COLLABORATIVE FILE SHARING SYSTEM USING JXTA P2P NETWORKING INFRASTRUCTURE – AN APPLICATION DEVELOPMENT

Kasyful Amron

Faculty of Computer Science, Universitas Brawijaya
Email: kasyful@ub.ac.id

ABSTRACT

This work aims to develop a simple workflow based collaborative application running over peer to peer network. Basic features of the application are to support communication and coordination in a workflow-based document production. Its offer services for text chat and file sharing. Text chat has two features, group chat or conference and personal chat, while file sharing service supports both synchronous and asynchronous mode that implies a repository function. As this application is developed with the assumption that it will be applied within a close environment, it is complied with a general security mechanism. Design and development processes of this application are depicted in the form of UML diagrams and implemented using Java Programming Language.

Keywords: *Peer to Peer, Collaboration, Juxtapose, Java, File Sharing*

1. INTRODUCTION

Computer based collaborations are still in their evolution. The demand and idea on how to do collaboration grow all the time. Every system then has its own characteristics. Some characteristics may be considered very useful in certain aspects of collaboration, but found irrelevant at the others.

This work discusses about the design and development of a system that can be used to accommodate collaboration in the aspect of document production and sharing, and also as an alternative to keep the availability of document by offering a restricted repository service that allow only certain authenticated members.

The system is designed and tested to run on independent environment and distributed geographically over an overlay network using a peer-to-peer platform.

1.1 Background

The research work in peer-to-peer groupware platform has been focusing on a workflow based group interaction. The current state of the work requires studies in the implementation of several scenarios in which one of them is to support collaborative documents production within the workflow context that should run on a peer-to-peer infrastructure over real globally distributed users.

As the system is design on a workflow based, there is a limitation of members. To control this membership, there are many options that can be chosen. Set and lock the environment, implied an authentication scheme, or doing user management with a clear classification and grouping rules are three of these possibilities (Androutsellis, 2004). These assumptions are used in the system development.

1.2 Objective and Boundary

The objective of this work is to develop a collaborative document production based workflow system that will support at least three specific features:

- common authentication scheme,
- document sharing/P2P document repository system, and
- a generic workflow template and its management system.

The application is to be designed and documented using UML as described by Alhir (2002), and to be implemented using Java platform and programming language. After the application has been implemented, it should be tested in a real globally distributed environment and its operation is therefore will also be evaluated.

1.3 Outline

This project is written in five different chapters. The introduction in Chapter 1 gives the backgrounds of the idea, objectives and boundaries that implies and the outline of the project.

The theoretical aspects that have high influence over the development of the system are then presented in Chapter 2. It is divided into three sub topics discussing about the peer-to-peer networking, the project JXTA, with the focus on Java implementation of JXTA, and a brief overview about UML and groupware.

In Chapter 3, the system design and development plan are described. It includes but not limited to the designed system architecture, JXTA protocols that are implemented and networking model for system's testing-environment.

The Analysis of the developed system in Chapter 4 is intended to describe the result of developed system as well as the results of the testing over real network.

At the end, the conclusions in Chapter 5 contain the summary of project work result and some suggestion for future work.

2. BASIC THEORY AND CONCEPTS

This second chapter will focus on the basic concepts of peer-to-peer networking. Then tries to detailing one of peer-to-peer technology, that is Project JXTA or in short JXTA, as it is the basic idea used to build this project. And at the end, briefly talk about other theoretical aspects that supporting the development process such as UML and groupware.

2.1 Peer-to-Peer and JXTA Framework

Originally, the term "peer-to-peer" has the meaning to show the equal or symmetric properties of networking. Peers, could be computer or any devices connected to the network, that have an equal chance to contribute to the overall system by sharing their resources, such as processing times, disk storages, files, and applications directly with or without support of global server or authority (Setton, 2007), (Shen, 2009).

Peer-to-peer often constructs an overlay network; a virtual network that differs from the physical view of a network; and also has a specific routing mechanism, to accommodate communication among peers. In another word, peer-to-peer has their own network architectures. Many designs of peer-to-peer result in many ways to classify the system. But in general, architectures of peer-to-peer are categorized as unstructured network (Buford, 2009).

The common characteristics of today's typical P2P systems include most of the following (Brookshier, 2002):

- peer has awareness of other peers,
- peer creates a virtual network that abstracts the complexity of interconnecting peers despite firewalls, subnets, and lack of specific network services,
- each peer can act as both a client and a server, peers form working communities of data and application that can be described as peer groups.

JXTA, as defined by its developer at Sun Microsystems (now: Oracle Corp.), is "a set of open, generalized peer-to-peer (P2P) protocols that allow any networked device-sensors, cell phones, PDAs, laptops, workstations, servers and supercomputers to communicate and collaborate mutually as peers" (Sun, 2007). JXTA itself was taken from a Greek word juxtapose that mean in English as side by side.

JXTA protocols as shown in Figure 1, are composed of six basic building blocks. These protocols divided into two categories, Standard Services Protocols, and Core Specification Protocols. It defines a set of XML messages to coordinate peer-to-peer networking.
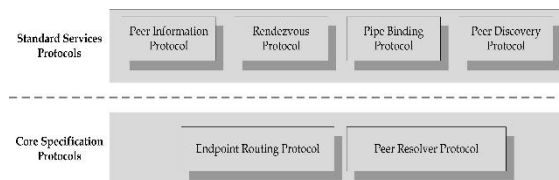

Fig. 1. JXTA Protocols Building Block
(Sumber: Sun Microsystem, 2007)

In practical manner, JXTA protocols describe some basic components of a JXTA technology. Some references define only five components, while the others mentioned six even ten different components.

However in general, the core component are (1) peer, a participant in peer-to-peer network; (2) peer group, group of peers with the same interest; (3) pipe, acts as the communication channel from sender to receiver; (4) services, specific content or application provided by certain peer or peer group that can be used by other; (5) advertisement, describes shared resources in XML format; and (6) discovery, the basic process by which peers locate advertisement about resource that it can use (Verstrynge, 2010). The first four components are identified by one or more unique IDs.

To simplify the concepts, the first three components of JXTA defined in the previous part can be seen as the entities that construct the JXTA network. Meanwhile, the last three components define how peers can perform communication and collaboration.
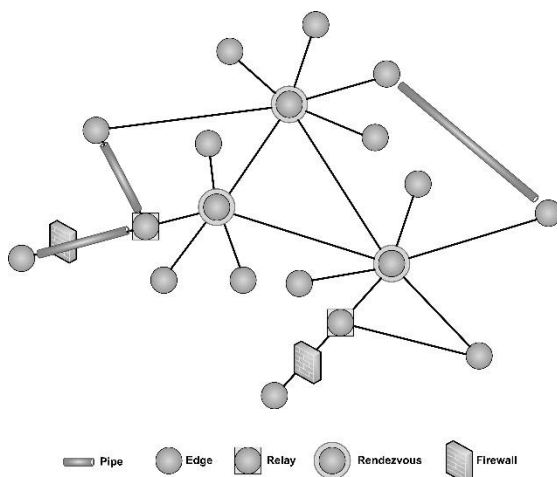

Fig. 2. Generic JXTA Network Model
(Sumber: Verstrynge, 2010)

Figure 2 shows a generic JXTA network. This network was composed of peers in three different operational modes, network connection, and communication tunnel or pipe. Two of those peers are connected indirectly via relay peer because of firewall or NAT.

Firstly, peers enabling the service and join to JXTA global network. If it is required, peers use PDP to discover peers, pipes, advertisement and all other available JXTA resources. Once other peers are found, peer can start to establish communication to one or more peers. This is done using the combination of endpoint and pipe.

Pipe that is constructed between peers could be a direct connection between the two peers, in case both of those peers can be reached directly, or indirect through a relay peer. Its depend on the position of a peer, whether it is connected without NAT and Firewall or exactly located behind one.

Data are exchanged between peers using a container called message. A message contains many sections that describe how to handle any form of data that it is associated with.

2.2 Groupware

The definition of collaborative software; also refer to as groupware; are varies. Early definition is given by Peter and Trudy Johnson-Lenz as "intentional group processes plus software to support them" Later, they redefine it; in another article; as "computer-mediated culture... an embodiment of social organization in hyperspace" (Lenz, 2012).

Other definition of groupware is "computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment" (Ellis, 1991).

Yet another definition: "groupware is a generic term for specialized computer aids that are designed for the use of collaborative work groups. Typically, these groups are small project-oriented teams that have important tasks and tight deadlines. Groupware involves software, hardware, services and/or group's processes support (Dix, 1993).

Groupware is designed to transform the way documents and other information is shared. Its goal is to enhance group process by providing tools that aid communication and collaboration among members.

Groupware can be classified according to the way it assists the processes within group or consider which aspects a groupware is intend to use. As their definition, there are also some approaches to classify groupware. Groupware system may be classified to the common time space matrix, application domains, or the so-called 3C model (communication, coordination and cooperation).

Time space matrix shown in Figure 3 enable team members to be either in the same place or geographically separated. Team members can also interact directly, or within a different time. Thus, the two-dimensional time

space matrix becomes the most common and most often used to classify groupware (Baecker, 1995).

Groupware also can be identified depend on the intensity of cooperation within a group. Teufel (1995) classify groupware systems as to the degree of support they give to these three basic aspects, communication, coordination and cooperation (Borghoff, 2000).

Communication focuses on the mutual understanding of persons through information exchange.

Coordination aims at finding the best way in which to arrange task-oriented activities and the allocation of resources in the best possible order, whereas the additional requirement of common goals makes Cooperation the most demanding of the three. Based on these three components, application classifications are positioned in a triangle as can be seen in Figure 4.

	One meeting sites (same places)	Multiple meeting sites (different places)
	Face to face Interactions	**Remote Interactions**
Syncrhronous communications (same time)	• Public computer displays • Electronic meeting rooms • Group decission support systems	• Shared vew desktop conferencing system • Desktop conferencing with collaborative editors • Video conferencing • Media spaces
	Ongoing Tasks	**Communication and Coordination**
Asyncrhronous communications (different time)	• Team rooms • Group displays • Shift work groupware • Project management	• Vanilla email • Asynchronous conferencing bulletin boards • Structured messaging system • Workflow management • Version control • Meeting schedulers • Cooperative hypertext & organisational memory

Fig. 3. Groupware Time Space Matrix
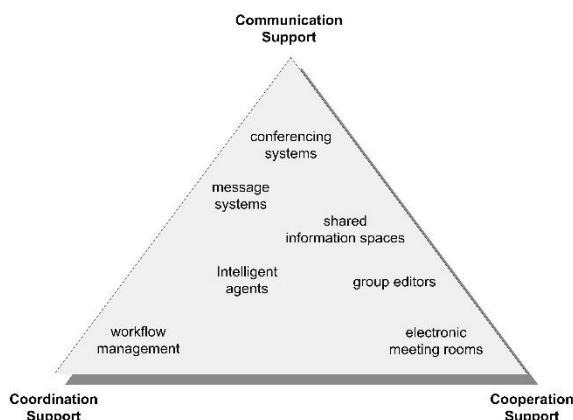(Sumber: Baecker, 1995)

Fig. 4. 3C Model of Groupware
(Sumber: Borghoff, 2000)

3. DESIGN OF SYSTEM

This work is to develop a peer-to-peer application that has features for document sharing and repository and implement a common security mechanism. A specific peer-to-peer infrastructure, based on JXTA implementation in Java was chosen. A requirement analysis is then performed resulting in five aspects of requirements listed below:

• *Security*. This aspect could be reach by localized the applications and data within a certain environment and or area, or by implementing authentication schemes.

• *Document Sharing*. Every user can share their document to the others using this application.

• *Repository*. There is somewhere within the network, a machine that has to able to save and backup the documents.

• *Document Production*. This could be a machine or agent, doing compilation over the documents and give final documents as the result.

• *Peer-to-Peer Network*. It is stated that the application will be implemented by JXTA, one of peer-to-peer network technology.

In this system, there are three basic services that will be provided. They are the text chat, the file sharing and the repository system. Hence, there will be at least three kinds of JXTA message with different tag elements required to develop all of these services. General view of system is shown in the form of use-case diagrams and were shown consecutively on Figure 5 until Figure 7.
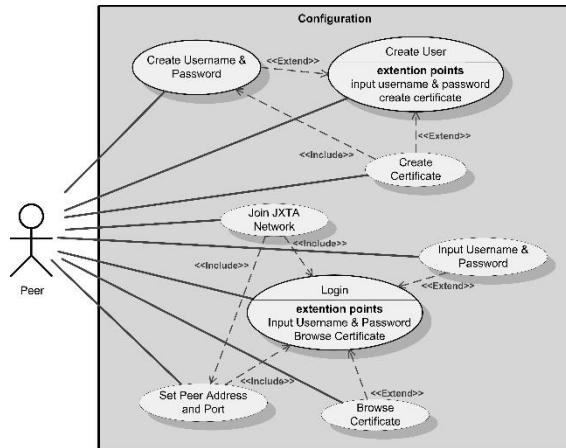
Fig. 5. Use-Case Diagram: Peer Configuration

These three basic services are used to support communication and collaboration between team members to generate specific documents required by a project.

The basic idea of this document collaboration is as follows:

• Some split files send by team coordinator to the right member using file sharing service.

• Each member completes their task and then sends the file back to the team coordinator.

• Final files are created by merging member files. This is done by team coordinator after he gets the file back from all group members.

This file sharing services is supported by message service for online communication and coordination and also by repository service to secure the availability of shared files especially for offline members.

One important note in this step is about the ID creation. IDs; all IDs including peer, group, pipe and advertisement IDs; are generated with custom method, using specific peer name, group name and hash function to make sure these IDs are unique and limited in number. This is based on the assumption that the group is closed to certain people in the same organization.

All users have to successfully unlock desktop application before using it. For the first time login, a peer must create a new username (or peer name) and password. Based on the combination of username and password, a file called certificate will be created and store in certain directory on the local host. Creation of this certificate uses MD5 encryption

mechanism. This use-case is sketch in Figure 5.

As the first services is to join to the system. After successful login, user has to configure their local IP address and port number to identify communication channel of the peer. Other configurations such as rendezvous seeding and group name are set as defaults.

Now, it is the time to join to the network by starting the JXTA network service. Processes behind starting network are left to JXTA mechanisms. Begin with creation of peer profile and ID, create a key store and X509 certificate file, and ended by configuration of rendezvous point and routing path.

Processes involved in this step are as follows:

• Peer tries to discover the parent group using PDP discovery protocols.

• If the advertisement of the parent group is found, then it joins to the parent group.

• Else, in case the peer cannot find the advertisement, it will create and join into the parent group.

All services in this default parent group are activated after a peer join to JXTA network. And when it run, they are basically treated in broadcast scheme where all peers will get all messages that exchange between peers. To accommodate a specific usage of this system, peers can create a new child group, join and send an invitation to certain peers to join the child group. There is no limitation of how many groups a peer is allowed to join into.
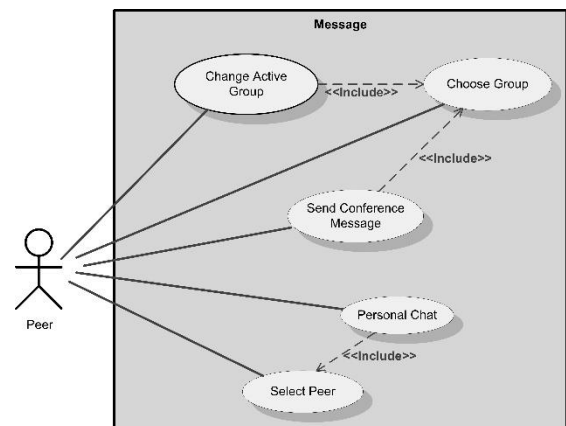

Fig. 6. Use-Case Diagram: Message Service

Second step is using message service; peers are able to communicate with other peers in form of text message. This service will be constructed using endpoint communication that is implementing Pipe Message Listener, one of JXTA standard library, used to capture all events in the communication channel. Text message can be sent to a specific peer using personal chat feature, or broadcasted to all peers within the parent group or child group in conference mode. Use-case of this service is shown in Figure 6.

Text message service has two main use-cases; the first one is to receive and process an incoming message and the second one is to send the message. This text message can be either sent to all peers in the parent group, peers within the child group, or to a single peer. Receive message function is set to always ready in every peer as a message can arrive at any time, while send message function only activated when a peer decides to send a message.

Each group has their own message service. This is the scheme used to localized message receivers. Every single message is broadcasted within their group, even if the members of a group are only two peers. Exception only apply if it is defined before that the message will be sent to a single peer by adding message elements in which specifically state the endpoint address of the destination.
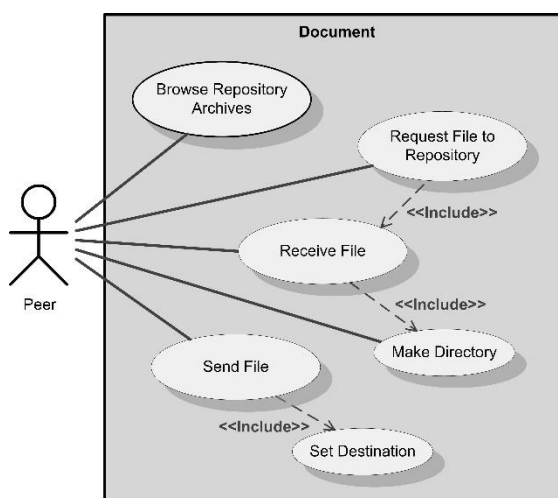

Fig. 7. Use-Case Diagram: Document Service

The last third step is the main function of system, document services. This service enable peer to send a file to a selected online peer. File is sent as stream and re-written to certain shared directory in the targeted peers and a copy is also stored in the repository peer. One peer can send; or push, in this context; a file to other peers using this service with one condition, sender and receiver peers are activated in the same parent or child group. If a file is sent with its active group as parent group, then the file will be received by all online peers.

With the assumption that team members are potentially spread geographically, there is no guaranty that the sender and the receiver peers will be online in the same time. To resolve this problem, Document service has a feature to send a backup file automatically to the repository. Here, this feature is named offline sharing.

The offline receiver peer, when it starts to be online, can then check the list of shared file within the group, comparing the files, and then send a message to the repository to request a missing file. List of files generated by repository is classified by peer group and sender peers. And only peers that are joining to the group can see the list of files. This is to control the distribution of shared files. Figure 7 shows the use-case for file sharing service.

In order to secure the availability of a file that shared within a group, a backup file is stored in repository. So, if in certain case, a peer is offline, it can send a message to repository to get the file it missed before, regarding to their offline status.

This repository peer will copy and indexing all files shared within the JXTA network. Files are restored in a corresponding peer group and peer name directories. It then creates the list and sends it as a response message to the appropriate member peers that request the list. Access right associated with peer group membership. Only an appropriate peer can see and send request to get the files.

4. REVIEW AND EVALUATION

This part discusses the system's review, the testing that has been performed and its result's evaluation. From software point of

view, review is done by comparing the objectives and the design discussed in chapter one and three with the results of the application development.

4.1 System Review

The real implementation of the JXTA network is shown in Figure 8. Three rendezvous peer are configured and are running at continuously. The first (JP) rendezvous is placed in Japan, acting as the main rendezvous with first priority routing or 1st seed in JXTA terminology. The second (ID) rendezvous is set as 2nd seed and the third (UB) rendezvous as 3rd seed. ID and UB rendezvouses, and also the repository edge peers are placed in Indonesia.

All rendezvouses and repository peer run over a virtual machine environment with different specification. UB rendezvous, for example, use two processors 3.3 GHz clock speed with 4Mb Cache Memory, 2 GB Physical Memory and 1 GB Swap Memory. It runs fedora 14, 32 bit, as their operating systems.

Operating systems is installed in a minimum option without GUI desktop. There is no other application running over the machine except standard processes of the operating systems, secure shell to control the machine remotely, and Java Runtime Environment (JRE) in which the application is developed and run. JRE version used for this development is part of Java 1.6 update 26 for 64 bit Linux operating systems.

The graphical view of the system application is divided into four different tabs, Configuration, Group Management, Messenger and Document. And can be seen respectively through figure 9 to 11.
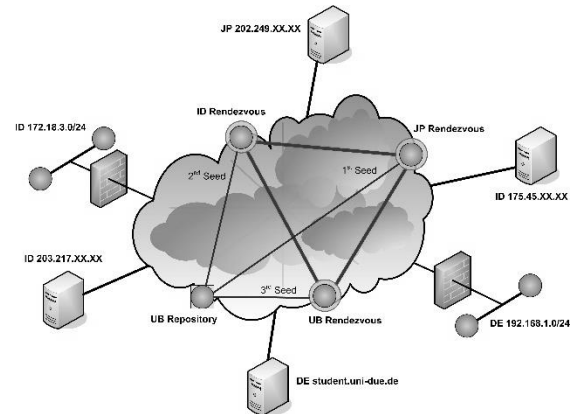

Fig. 8. Developed Networking Infrastructure

The first tab, Configuration informs and handle peer configuration, implement Authentication, Encryption and PeerConfig classes. Everything that is needed to be accomplished before a peer can join to the JXTA network is done in this tab. Create New User button used for first time login, or to create another username. There are three text fields here, username and password field, which in combination are used to create a certificate for related username.

Start Network Button changes to enable state only if the login process is passed. Pressing this button will trigger the class PeerConfig to run its methods and bring the peer to join to JXTA default parent group of the system.

Messenger tab run text chat service for peer. This tab has two main areas for doing the chat. One is for conference chat and another one for private chat. It is completed with information about running configuration, rendezvous to which the peer connected and current active group. On the top right, there is an option to change the active group, shown as group list and Change Group button. The group list here shows all groups that the peer has already joined with.

Document tab is responsible for file sharing and repository services. Left side show the archives store in the Repository Edge and the right side handle interactive process to share a file to the JXTA network. File archives are rearranged according to their peer group name. Group name is set as the parent directory, with peer name as its subdirectory. All files share by a peer will be placed in the

peer name subdirectory. Global directory, Development and its entire content can be seen and downloaded by all peers. While any files within a peer group subdirectory can only be browsed and downloaded by peer group's members.
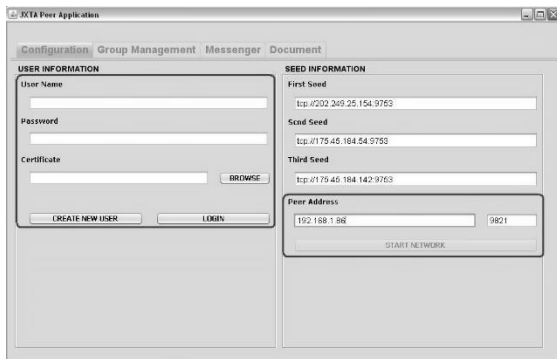

Fig. 9. Application Tab: Peer Configuration

As archives can change at any time, there is a Refresh button at the bottom side of this tab. By pressing this button, one peer sends a message to the repository and shall expect a newest list reply back. To download a file, the user can push the Request button. This will trigger the repository to send the requested file.

Process and option for sending a file take place on the right panel. This panel composed by a file browser, and combo box to select target peer. If the file and the remote peer have been selected, the file will start to be sent right after pressing the Send button.

4.2 System Evaluation

This part describes testing scenarios related to the usage of the systems and discussed the analysis of its results. The test is performed over the internet with twelve peers that are distributed geographically. The usability tests are performed over the system's basic functions or features and its general behaviors over the JXTA network. Analysis is performed mainly on the results of transfer rate of certain specified file sizes.

For this test scene, Application is executed in all connected peers; in this case the twelve peers. But only four machines are being used to test all the application's features. Other peers are used only in the correlation to investigate if there is significant influence of the number of users connected to the network.
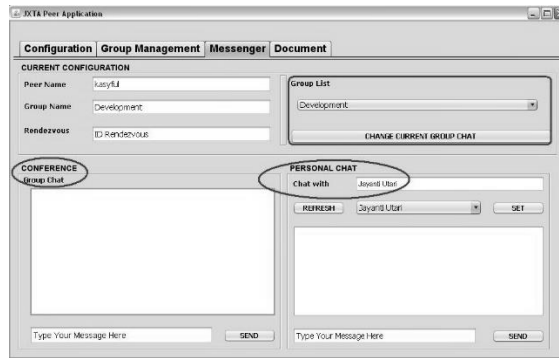

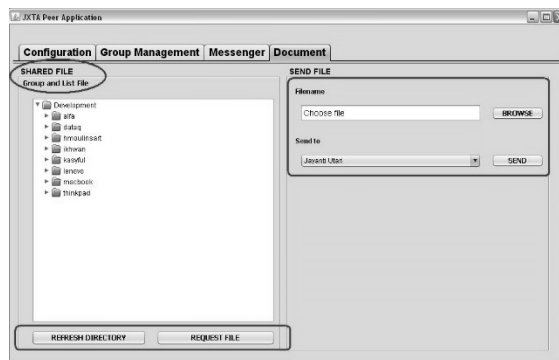Fig. 10. Application Tab: Unicast and Group Text Messenger

Fig. 11. Application Tab: Document Services

At the end, all the designed basic functions of the system have been achieved. The developed system has reached the objectives. As a team member, user can collaborate with other team member using the system. Communication is done via chat box, a feature that provide in text chat service. This service can accommodate conference chat in parent and a child group as well as personal chat.

Project documents can be shared with a privilege among the working group member in synchronous and asynchronous communication mode. Member can send their documents directly to another online member when they are online. Asynchronous mode run with the help of repository edge that archive all files shared within the JXTA network.

Application Features	Testing Result	Note
Configuration		(1) List of online peer is not distributed correctly between rendezvous. Only some online peers can be seen
- Create New User	- Passed	
- Login Process	- Passed	
- Start Network	- Passed	
Peer & Group Management		(2) Setting of a remote peer is depending on the list that is displayed in Online Peer.
- Available Group	- Passed	
- Online Peer	- Failed (1)	
- Create New Group	- Passed	
- Send Invitation	- Passed	(3) and (4) Repository sends an update list, but message can not be delivered when rendezvouses are busy
- Join New Group	- Passed	
Messenger		
- Rendezvous Information	- Passed	
- Active Group Information	- Passed	(5) As are encountered at (2), (3) and (4)
- Group Chat	- Passed	
- Personal Chat	- Passed Partially (2)	
Document		
- List of Files	- Passed	
- Refresh Directory	- Passed Partially (3)	
- Request File	- Passed Partially (4)	
- Send File	- Passed Partially (5)	

Fig. 12. Application Evaluation: Basic Function Test Results

But, along with this achievement, there are some problems encounters during the testing runs. Most problems are dealing with late update. In certain situation, advertisement discovery broadcast by a peer, returned with old advertisement, or failed. As the result, list of online peer filled with wrong information. Peers that are already offline are still captured as online ones. Or, files and directory list generated by the repository is actually an old list.

Other problem faced with the system is synchronization between rendezvous peers. JXTA routing schemes sometimes end with failure. When a peer joins to the network, JXTA will arrange with which rendezvous it will connect to. And if in the same time, two peers connected to different rendezvous, each peer return with different result, one peer can see the existence of the other, while other peer sees nothing. All of those problems are tabulated in Figure 12.

5. CONCLUSION

Based on the systems reviews describe in chapter three and four, it is can be concluded that systems development in overall can achieve the target. This is as:

• An overlay network need to deploy the system is running. System has a JXTA network with three rendezvous peers and one repository edge. These main peers are placed separately in Japan and Indonesia.

• Rendezvous peer can route and propagate text message and file from every edge connected to the network, whether the edge running with a routable IP addresses or hide behind the Firewall and or NAT.

• Application has a minimum feature as required by offering message and document services. These features can be used in collaboration for a document production among geographically distributed members.

• Repository system archives all file send within the network and can be located by an appropriate peer based on their information of peer group membership. Its function as a backup to keep the availability of file is running as well.

As the demand on collaborative software still increasing, the system should be enhanced. There are many aspects that can be revised or added to this system. Some of them are:

• System Optimization. It is true that the system still need to be optimized, both in network and software.

• Document Merge. System can be set to merge documents based on certain condition such as group name, project name or any specific tag that is defined as a goal of collaboration.

• Application Service. It is possible to share not only a document, but also an application.

• Network Scalability. The system should be implemented in a real wide network with tens rendezvous and repositories and a massive number of edge peers.

• Mobility. As the system is developed in environment independence, it can be applied also with mobile devices.

6. DAFTAR PUSTAKA

ALHIR, SINAN S. 2002. *Guide to Applying the UML*. Springer-Verlag, New York.

ANDROUTSELLIS, S. 2004. *A Survey of Peer-to-Peer Content Distribution Technologies*, ACM Computing Surveys, pp. 335–371.

BAECKER, RONALD M., GRUDIN, J. 1995. *Reading in Human-Computer Interaction: Toward the Year 2000*. Morgan Kaufmann Publisher, Inc., Toronto.

BORGHOFF, U. M., SCHLICHTER, J. H. 2000. *Computer-support Cooperative*

Work: Introduction to Distributed Application. Springer-Verlag, Berlin.

BROOKSHIER, D., GOVONI, D. 2002. *JXTA: Java P2P Programming.* Sams Publishing, Indiana.

BUFORD, J., YU, H., LUA, Eng K. 2009. *P2P – Networking and Applications.* Morgan Kaufmann, Massachusetts.

DIX, A. F. J., ABOWD, G., BEALE, R. 1993. *Human Computer Interaction.* Prentice Hall, Madrid.

ELLIS, C. A., GIBBS, S. J., REIN, G. L. 1991. *Groupware – Some Issues and Experiences,* Communication of the ACM, Volume 34.

JOHNSON-LENZ, P., JOHNSON-LENZ, T. 2012. *Rhythms, Boundaries and Containers: Creative Dynamics of Asynchronous Life - Awakening Technology.* http://nexus.awakentech.com:8080/at/awaken1.nsf/UNIDs/CFB70C1957A686E98825654000699E1B?OpenDocument. Accessed at March 27 2012.

SETTON, E., BERND, G. 2007. *Peer-to-Peer Video Streaming.* Springer, United State of America.

SHEN, X., YU, H., BUFORD, J. AKON, M. 2009. *Handbook of Peer to Peer Networking.* Springer, Berlin.

SUN MICROSYSTEMS, Inc. 2007. *JXTA Java Standard Edition version 2.5: Programmers Guide.*

SUN MICROSYSTEMS, Inc. 2007. *JXTA v2.0 Protocols Specification.*

VERSTRYNGE, J. 2010. *Practical JXTA II – Cracking The P2P Puzzle.* Downing Streams, Inc.