25846.

25845

Scalable and Reliable Multi Session Document Sharing System

R. Sureswaran, S. Noori, R. Budiarto & S. Rao.
Network Research Group
School of Computer Sciences
University Science Malaysia
{sures, salah, rahmat, subbarao}@nrg.cs.usm.my

Abstract

The explosive growth of the Internet and the availability of bandwidth, demands applications that allows for real time collaborative work. This paper proposes a document-based sharing between clients in a distributed network environment. Current applications that allow for collaborative work often face problems in synchronizing and are not real time. Furthermore, they fail to meet the requirements of today's explosive number of Internet users. Due to the nature of the Internet, they are also prone to error and thus not reliable. Present system is a real time, synchronized document sharing application that enables users to share all web based and popular productivity application documents over existing network environment.

1. Introduction

Collaboration can be defined as involving *people* working together by *sharing* information and processes. This expansive definition covers many different business situations involving people in various roles interacting with each other and using different types of content in a multitude of ways. It includes collaborative tools used as means to combine the skills, knowledge, and efforts of two or more people to achieve shared objectives.

The number of users is a key measurement of how important a collaborative application is, as we can see in figure 1[1], now all internet applications are towards collaboration especially standalone email and integrated collaborative environments (e.g., Microsoft Exchange/Outlook and IBM Lotus Domino/Notes) also conferencing applications (Document Conferencing/Sharing).

As the Internet goes widespread, information-sharing technology becomes various. Until now, documents are shared by means of the Web and FTP in wide area networks, and Windows file sharing in local area network [2], these systems could be accessed independently by more than one person for their own purposes but do not allow for iterative or bidirectional information sharing, or other forms of human interaction. So the need for flexible, reliable and scalable document sharing system will become more pressing.

The document conferencing system that we have implemented is a real time, synchronized document collaboration and sharing application that enable users to share all web based and popular productivity application documents over existing network environment. Real time here implies that all the users will be viewing the same page of the document while synchronized means that the control of the documents that are done are reflected in real time but also with means so that all users will get the same changes without being out of sync. This includes the control of power point slides, updating of documents, which includes word, excel etc.

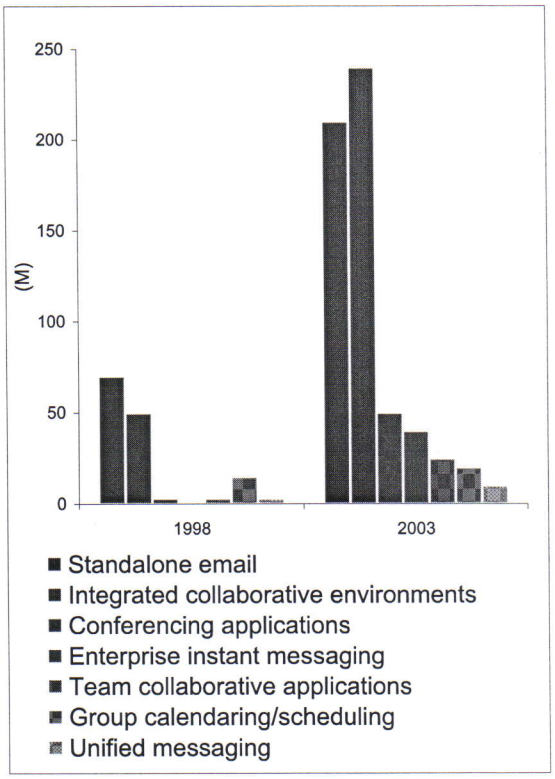

Figure 1. Worldwide users of collaborative applications

There are two approaches to implementing a document sharing system where real time sharing of documents is needed, Image Based Sharing and File Based Sharing. In image based sharing, the shared

document is opened locally in the chairman's machine and the screen of the chairman is periodically captured as an image file. This image file is then sent to the participants periodically as changes occur. This will enable the users to see the changes that are being made to the documents in real time [3]. The drawback of image based sharing is the bandwidth overhead of transmitting the images in real time to the participants. Another limitation is that the users will not be able to save the documents viewed as these are in image format, and this also posses limitations on sharing of multimedia content where it is accompanied by audio.

File based sharing works on file transfer mechanism as its backbone. Documents to be shared are first transmitted from the presenter to the participants and then opened locally in their respective machines. There is an initial bandwidth overhead when transmitting the documents but further updates only need the update commands to be transmitted. As the documents shared are present locally they can be saved for later reference. This architecture also gives the users a file transfer service. We have adopted the file based sharing approach in our system.

2. System design

Document sharing is based on distributed network architecture (client-server model). The server acts as a central unit for authentication, exchange of data and controls. The server is the coordinator for document conferencing sessions. It is designed to handle multiple sessions.

Reference to document here includes Image files, Flash, Word, Excel, Power Point, Html, Media, Adobe PDF, AutoCAD and etc.

2.1. Multi session conferencing

The document conferencing system allows for multiple sessions or conferences to be held concurrently. Each conference is initiated by a different chairman identified by session ID. In cases where a user is invited to more than one conference he can choose to join any one conference or to revert between the conferences.

2.2. System scalability and reliability

A scalable system should be able to accommodate the growth in the number of users. This is done by adding the number of servers. Each added server would accommodate additional users. The servers can communicate with each other and this is done by identifying each server with a unique server ID. With this architecture each server will be servicing clients who have registered with them before hand. This allows for clients to do a conference with each other irrespective of the server they are attached to, as shown in figure 2.

Server load balancing for deployments that require fail tolerant system can also be implemented by using the above methods [4], [5], [6]. Upon failure of a server the clients can be redirected to another

This also opens the possibility of a server selection mechanism to be implemented. This would mean that a client will be serviced by a server which will deliver the most improvement in performance, based on bandwidth, location and etc (requires cost calculation).

The system uses multithreading for its server engine core [7]. Each client is serviced by a thread and each transfer is serviced by a newly spawned thread. There is also client threads handling updating and control. This ensures that a failure in any client will not affect a

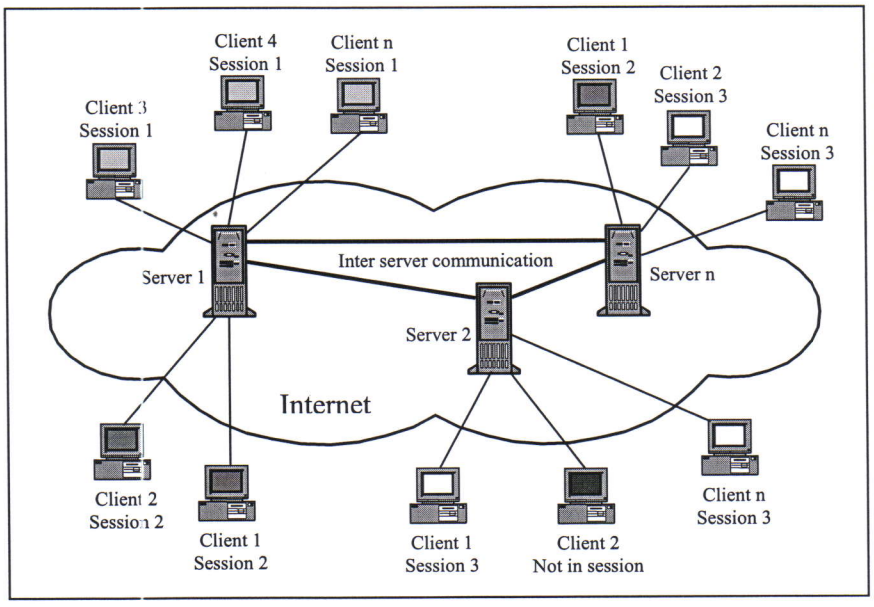

Figure 2. Multi point multi session document sharing

conference or the system as a whole. Any failure will result in termination of the particular thread and the system being updated to reflect the new changes by the client threads. An advanced method, thread pooling is also used to keep track of thread status. This coupled with the underlying TCP protocol the system is implemented on adds to the systems reliability [8], [9].

2.3. Session control mechanism

A document sharing session will be referred to as a conference. For each conference, there will be three characters:

- Chairman
- Presenter
- Participant

Chairman is the user who begins the conference and invites selected users to this conference. The chairman controls the conference by letting user/s join or leave the conference. The chairman is also responsible for the whole conference (ending and moderating).

Presenter is the user in control of the current presentation (sharing file, web page, updating file and controlling power point presentation slides). Only the presenter is able to share files.

Participants are the users who are watching the presentation.

A chairman initiates a conference by inviting users who are connected to the server or are registered to the server; the users will get an invitation to join the conference. For users that were not connected to the server when the chairman initiated the conference, they will get the invitation once they connect to the server while the conference is still in session. The chairman will be the default presenter in the beginning of the conference.

2.3.1. Delegation control

A delegation mechanism is also implemented for the facilitation of control passing [10]. This allows the chairman to delegate control from any presenter to any participant. The current presenter is also able to pass the delegation to the next presenter or return it back to the chairman. The chairman also has the ability to withdraw the delegation at any time. As an added convenience the participants are allowed to request for delegation and this will result in the chairman getting a prompt for the delegation request and can decide whether to pass the delegation or not, as shown in figure 3.

There are a number of problems to be addressed where implementation of delegation is concerned. This includes participants not accepting the delegation control, resulting in the conference session being in a hanging state as there is no presenter. This of course can be overcome by the chairman withdrawing delegation. A better solution for this problem would be a timer. When the delegation token is passed and is not accepted within certain duration of time the control is passed back to the last presenter. In cases where the presenter is disconnected for any reason the delegation is automatically passed return back to the chairman.

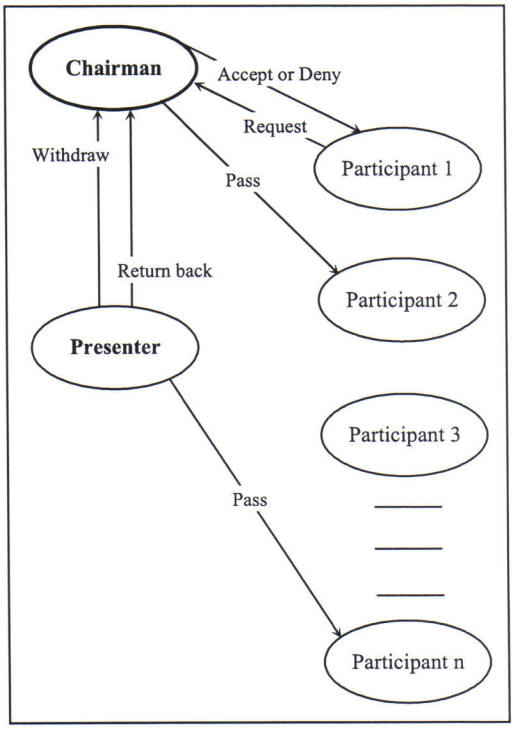

Figure 3. Delegation control

2.4. File transfer mechanism

The systems core engine is the file transfer mechanism. Shared documents or files are first uploaded to the server by the presenter before being distributed back to the participants in the conference.

The system also enables users to transfer files to all or any selected users with the optional status feedback feature. But users can only transfer files to users that are connected to the server. Users can choose to send file/files to specific user or .to all the users.

2.4.1. Document sharing process

After selecting the document and before uploading it to the server a check will done in the server if the file exists with the same modified date otherwise the server will begin receiving the file from the presenter. After successfully transferring this file a copy will be saved in the server under that conference ID and will be used later when any client joins late, as shown in figure 4.

The document distribution process begins by sending the file to all the participants. The client will not receive the file in cases where the application that is needed to open the document is not installed (e.g. Microsoft Word for opening word documents) which will result in that participant getting a missing application notification. In

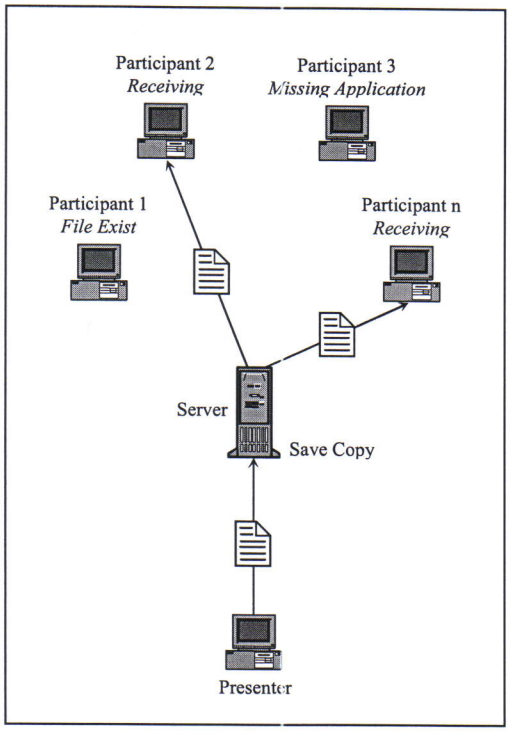

Figure 4. Document distribution process

cases where a copy of the file is already existent with the same modified date, the file will be opened directly without the need for downloading it.

These two checking have been added to the system for better bandwidth utilization.

Note: each transferred file carries its original modified date in coordinated universal time (UTC) format, this is to overcome the problem faced with different regions; and this will keep all the participants up-to-date with the original file.

2.4.2. Status feedback

Sharing a document would require the presenter to transfer the file first. Due to the inherent congestion and other variables present on the Internet which makes the nature of the Internet unpredictable, the presenter requires a method to know the status of the document transferred. Thus we present the user with the status feedback mechanism option. As this requires extra bandwidth to check each client's status, it is optional. Current status includes Waiting, Receiving, Cancelled, Failed, Stopped, and Completed.

2.5. Document load function

In cases where all the participants already have the document to be shared there is of course no need to transfer the document and thus the presenter can choose to load the document directly on all the clients locally with the load option. The location to load the document from is limited to the specified folders and this can be changed as need be. Thus the user has the option of transferring the document only to the users that don't have it or altogether ignore the users without the document and carry on with the session. The presenter can know the existence of the documents in the clients by using the check file availability feature.

2.5.1. Check file availability

The presenter can check the availability of any document in the participants before sharing or loading it. This checking includes the application needed to open the file, the existence of the file and whether the file is up-to-date or unmodified. The presenter will get a report from all the clients regarding the existence of the document (Report Format: Missing Application, Up-To-Date, Unmodified and Not Exist).

2.6. Join late problem

When a user joins late to a conference he can choose to download the current file being presented or any selected files that was presented in the conference. The list of presented documents and web links that were shared in the conference is presented to the client for downloading. This is possible as the server keeps a copy of the presented documents for each conference. This copy is deleted soon after the conference ends. The following files that are presented will be received by this user as usual [11].

2.7. Document compatibility

A Document conferencing system should be able to support a wide variety of document formats and the design should be such to also support future formats. For this requirement and after much research we came to a conclusion that the best document container would be a web browser object. Web browsers already support a wide variety of document formats. Any added support for new documents can be added by implementing or adding the plug-ins required for the particular format to the system. The availability of the required plug–in to view the document is checked before the document is loaded to avoid errors or bandwidth wasting. On the unavailability of the required plug-in the user will get a message that the application required for viewing the document does not exist.

Following are some of the documents and files supported by document conferencing:

- MS Power point files
- MS Word
- MS Excel
- Image files
- Auto CAD drawings
- Adobe PDF files
- Audio and Video files
- Macromedia Flash

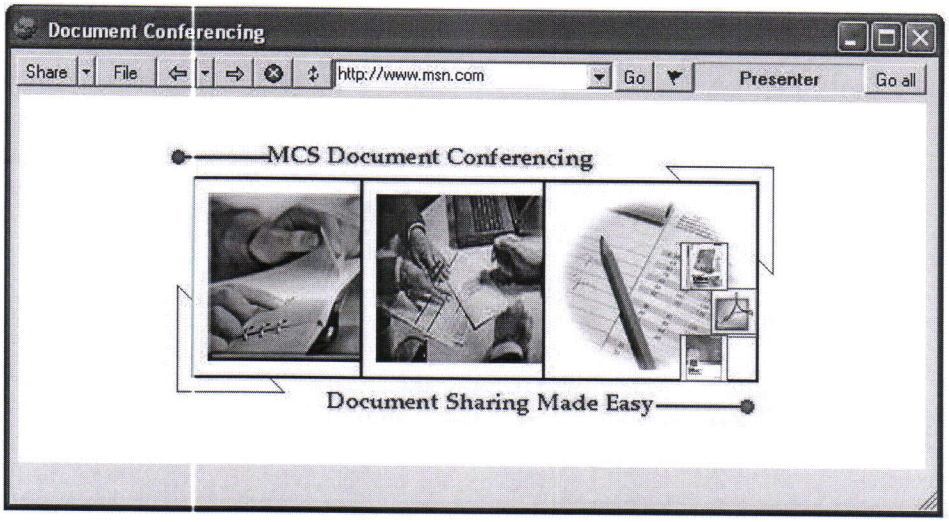

Figure 5. Document conferencing client

- HTML or web pages
- All text based files

Figure 5 shows the document conferencing interface with the web browser object inside.

3. Real time document updates

3.1. Synchronization

The presenter can either control the documents that is shared or to modify it. If the presenter is controlling a document for example a power point presentation, only the control command is passed such as slide number, next animation and etc. Sharing a web page is done by passing the URL of the site and thus it is subject to the Internet connection speed of respective clients. The presenter can go to any presented document and synchronize to the participants to the current opened document.

3.2. Document updates

When the presenter modifies a document such as a word or an excel document, a copy of the new document is retransmitted to the clients. Synchronization of the latest document is done by depending on the current modified date of the document.

The users (chairman, presenter, and participants) can update themselves with all the presented files/links or with the current document being presented. The user can also surf forward or backward through the presented document list while for synchronizing the user will get a list of all missing documents and web links with the latest presented one.

4. Other standard functions and options

These functions are added as a measure of convenience and user friendly features.

4.1. Auto receive

When a user has to be away from his PC for a while and still be able receive documents or files he can turn the auto receive function. With this option turned on any files sent to this user will be automatically accepted and saved to the default folder.

Note: documents received from presenter for sharing are always received and opened automatically.

4.2. Save files at end of conference

At the end of the conference clients are presented with the save file and links dialog. With this option the users can save the files and links for future references. As the files that were downloaded for the conference are saved to a temporary directory and deleted after the conference ends.

4.3. View document locally

A presenter might want to preview the file locally before sharing it with other users. To allow this the users can choose the view document locally option. After that the users can just use the share option and the file will be transmitted to the participants (the load option can also be used).

5. Future enhancements and suggestions

As the system currently uses TCP to deliver documents by transferring them this can require a

substantial amount of bandwidth directly proportional to the number of users. As such implementing a reliable multicast protocol to deliver the files could significantly reduce the bandwidth requirements, making it proportional only to the number of conferences in session. But implementing reliable multicast offers its own challenges and solutions. The system can also be further enhanced by adding server selection algorithm which will make the transfer of files between servers more efficient.

Security is an important issue in sharing documents so that an option should be added for sharing important files to prevent participants from saving them during or at the end of the conference. Documents can also be encrypted prior to the conference or during the conference by the system.

6. References

[1] Levitt, M. and Mahowald, P P. *Collaborative Email-Lowering the Barriers to Successful Collaboration*, IDC White Paper,USA, www.kubisoftware.com/docs/idcwhitepaper.pdf.,May, 2003.

[2] Han, S. and Nam, D. *Security Issues in Distributed Document Sharing System,* Network Security (ICE 615) – Fall 2001, School of Engineering, ICU, http://caislab.icu.ac.kr/course/2001/autumn/ice615/termproject/proposal_hshndy.doc , 2001

[3] Join PhoneOffice, *Multimedia Communication Integration Solution*, Technical white paper, www.joinphone.com , ,Digital Media Tech Co., Ltd. Korea, December, 2002.

[4] Miura, H.,Yamamoto, K., Nishimura, K., and Ikeda,H., *Server Load Balancing with Network Support: Active Anycast,* Japan, IWAN 2000.

[5] Bryhni, H., Klovning, E., and Kure, O., *A Comparison of Load Balancing Techniques for Scalable Web Servers*, IEEE Network, July/August 2000.

[6] Minasi, M., Anderson, Ch., Smith, B.M. and Toombs, D. *Mastering Windows 2000 Server*. SYBEX, Inc, 2000.

[7] O'Brien, L. and Eckel, B. *Thinking in C#.*Prentice-Hall, December, 2002.

[8] Comer, D.E. *Internetworking With TCP/IP Vol I: Principles, Protocols, and Architecture*. 2nd ed. New Jersey. Prentice-Hall, 1991.

[9] Comer, D.E. and Stevens, D.L. *Internetworking With TCP/IP Vol III: Client-Server Programming And Applications*. New Jersey. Prentice-Hall, 1993.

[10] MCS Ver 5.0, *Multimedia Conferencing System*, System discretion, http://www.mlabs.com.my/WhitePaper.htm , Mlabs, Multimedia Research Lab Sdn. Bhd. Malaysia, 2003.

[11] Tanenbaum, A. and Van steen, M. *Distributed Systems Principles and Paradigms*. [Prentice Hall, ISBN 0-13-088893-1], 2002.