RESOLVING COMMUNICATION CONCERNS AND EXPLORING THE EMERGENCE OF SOCIAL NETWORKS IN COLLABORATIVE SOFTWARE DEVELOPMENT

S MD HAROON,

M.Tech.

D. KHADAR HUSSAIN

M.Tech,CSE Dept,

JNTUA College of Engineering,

Anantapur-515002, A.P.

C. NAGARJUNA

M.Tech,CSE Dept,

JNTUA College of Engineering,

Anantapur-515002, A.P.

J. SHILPA

M.Tech,CSE Dept,

JNTUA College of Engineering,

Anantapur-515002, A.P.

ABSTRACT

Collaborative software development is very important as the developers in a team might be located in various geographical areas. There are groupware products that accommodate the needs of the developers who collaborate. Formal and technical communication among the team members is crucial for the success of the endeavor in this regard.Towards this end Treude and Storey explored work item tagging in which they consider a scenario where multipleparties or developers can involve in technical communication. Tagging a work item and updating it as the work progresses can provide useful knowledge to team members so that they can avoid lengthy technical discussions. Here Developa prototype application is designed and built in order to facilitate multiple users of the development team who collaborate from different geographical regions. The users involved include tag authors, work item authors and work item owners.

Technical communication among the three users is provided by the prototype besides allowingthem to have informal communication. The experimental result reveals that the prototype is useful to support communications among the members of collaborative software development team.

Introduction

Software development has gone through many phases over a period of time. From the initial "trial and error" kind of development, it went to the software process models that can help in systematic software development. The latest development is that the team members need not be located in particular geographical area. It is possible to have collaborative software development that helps team members to work a software product without time and geographical restrictions. With these provisions in place, humans can solve very complex problems as well [1]. Computer Aided Software Engineering (CASE) is the discipline that helps developers to utilize the computerized software development which increases the productivity. With the help of CASE tools available, it is possible to achieve smooth communication across the stakeholders of a project. Though individuals are involved in software development, an individual is not doing it alone. It is the combined effort of many users and now it is also possible for a team of members to collaborate from various geographical locations across the globe.

The tools that are existed for collaborative software development include Jazz [2] and INCOME/START [2]. The former is from IBM. These tools help the communication process of the team members. However, these tools do not have any communication procedures with respect to social aspects [4]. Thus there is possible necessity for such tools to have communication provisions for social networking as well among the software team members. The formal and informal communications can help to built relationships among the team members besides solving problem in technical communication. The Jazz tool is used to have light weight communication in the form of tagging. Tagging work items help in communication scenarios of projects being done in collaborative fashion. In [5] the Jazz tool was explored in collaborative environment. Here built a prototype application with web interface that helps people from many geographical locations to have collaborative effort in technical communications of software development. A prototype application is built in order to facilitate multiple users of the development team who collaborate from different geographical regions. The users involved include tag authors, work item authors and work item owners. Technical communication among the three users is provided by the prototype besides allowing them to have informal communication.

The rest of this paper is designed as charts. Section 2 analysesrelatedworks. Section 3 focuses on Jazz tool as described in [5]. Section 4 provides information about the proposed design and implementation of web based prototype. Section 5 presents the experimental results while the section 6 concludes this paper.

Related Works

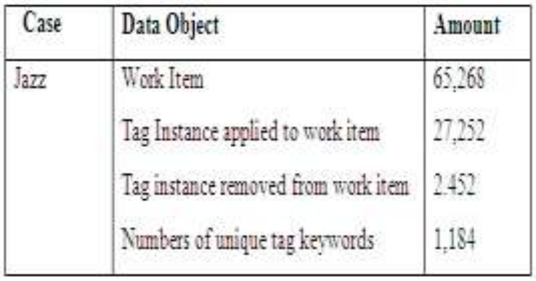
This section reviews literature on prior works pertaining to collaborative computing and work item tagging. Human beings saw the software development as a challenging task for many years [1]. Traditionally the software development is carried out by group of people forming a team. As part of software engineering there are many process models came into existence. As the task of software development is not simply writing some programs, a systematic approach and systematic communication process is required in the development process [6]. There are necessary mechanisms to bring about coordination among the members of a team [7]. There are many challenges and cultural issues involved in the team which is made up of people with diverse skills [8]. When colleagues are there in different countries, they might be facing difficulties to have collaborative access to software being developed [9]. In this case there should a tool to achieve this kind of communication among team members.

The success of projects in collaborative environment depends on the proper communication facilities provided by tools [10], [11]. Configuration management tools are also in place that help in providing communication and resolve communication conflicts as explored in [12]. The CASE tools such as Bugzilla can help developer to work faster and improve productivity in software development [13]. A light weight social computing phenomenon such as tagging can be used with social dimension. Many software tools used tagging for supporting technical communication among the software development team members. With tagging in place, developers can have control technical communication that helps in collaborative computing scenarios [14]. Another technique is annotations [15] which can be used for additional and important communication in software development. Recent tool named Tag SEA [16] is used for tagging in collaborative software development environment. Other tools which are in use include concern Graphs [17] etc. Here build a tool that helps collaborative members to have formal and informal communications.

JAZZ Tool EI Case Studies

In this section two tools which are already existed are studied. They are Jazz and EI. They provided collaborative software development provisions and also resolve communication problems among the team members. The Jazz tool supports integration of various phases in the software development. Work items of various categories can be organized in it. Jazz has merit in having tool support for collaborative software development. On the other hand EI is another tool that also supports collaboration. It also helps the teams in using work item tagging.

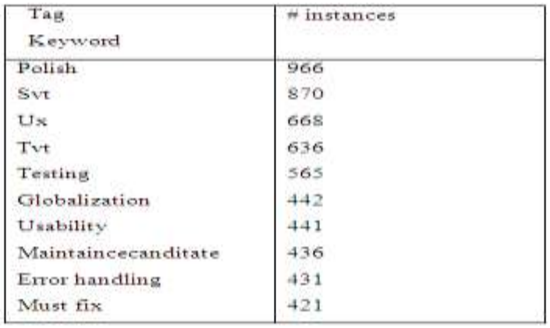
Table 1 shows the data extracted from various sources.



Case	Data Object	Amount
Jazz	Work Item	65,268
	Tag Instance applied to work item	27,252
	Tag instance removed from work item	2.452
	Numbers of unique tag keywords	1,184

As shown in table 1, the data related to Jazz tool is taken from repositories. The count of work items, the work item tagging used for work items and the instances of tags, number of keywords used for tagging is shown in table 1.

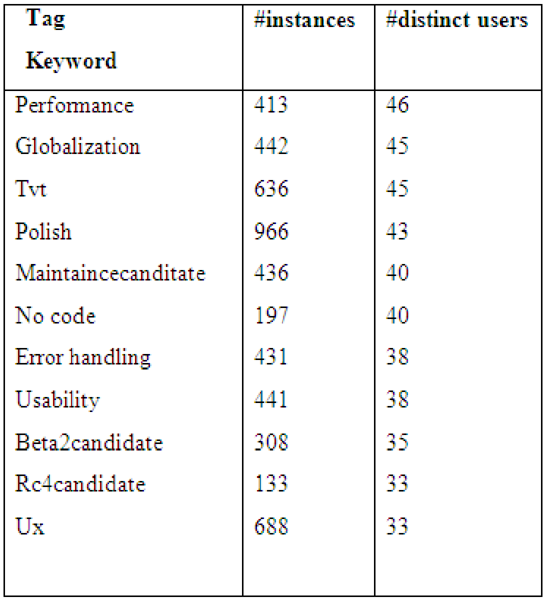
Table 2 - frequency of tag instance of Jazz tool



Tag Keyword	# instances
Polish	966
Svt	870
Ux	668
Tvt	636
Testing	565
Globalization	442
Usability	441
Maintaincecanditate	436
Error handling	431
Must fix	421

As seen in table 2, the frequency of tag instances of Jazz tool is presented in descending order by the count of tag instances. Table 3 provides tag keywords that are most frequently shared in Jazz tool.

Table 3- Most Frequently Shared Tag Keywords in Jazz



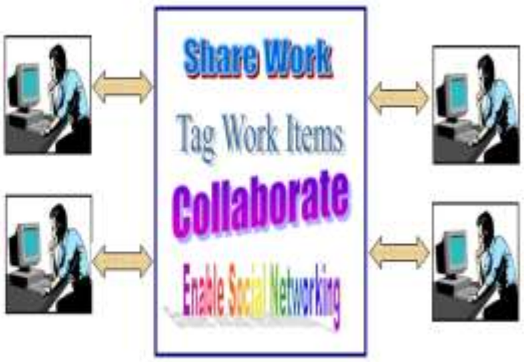
Tag Keyword	#instances	#distinct users
Performance	413	46
Globalization	442	45
Tvt	636	45
Polish	966	43
Maintaincecanditate	436	40
No code	197	40
Error handling	431	38
Usability	441	38
Beta2candidate	308	35
Rc4candidate	133	33
Ux	688	33

As seen in table 3, it shows tag keywords, the number of instances and the number of users who used those tag instances.

Proposed System

The proposed system is modeled after the concepts provided in [5]. The aim of the proposed system is to support collaborative software development in terms of work item tagging which provides technical communication possible among various users of the system. The users include work item owner, work item author and tag author. Each user has specific role to play in the application. All users can have collaborative communication with little effort. There will be no lengthy discussions required as the tags can carry sufficient meaning that can be understood by the team members. The communication concerns are taken care of by the tool that helps smooth communication among the team members. The proposed tool is also compared with other tools such as Jazz. The proposed application also supports limited social networking among the team members to have informal communications. Thus the tool proposed can support both formal and informal communications among the team members. Figure 1 shows the schematic overview of the proposed application.

Fig. 1 – Schematic Overview of the Proposed System

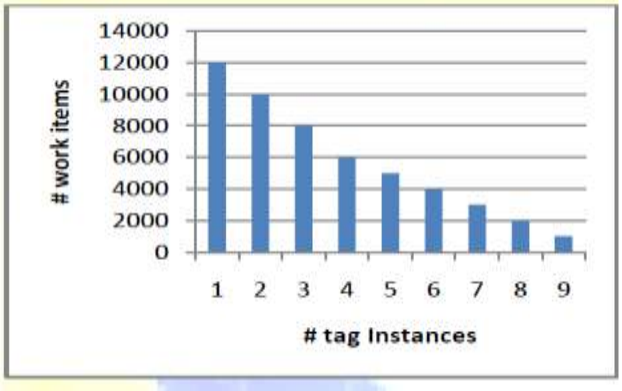


As shown in Figure 1, it is evident that the application has many provisions. They include sharing of work, tagging the work items, collaborative communication and limited social networking for informal communications among the team members.

Experimental Results

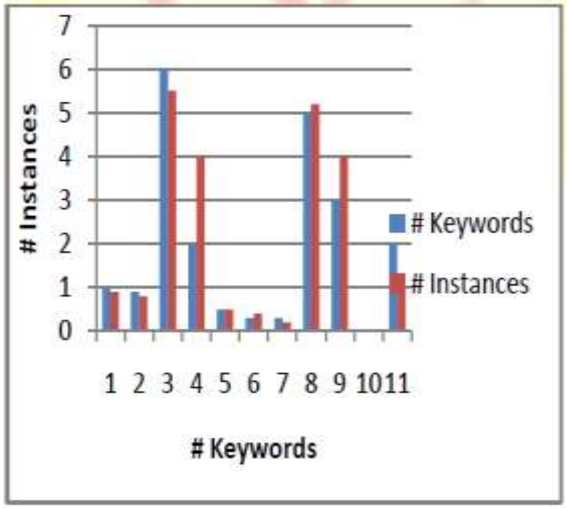
The environment used for the tool implementation includes a PC with 2GB of RAM and Core 2 Dual processor. JDK 1.7 and Tomcat 7.0 and Net Beans are used for the development of the tool. First of all Jazz and EI tools were studied before implementing the proposed application. The results of observations exist in fig. 2, 3 and 4.

Fig. 2 – Distribution of Tag Instances to Work Items with respect to Jazz



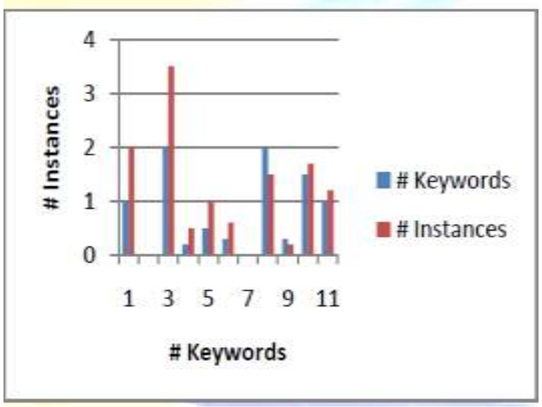
As can be seen in fig. 2, it is evident that the horizontal axis represents tag instances while the vertical axis represents number of work items. As the work items are increased the tag instances are also increased.

Fig. 3 – Number of instances and tag keywords (Jazz)



As Figure 3 shows there are number keywords and number of instances represented by horizontal and vertical axes. The tag keywords include tooling, unclassified, testing, planning, environment, documentation, cross cutting, collaboration, component, architecture etc.

Fig. 4 – Number of Instances and tag keywords (EI)



As Figure 4 shows there are number keywords and number of instances represented by horizontal and vertical axes. The tag keywords include tooling, unclassified, testing, planning, environment, documentation, cross cutting, collaboration, component, architecture etc.

Conclusion

This paper focuses on studying the need for collaborative software development and the tools available for the same. Here build a prototype application or a tool that can help a group of people to collaborate and have technical communication among them. The users who use this application include work item owners, work item authors and tag authors. The tagging concept helps them to have purely technical communication among them. The tagging eliminates the need for complex interactions unnecessarily. Moreover the tool also facilitates to have informal communication among the team members in order to have limited social networking capabilities. Thus the tool became important communication medium for both formal and information communications among the team members who participate in collaborative software development. The empirical results are encouraging.

References

[1]. C.R.B. de Souza, D. Red miles, and P. Dourish, “„Breaking the Code‟, Moving between Private and Public Work in Collaborative Software Development,” Proc. Int‟l ACM SIGGROUP Conf. Supporting Group Work, pp. 105-114, 2003

[2] R. Frost, "Jazz and the Eclipse Way of Collaboration," IEEE Software, vol. 24, no. 6, pp. 114-117, Nov. /Dec. 2007.

[3] S. Golder and B.A. Huber man, "Usage Patterns of Collaborative Tagging Systems," J. Information Science, vol. 32, no. 2, pp. 198-208, 2006.

[4] H. Kagdi, J.I. Maletic, and B. Sharif, "Mining Software Repositories for Traceability Links," Proc. 15th IEEE Int"l Conf. Program Comprehension, pp. 145-154, 2007.

[5] Christoph Treude and Margaret-Anne Storey, "Work Item Tagging: Communicating Concerns in Collaborative Software Development", IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 38, NO. 1, JANUARY/FEBRUARY 2012

[6] T. Ostrand and E. Weyuker, "A Tool for Mining Defect- Tracking Systems to Predict Fault-Prone Files," IEE Seminar Digests, vol. 2004, no. 917, pp. 85-89, 2004

[7] M. Cataldo, M. Bass, J.D. Herbsleb, and L. Bass, "On Coordination Mechanisms in Global Software Development," Proc. Int"l Conf. Global Software Eng., pp. 71-80, 2007.

[8] J.D. Herbsleb and D. Moitra, "Guest Editors" Introduction: Global Software Development," IEEE Software, vol. 18, no. 2, pp. 16-20, Mar. /Apr. 2001.

[9] J.D. Herbsleb, A. Mockus, T.A. Finholt, and R.E. Grinter, "An Empirical Study of Global Software Development: Distance and Speed," Proc. 23rd Int"l Conf. Software Eng., pp. 81-90, 2001.

[10] S. Faraj and L. Sproull, "Coordinating Expertise in Software Development Teams," Management Science, vol. 46, no. 12, pp. 1554-1568, 2000.

[11] C. Gut win, R. Penner, and K. Schneider, "Group Awareness in Distributed Software Development," Proc. ACM Conf. Computer Supported Cooperative Work, pp. 72-81, 2004.

[12] S. Sen., S.K. Lam, A.M. Rashid, D. Conley, D. Frankowski, J. Osterhouse, F.M. Harper, and J. Riedl, "Tagging, Communities, Vocabulary, Evolution," Proc. 20th Anniversary Conf. Computer Supported Cooperative Work, pp. 181-190, 2006

[13] J.B. Ellis, S. Wahid, C. Danis, and W.A. Kellogg, "Task and Social Visualization in Software Development: Evaluation of a Prototype ,"Proc. SIGCHI Conf. Human Factors in Computing Systems ,pp. 577-586, 2007.

[14] C. Treude and M.-A.Storey, "How Tagging Helps Bridge the Gap between Social and Technical Aspects in Software Development," Proc. 31st Int"l Conf. Software Eng., pp. 12-22, 2009.

[15] A.E. Hassan and R.C. Holt, "Using Development History Sticky Notes to Understand Software Architecture," Proc. 12th Int"l Workshop Program Comprehension, pp. 183-192, 2004.

[16] M.-A. Storey, J. Ryall, J. Singer, D. Myers, L.-T. Cheng, and M. Muller, "How Software Developers Use Tagging to Support Reminding and Refinding," IEEE Trans. Software Eng., vol. 35,no. 4, pp. 470-483, July/Aug. 2009.

[17] M.P. Robillard and G.C. Murphy, "Concern Graphs: Finding and Describing Concerns Using Structural Program Dependencies,"Proc. 24th Int"l Conf. Software Eng., pp. 406-416, 2002.