the benefits of distributed development [2, 5, 8, 3]? To solve these problems, various number of methods are employed to facilitate communication and coordination among the members in a geographically distributed team. These methods include telephone, email, virtual conference, and so on. However, all these methods have their limitations; they are either less productive, inconvenient, or expensive. Therefore, there is a need for more productive, more convenient, and more reliable tools to support distributed software development.

Wiki, a tool for writing the web, has been extended to other areas, such as course management and bulletin board. However, wiki, or wiki-similar tools have not been widely used in software development. In this study, we form a distributed team and use wiki as the communication, coordination, and documentation infrastructure to work on a software project at Indiana University South Bend. The objective of this study is to understand how wiki or wiki-similar tools can facilitate distributed software development.

The remaining of the paper is organized as follows: Section 2 discusses various communication tools in distributed software development. Section 3 describes wiki. Section 4 describes the the project and the communication infrastructure. Section 5 contains the knowledge and the experiences learned from this project. Our conclusions are in Section 6.

2. Communications in distributed development

As discussed before, communication in a geographically dispersed team is the major concern in global software development[2]. To support communication and coordination

in distributed environment, some mechanisms have been used, include:

- **Travel.** Airplane can bring team members face-to-face. Globalization means people interact with each other more frequently using many different methods, including travel. Working geographically dispersed should be just like working in the same building. Therefore, travel is still a necessary communication mechanism for distributed team. However, travel is one of the most expensive and time-consuming communication methods; the more you travel, the more you reduce the benefits of distributed development. Therefore, travel should be used only if necessary.
- **Phone.** Phone conversation is a traditional communication technique. A phone call is more efficient for a step-by-step instruction or an urgent issue resolution. However it has the drawback that the involved developers need to be available at the same time. Therefore, it is limited by the time zone differences and the work schedule conflicts between team members.
- **Email.** Email is one of the most widely used tools for distributed teams. People use email for day-to-day communications. However, email is an informal communication method and it does not support automatic backup. Other similar tools are voice mail and fax machine.
- **Newsgroup.** The threaded discussion in a newsgroup allows multi-developers to communication asynchronously on any topic. It is also an informal method, but, better than email, the discussions can be saved for future reference.
- **On-line project management software).** The on-line project management software can provide information like project overview, project organization, project plan, time schedule, and work assignments. It is the formal communication method and is maintained by the project manager. However, most project management software only support one-way communication, that is from manager to developers; they do not support communication from developers to managers and multi communication channels between developers.
- **Web(video) conference.** Web (video) conference tools can be used to give presentations and demonstrate deliverables. The traditional formal team meetings such as project kick-off and product demo can be done remotely using web(video) conference software. It has been reported that the more visual contacts within a team, the more the members feel like working in a team. The drawback of Web(video) conference

software is that they are not convenient for team members living in different time zones.

- **Content management software.** Content management software allows members in distributed a team to work together on the same document, such as the system design specification, at the same time. Other software tools, such as Jakarta Slide, can handle more complex documents, including PDF, Microsoft Word. However, most of the content management software is complex and does not support communications between developers.

Therefore, there is a clear demand for a tool that can support both communication, coordination, and documentation in distributed software development. It should be easy to use and not expensive. Wiki is such candidate software.

3. Wiki

Wiki is a software tool that allows users to freely create and edit Web page contents using any Web browser. It supports hyperlinks and has a simple text syntax for creating new pages and editing existing pages on the fly. It has inherent mechanisms to protect the contents from malicious altering. These mechanisms include page history, email notification, and password. Most important, because wiki emphasizes on open and easy access/modification, it provides a great opportunity for collaborations and interactions [4]. Table 1 shows the applications of wiki in supporting various tasks. We can see, wiki can support most of the tasks related to communication and coordination. Therefore, wiki is an ideal tool for distributed software development.

Table 1. The applications of wiki

- | |
|--|
| <ul style="list-style-type: none">• Scheduling• Clarification• Meeting agendas and notes• Project Plan and related documents• Task assignments• Group announcements• Document preparation• Status Reports• Contact Information |
|--|

4. IPDNS project

The Informatics Plasma Display News Server (IPDNS) is a project intended to develop a news server to drive the plasma display operated by the Informatics Department at

Indiana University South Bend. The target software is expected to facilitate the posting of top Informatics and IT news stories from the department, across the campus and around the world.

Five stakeholders are involved in this project, they are three developers, one primary client and one instructor mentor. The main areas of work for this project are identified first. Each developer chooses two modules and a research work for which he would assume full responsibility, including the drafting of the documentation of his work. However, the final product and the documentation are edited, revised and finalized by the group.

The Wiki site (<http://mypage.iusb.edu/~rbatz-ing/wiki101/pmwiki.php/P565Project/HomePage>) is used to communicate, coordinate and jointly develop concepts, documents and software. As items are developed they will be posted for review and comment. Various indices and RSS feeds are associated with this site to make it easier to identify new materials to be reviewed.

To work in a distributed environment, besides wiki, we also use other communication methods, such as email and phone. However, most of the communications are wiki-based. Table 2 shows the different methods used for different type of communication within the development team.

Table 2. Methods for different types of communication within the team

Methods	Email	Phone	Wiki
Problem presentation			✓
Project status review			✓
Peer review			✓
Coediting			✓
Task assignment			✓
Project progress update			✓
Project schedule			✓
Brainstorming	✓		✓
Release notice	✓		✓
Request for clarification	✓	✓	✓
Issue resolution	✓	✓	✓
Postmortem reviews			✓

5. Knowledge and experiences

Wiki is used in the entire software development. Before the project starts, we set up wiki. It is continually used even after the project is finished. In software development phase, wiki is used to facilitate the coordinations among team members; in software maintenance phase, wiki is used to monitor the operation of the software product. In this section, we present the knowledge and experience we learned

about using wiki.

5.1. Coediting

In this project, the most important application of wiki is coediting. Table 3 shows the summary of the four major documents produced in this project. They are project proposal, software requirement specification (SRS), software design specification (SDS), and testing plan. The second column is the size of the document measured in number of words. The third column shows the total changes to the document measured in number of words. Each addition, subtraction, or modification of one word is considered one change. The other columns contains the information about the time span in the coediting of the document, how many times the document has been coedited, and the number of different locations, measured in different ip addresses that the team members participated in coediting.

Table 3. Summary of the coediting of four documents

	Size (words)	Change (words)	Change (times)	Span (days)	IP
Proposal	2756	9927	169	18	7
SRS	1147	3471	88	125	9
SDS	2542	2380	63	27	7
Test plan	1262	1140	13	2	2

From Table 3, we can see that the changes made to proposal and SRS are about four times and three times of their final documents respectively. Especially for proposal, in 18 days, it was modified 169 times. This means, on average, proposal is modified 9 time a day by different members. Without the support of wiki, this would be impossible even for professional developers by using the traditional coediting techniques, such as exchanging files using floppy disk or CD, not to mention that all our developers are either full time students or full time staff. Besides working on this project, they have other duties to carry on.

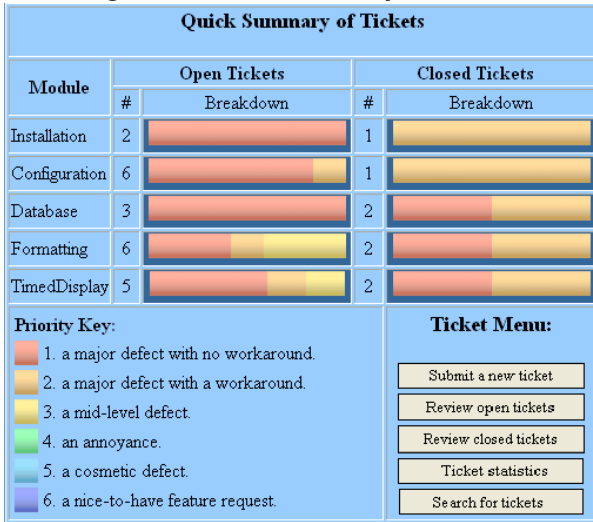
From our experience, wiki not only facilitates the coediting, but also speeds the documentation process. It reduces the time for file exchange and allows multiple programmers to work on the same document at the same time. Without wiki, the project can not be finished within five months.

Another advantage we found about wiki is the unlimitation of location and time. As long as the developer can access the internet, no matter where he is and what time it is, he can modify the document if he wants. For example, when a developer is on his trip to New York, no one expected he can work this project. However, with the laptop and the internet access, he continually contributed to this project in his hotel.

5.2. Trac ticket system

Trac (<http://www.edgewall.com/trac/>) is a system for managing software projects. It provides flexible web-based issue tracking services. In this project, we integrate Trac system with wiki and use it for various missions, especially project management. Figure 1 shows the progress of current tasks displayed using Trac ticket system.

Figure 1. Quick summary of tickets



In our project, we use ticket to report bugs, request for clarification, and assign tasks. We present an issue or a problem by creating a new ticket or reopening an exist ticket. A ticket contains enough information about the problem or the issue, such as the reporter, the status, the type, the priority and the resolution of the issue. Most important, Trac integrated with wiki provides a flexible issue management system; we can change and comment to the ticket at any time.

We found tickets are also helpful in reviewing the project status. The graphical notation makes it user friendly. Tickets can also facilitate project review after it is finished. We can extract data about a particular task or issue. The data can then be analyzed using integrated statistical tools. The data and knowledge gained from ticket system are valuable assets to improve the software process.

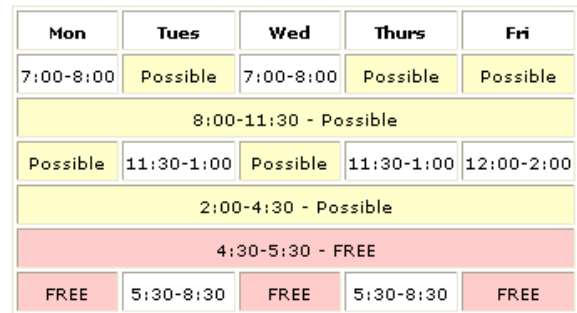
5.3. Other tasks

In our project, at most of the time, wiki, instead of a phone and an email is used for communication. We found wiki has the following advantages. (1) Compared to phone conversation, wiki is not constrained by the time schedule of the developers. (2) Compared to email, message post to

wiki is more easy to be noticed. On the other hand, developer's email account may be filled with junk mails or other unrelated mails. It is likely that the developer may ignore and accidentally delete an important email regarding the project from other developers. This kind of communication failure can be avoided by using wiki.

Wiki is used in many activities: we use wiki to organize the meeting notes, we use wiki for peer review of source code, and we use wiki to indirectly communicate with others. For example, wiki is used to show the time schedule of the developer. Therefore, the schedule is available to all other team members. If there is a new appointment or a change of schedule, the developer simply updates the wiki web page. This changes can be instantly available to others. Figure 2 shows the screen shot of one such schedule.

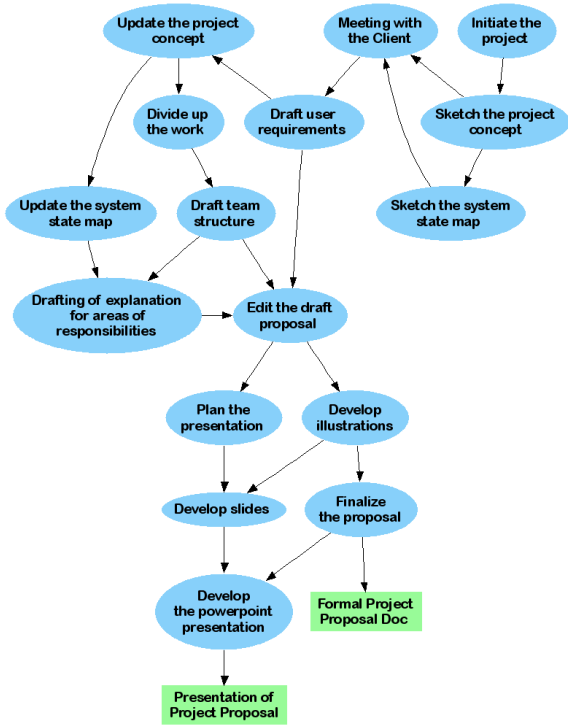
Figure 2. Wiki is used to show time schedule



In our project, wiki is the only software used for the management of the entire project. The project manager post the finalized document on the web site to make it available for all the developers. The manager can control the modification of these documents. Beside formal project document, other project information is also exchanged with wiki. Figure 3 shows the dependency of all the tasks created in the planning phase by the project manager. It is available in wiki and can help other developers to understand the project progress and the current development stage.

So far, we talked about how wiki can facilitate the coordinations within the development team. On the other hand, we also use wiki in the communication between the developer and instructor mentor. The final document and the presentation slides are made available through wiki. According to our experience, wiki is also a ideal platform for the communication between client and developer. By directly accessing the wiki web site, the client can know the progress of the project, download demo, and interact with developers.

Figure 3. Wiki is used for project management



5.4. Limitations

Like with any other software tools, wiki has limitations. (1) Currently, wiki does not support coediting of complex document type, such word, excel. It would be much helpful, if in the future, wiki can support coediting of different types of document. (2) We believe visual contacts are important for software development, if wiki can be integrated with other web conference software, it would be more helpful for some specific tasks, such repairing relationships within developers or between developers and client. (3) Wiki is best for software development and maintenance, however, it is not suitable for requirement elicitation. A face-to-face interview or an on-site observation is more efficient than wiki to obtain the software requirement from the client. (4) To provide a more flexible coediting environment, wiki should be able to merge conflicts automatically based on some grammar and word-checking systems.

6. Conclusions

In this study, we form a distributed team to work on an academic project. Wiki is used as the communication, coordination, and documentation infrastructure. Our experience

shows that wiki can support various communication and coordination tasks. Compared to other communication methods, wiki is easy to use, reliable, and inexpensive. Therefore, wiki is an ideal tool for distributed software development.

References

- [1] E. Carmel and R. Agarwal. Tactical approaches for alleviating distance in global software development. *IEEE Software*, 18(2), 2001.
- [2] S. Cherry and P. Robillard. Communication problems in global software development: Spotlight on a new field of investigation. In *Proceedings of the 3rd International Workshop on Global Software Development*, Edinburgh, Scotland, May 2004.
- [3] D. Damian. Global software development: growing opportunities, ongoing challenges. 8(4):179–182, 2003.
- [4] A. Dsilets, S. Paquet, and N. Vinson. Are wikis usable? In *Proceedings of the International Symposium on Wikis*, San Diego, CA, 2005.
- [5] J. Herbsleb and A. Mockus. An empirical study of speed and communication in globally-distributed software development. *IEEE Transactions on Software Engineering*, 29(6):481–494, 2003.
- [6] J. Herbsleb, A. Mockus, T. Finholt, and R. Grinter. An empirical study of global software development: Distance and speed. In *Proceedings of the 23rd International Conference on Software Engineering*, 2001.
- [7] J. D. Herbsleb and D. Moitra. Special issue on global software development. 18.
- [8] F. Lanubile, D. Damian, and H. L. Oppenheimer. Global software development: technical, organizational, and social challenges. *ACM SIGSOFT Software Engineering Notes*, 28(6), 2003.
- [9] P. Mohagheghi. Global software development: Issues, solutions, challenges. Technical report, University of Science and Technology (NTNU), 2004.
- [10] R. Prikladnicki, J. Audy, and R. Evaristo. An empirical study on global software development: Offshore insourcing of it projects. In *International Workshop on Global Software Development*, pages 53–58, 2004.
- [11] S. Sahay. Global software alliances: the challenge of standardization. *Scandinavian Journal of Information Systems*, 15:3–13, 2000.