Improving Virtual Team Communication

Krešimir Pripužić, Luko Gjenero, Hrvoje Belani
University of Zagreb, Department of Telecommunications
Unska 3, Zagreb HR-10000, Croatia
E-mail: {kresimir.pripuzic, luko.gjenero, hrvoje.belani}@fer.hr

Abstract: **The foundation of successful software product development today is establishing an effective and efficient teamwork. In order to manage large software development projects it is needed to manage and coordinate virtual teams of programmers, engineers, business analysts and other project stakeholders. Trust is required for effective team communication. Presented software tool enhances cooperative work support by improving virtual team communication. Implemented software architecture is modular and easily upgradeable, offering multithreaded client-server communication with multiple servers. Secure communication eminent for maintaining message integrity and confidentiality among team members is implemented using SSL protocol on connections between all servers and clients.**

1. INTRODUCTION

Definition of groupware includes software, systems, and services that help groups of workers do their jobs better [1]. Cooperative work, more precisely computer-supported cooperative work (CSCW), refers to the field of study which examines the design, adoption, and use of groupware [2]. Besides dealing with cooperation or work, the term also examines competition, socialization, and play, aspects that include computer scientists, business managers, communications researchers, organizational psychologists, and anthropologists, among other specialties.

1.1 Groupware

Software tool [3] presented in this article is developed with intent to improve groupware support in virtual teams, conforming the principles of communication, collaboration and coordination, as shown in Figure 1.

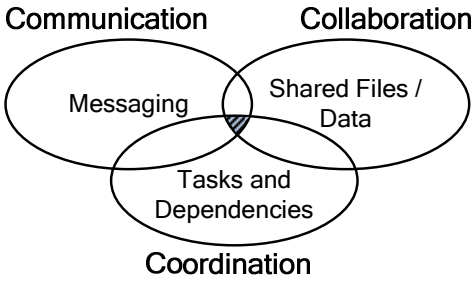

Figure 1 - Basic Groupware Principles (adopted from [4])

Communication is typically carried out in several ways of exchanging messages between team members. Collaboration is done by sharing documents and other project files. Coordination is basically manifested through tracking given project tasks and its dependencies.

In project management for software development, the team is given no-routine tasks in defined timeframe, expecting specific and measurable results in the form of software product. To successfully comply with these tasks, all three groupware principles should be applied.

1.2 Virtual Teams and Trust

There exist many different definitions of virtual teams. We selected the following definition [9] as the most suitable: a virtual team is a group of people who interact through interdependent tasks guided by common purpose. A virtual team works across space, time and organizational boundaries with links strengthened by webs of communication technologies.

From the definition above we can see that the electronic communication between team members is an essential activity in every virtual team.

Trust (i.e. confidentiality) is required for effective team communication [10]. Trust has been identified as a key factor leading to successful relationship development in virtual teams [11].

1.3 Groupware Tools

Groupware tools can be categorized into so-called Groupware Matrix, according to time and place of team's collaboration [5], as shown in Figure 2:

- Same Time/Same Place (STSP),
- Same Time/Different Place (STDP),
- Different Time/Same Place (DTSP), and
- Different Time/Different Place (DTDP).

First two categories are often called synchronous groupware support systems (SGSS). SGSS are practically used for “live” inter-project team collaboration, possibly among distributed team members.

Third and fourth categories together are called asynchronous groupware support systems (AGSS) and are used for more independent team member engagement on the project.

STDP and DTDP systems are particularly involved with virtual teams, also known as geographically dispersed teams. These teams work across time, space, and organizational boundaries in order to fulfil common goal.

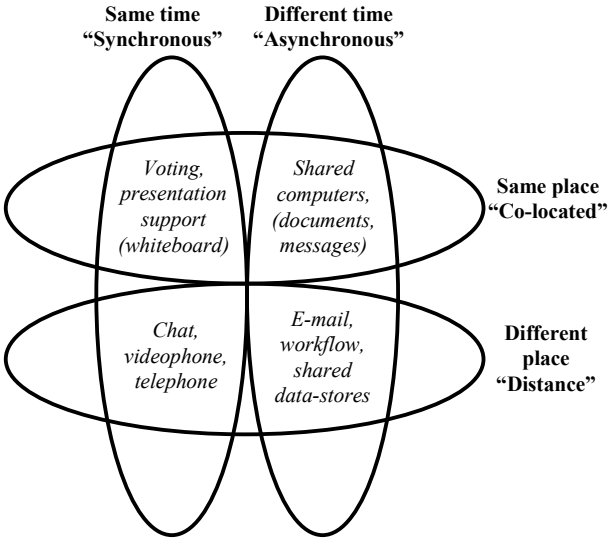

Figure 2 - Groupware categories and service examples [5]

1.4 Security

Cooperative work in virtual teams on sensitive project tasks requires the proper implementation of security mechanisms. Business and development activities ought to be managed in confidential manners and sometimes also to be conformed to very strict security regulations that the project team complies.

There are important prerequisites for establishing collaborative, secure and reliable but user-friendly environment for groupware support:

Confidentiality. An exchange of a confidential data requires secure communication channels. The confidential data is intended for authorized parties, so authorization mechanisms are also required, such as, for example, access control lists. Confidentiality is a necessary prerequisite for virtual team and trust building [13].

Authentication. Only the authenticated users are allowed to access the data. One of the most used authentication mechanisms is a password authentication.

Message integrity. Transmitted and stored data must not be altered and tampered with. In order to assure these conditions, communication must be supported with cryptographic mechanisms, such as one-way hash algorithms.

Non-repudiation. When user performs an action on data, it must be linked to the user so he cannot deny performing the action. When concerning non-repudiation of communication, it is important that a sender cannot deny having sent a message, and recipient cannot deny having received it.

2. RELATED WORK

Today, cooperative work is immanent both in academic and industrial environments. Groupware support system tools have been improving since emerging in the eighty years of the last century. The need for virtual teams to use groupware tools is necessary for establishing an effective and efficient teamwork.

2.1 Software Tools

As we mentioned before, the electronic communication between team members is an essential activity in every virtual team.

Many different groupware tools, which are intended for a virtual team communication, exist in the market today. We can group these tools in the following categories:

- Web Conferencing,
- Document Storage and Sharing,
- Instant Messaging,
- Group E-mail,
- Message Boards and
- Integrated Tools.

In our virtual team we have been interested in two of these categories: instant messaging and document storage and sharing.

The existing tools have very good communication abilities. Thus, using these tools, virtual team members may exchange almost any kind of digital information. But, however, we haven't been satisfied with either of these tools because they lack basic security and privacy mechanisms.

To override the restrictions of the existing software tools we have developed a new tool for our purpose. It is described in the following chapters.

2.2 Research Studies

When observing groupware usage efforts in academic environment, it is notable trend of comparing groupware support over traditional communication inside classrooms.

Results of the study [15] clearly show the superiority of groupware systems for education, over traditional classroom setting with no use of technology. Discussions supporting by groupware system were the most efficient and produced the most ideas for all class sections. They also had the highest amount of student participation.

The results of another study [16] provide evidence that groupware support systems can be used to enhance the classroom experience for a better learning environment. Therefore, the application of groupware systems improves team communication and team work process, and enhances individual productivity as well.

3. SOFTWARE REQUIREMENTS

Software requirement represents a condition or capability that must be met or possessed by a system or system component to satisfy a specification, standard, contract or other formally imposed documents [6].

Requirements ought to be elicitated in order to fulfil all desired needs. The idea of requirements elicitation is gathering the needs from stakeholders in new software product development. Elicitation will succeed only through collaboration between all stakeholders. Because of extreme importance of the requirements elicitation, many techniques are used for this purpose. Typical elicitation techniques are: interviewing, introspection, surveying [7], workshops, scenarios, sampling, prototyping, etc [8].

Requirements for groupware support tool presented here were elicitated using introspection, brainstorming and discussion techniques between stakeholders. The stakeholders were teaching professor and assistants, and the student that carried out the tool implementation. During requirements elicitation the following functional requirements for virtual team member were recognized:

- Enabling textual messages exchange between team members and from a member to rest of the team,
- Enabling files exchange between team members, with file repositories on dedicated servers,
- Enabling archiving messages on dedicated servers,
- Enabling textual conferences (chat) among more team members,
- Enabling audio conferences among team members,
- Enabling video conferences among team members, and
- Enabling member access from mobile devices.

In software architecture elaborated in the next section there are so called group servers. Every group server has an administrator, who requires the following functionality on his user interface:

- Starting and stopping a group server,
- Enabling connections between group servers,
- Creating, editing and deleting team member user accounts, and
- Creating, editing and deleting sub-groups.

Besides the functional requirements, there are some non-functional, regarding security, performance and similar issues that ought to be resolved:

- Securing communication on all connections,
- Preventing unauthorized access to the data, and
- Securing all user data in the database.

4. GROUPWARE TOOL

4.1 Architecture

Suppose we have a hypothetical geographically dispersed virtual team that is composed of a definite number of groups. The members of a virtual team that are located on the same location make a group. Also, suppose that the members in the different groups reside at the different geographical locations. This is obviously not the exception, but a very common case. In Figure 3 we can see an example with three geographically separated groups.

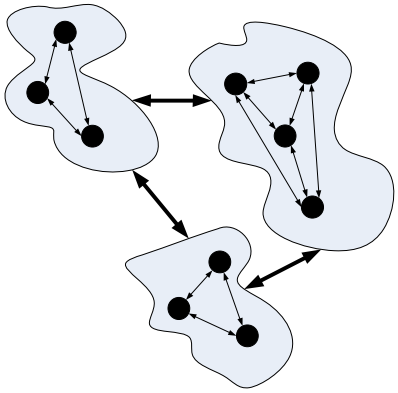

Figure 3 - Geographically separated groups

We can say that the probability of the communication between the members in a certain group (thin arrows) is larger than the probability of the communication between the members of the different groups (fat arrows). Thus, in order to reduce the communication expenses we decided to have a separate server for each group.

It is important to notice that in the case of the separate servers, they exchange information only during the communication between the members of the different groups.

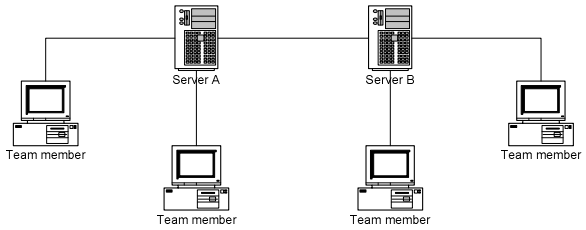

Figure 4 – Client-server architecture

Our architecture, shown in Figure 4, is based on the client-server model that supports multiple servers. Each client is connected to a single server. Clients that are connected to same server are considered to form a group and reside on the same geographical location. Different servers are connected to each other.

Communication between the servers and the servers and clients is established over a secure connection. Thus, using our tool the members of a virtual team may exchange information without being afraid that the malicious third party listens.

4.2 Implementation

All communication is asynchronous and guaranties a message delivery. This means if a receiver is currently disconnected from a server the message will be stored and sent to him the next time he connects to the server.

One of the security prerequisites that we mentioned in the previous chapters was the non-repudiation. It is important that neither sender nor receiver can deny that the exchange of information happened. Thus, in the case of our tool, the exchanged information isn't stored on the client side, but instead it is stored on the server side. A server stores such information into a database.

A server application is designed as a modular platform, as shown in Figure 5. Every module processes exactly one type of messages. One type of messages is processed by exactly one module. If a server receives a message that it doesn't support it will simply discard it. The client application is designed similar to the server application. The client module processes messages received form the server and displays information through the user interface.

It is possible to connect clients with different modules to a same server. Connection of servers with different modules is also possible.

This design allows groups of users to install modules designed for their specific needs but at the same time enables them to communicate to other users using standard modules.

To make the server as accessible as possible the server creates a separate “listening” thread for each connection it establishes. This enables the server to maintain connections to multiple clients and servers. This “listening” thread waits for messages form the other side (either the client or another server) and processes them using the appropriate module. Figure 5 shows how the server processes messages.

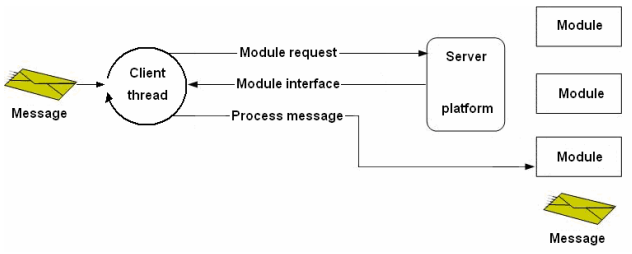

Figure 5 - Message Delivery

The original client application was intended to be used on a personal computer only. We have decided to support mobility of users. The majority of the population has own mobile phone. Thus, we have developed a client application that runs on Java-enabled mobile phones. In the office or at the home, the members of a virtual group may use their personal computers to connect to a server. In the other cases they would probably use their Java-enabled mobile phones.

4.3 Implementing SSL

Making the communication secure among virtual team members and therefore establishing trust, as we emphasized earlier, is one of the most important aspects of successful teamwork. For making it possible, we utilized SSL technology into our implementation.

SSL (Secure Sockets Layer) is a protocol layer placed between a reliable connection-oriented network layer protocol (e.g. TCP/IP) and the application protocol layer (e.g. HTTP). SSL provides secure communication between clients and servers allowing mutual authentication, the usage of digital signatures for data integrity, and encryption for privacy. SSL protocol stack is shown in Figure 6.

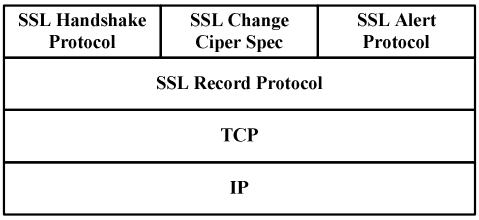

Figure 6 – SSL Protocol Stack

One of the most common examples of SSL usage is securing HTTP communication between a browser and a Web-server. In these conditions, HTTP used over SSL is called HTTPS. HTTPS uses different server port (by default 443) then HTTP, so simultaneous use of non-secured HTTP is also enabled.

Referring to the client-server architecture shown in Figure 4, it is important to emphasize that a connection between every two entities is SSL-secured. The network is built in the following way: any two group servers are started and SSL connection between them is established performing SSL handshake protocol. Then, every of remaining servers is connected with one of already active servers, until all designated servers are active. Finally, every client performs SSL handshake with its group server, and the tool becomes fully operational.

5. CONCLUSION

Emphasizing democratic relations inside virtual teams brings new ideas very welcomed in the early phases of software development. Brainstorming for requirements elicitation, and some architecture-specific remarks to system

design by an ordinary team member are strongly encouraged by usage of such a groupware tool.

Software tool described in this article meets the area of cooperative work in software development, fitted in the project management principles, especially suitable for small projects. However, designed modular platform enables embedding additional functionality for managing large projects.

Although similar scientific research have been performed and elaborated [15] [16], the preferable focus of this work is student-made software tool implementation and customization according to the needs of small virtual teams.

5.1 Future Work

Multithreaded design of the server-side enables various clients to connect to the one group server at the same time, so the tool architecture is extendible. Plans for future work in tool implementation may include adding support for audio and video conferencing among virtual team members.

Along with the four categories in the Groupware Matrix, there is also suggestion for a fifth category called Any Time/Any Place (ATAP) [5]. This kind of groupware activities is conducted independent of when or where the team is meeting. Tools that support ATAP practically come from the four categories of the Groupware Matrix that contain seamless interfaces enabling their use in the ATAP way. Observing our tool to be ATAP-typed is therefore not misleading.

REFERENCES

[1] Esther Dyson, “Why groupware is gaining ground”. Datamation, 36(5), 1990, 52-56.

[2] Tom Brinck, Darren Gergle, Scott D. Wood, “Usability for the Web”, Academic Press, USA, 2002.

[3] Luko Gjenero, “Groupware tool for virtual teams”, Diploma thesis No. 2620, Faculty of Electrical Engineering and Computing, Zagreb, September 2005. (in Croatian)

[4] Lotus Development Corporation, “Groupware: Communication, Collaboration, Coordination”, White paper, Lotus Corporation, 1995.

[5] Robert Johansen, “Groupware: Computer support for business teams”, Free Press, New York 1988.

[6] Roger S. Pressman, “Software engineering: a practitioner’s approach”, McGraw-Hill, London, 2000.

[7] Hrvoje Belani, Krešimir Pripužić, Katarina Kobaš, “Implementing Web-survey for software requirements elicitation”, Proceedings of the 8th International Conference on Telecommunications ConTEL 2005, 2005, pp. 465-469.

[8] Joseph A. Goguen and Charlotte Linde, “Techniques for Requirements Elicitation”, Proceedings of Requirements Engineering '93, IEEE Computer Society, 1993.

[9] J. Lipnack and J. Stamps, "Virtual Teams", Wiley, London, U.K., 1997.

[10] J. Suchan and G. Hayzak, "The Communication Characteristics of Virtual Teams: A Case Study", IEEE Transactions on Professional Communication, Volume 44, Issue 3, Date: Sep 2001, Pages: 174 – 186.

[11] Henttonen, K., & Blomqvist, K., "Managing Distance in a Dynamic Environment – The Role of Trust in a Virtual Team in a Global Telecommunications Company", Trust within and between Organisations, Amsterdam, the Netherlands, 23-24 October, 2003.

[12] Bradley C. Wheeler, Alan R. Dennis, Laurence I. Press, “Groupware Comes to the Internet: Charting a New World” The DATA BASE for Advances in Information Systems, Summer-Fall 1999, Vol. 30, No. 3, 4.

[13] Peter Andrews, “Trust-building for a virtual team”, Executive technology report, IBM Business Consulting Services, August 2004.

[14] Simson Garfinkel, Gene Spafford, “Web Security and Commerce”, O’Reilly & Associates, Sebastopol, 1997.

[15] Pierre Balthazard, William Tullar and Tricia Romani, “Groupwork, GSS and the classroom: Teaching challenges and opportunities for the 21st century”, A research paper, The University of North Carolina at Greensboro, Greensboro, NC, 1998.

[16] Mohammad Z. Bsat, Carl Rebman Jr., Casey Cegielski and Fred L. Kitchens, “An exploratory study of a classroom application of group support systems technology”, A research paper, 2003.