

1. Name of workshop : Computer Networks

2. Title of the paper : Design and Implementation of Group Management Protocol
for Multicast Communication

3. Abstract

A multicast communication requires a dynamic membership management function since the membership information changes as the receivers join or leave. Though some protocols include a 'loosely-coupled' mechanism, most conventional multicast transport protocols do not have a 'strictly-coupled' mechanism for group management. One of the main reasons may be that the conventional transport protocols like TCP or UDP are not suitable, yet. However, recently a new enhanced transport protocols are proposed and implemented in ITU-T SG 17 and ISO/IEC JTC1 SC6. Therefore it is reasonable for us to assume that soon or later a new protocol such as Enhanced Communications Transport Protocol (ECTP) will be available in the market. In this paper, under this assumption, we designed and implemented the membership management function to support desirable management of multicast members.

4. Name, Affiliation, Postal address, e-mail, telephone, fax

– **Jung-Jin Park**

Affiliation : Graduated student, Korea University

Postal address: 208 Suchang-Dong Chochiwon Chungnam, Republic of Korea, 339-700

e-mail : pjj@korea.ac.kr Telephone : +82-41-860-1424 Fax : +82-41-862-4255

– **Seok Joo Koh**

Affiliation : Senior Member of Engineering Staff , ETRI/Protocol Engineering Center

Postal address: 208 Suchang-Dong Chochiwon Chungnam, Republic of Korea, 339-700

e-mail : sjkoh@pec.etri.re.kr Telephone : +82-42-860-6218 Fax : +82-42-861-5404

– **Hyun-Kook Kahng**

Affiliation : Professor, Korea University

Postal address: 208 Suchang-Dong Chochiwon Chungnam, Republic of Korea, 339-700

e-mail : kahng@korea.ac.kr Telephone : +82-41-860-1424 Fax : +82-41-862-4255

5. Name of contact author : Jung-Jin Park

6. Key word : GMP, group management, multicast,

Design and Implementation of Group Management Protocol for Multicast Communication

Jung-Jin Park[†], Seok Joo Koh^{††} and Hyun-Kook Kahng[†]

[†]Dept. of Electronics and Information Engineering, Korea University, ROK

^{††}Electronics and Telecommunications Research Institute, ROK

Abstract A multicast communication requires a dynamic membership management function since the membership information changes as the receivers join or leave. Though some protocols include a ‘loosely-coupled’ mechanism, most conventional multicast transport protocols do not have a ‘strictly-coupled’ mechanism for group management. One of the main reasons may be that the conventional transport protocols like TCP or UDP are not suitable, yet. However, recently a new enhanced transport protocols are proposed and implemented in ITU-T SG 17 and ISO/IEC JTC1 SC6. Therefore it is reasonable for us to assume that soon or later a new protocol such as Enhanced Communications Transport Protocol (ECTP) will be available in the market. In this paper, under this assumption, we designed and implemented the membership management function to support desirable management of multicast members.

1. Introduction

Even though multicast communications require a dynamic membership management

function since the membership information changes, most conventional multicast transport protocols do not have a ‘strictly–coupled’ mechanism for group management. One of the main reasons may be that the conventional transport protocols like TCP or UDP are not suitable, yet. However, recently a new enhanced transport protocols are proposed and implemented in ITU–T SG 17 and ISO/IEC JTC1 SC6. Therefore it is reasonable for us to assume that soon or later a new protocol such as Enhanced Communications Transport Protocol (ECTP) will be available in the market, and then the ‘strictly–coupled’ mechanism for group management will be required.

In this paper, under this assumption, we design and implement the membership management function in the group management protocol for the reliable multicast communication, and test the internetworking with the other multicast protocol and the group management protocol (GMP) [12]. GMP includes Session Management (SM) and Membership Management (MM). GMP provides a framework of the multicast session management (SM) mechanism and the membership management (MM), which have been designed and implemented to support desirable management of multicast sessions and members. This protocol can be the key basis of the multicast communication for the reliable multicast communication. GMP will operate over the conventional transport protocol and/or ECTP [10] as shown in Figure 1.

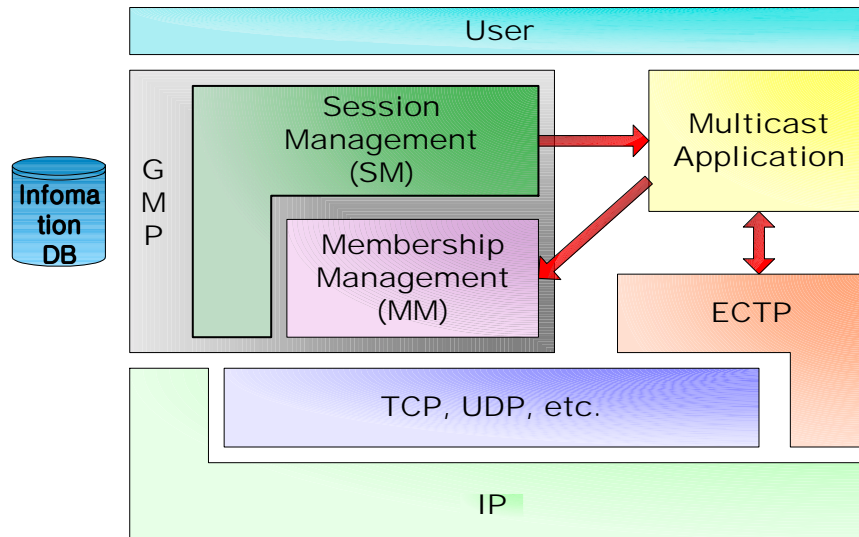


Figure 1. GMP Model

Paper outline : After this introduction in section 1, we introduce the session management and the membership management in section 2 and section 3, respectively. While the session management refers to [12], the membership management will be described in detail in this paper. In Section 4, the test environment and the implementation for the membership management are described. Especially, a new multicast application for ECTP is applied to demonstrate its proper operation over ECTP. Finally, in Section 5 future works are mentioned with summary.

2. Session Management

The session management function [11] should be activated before the membership management function is operating. In this section, we just summarize the session management function and the session information that is passed in the membership management, with which our MM operates.

Figure 2 shows the GMP control information flow from the session creation to the termination among session creator, participants, and the GMP server.

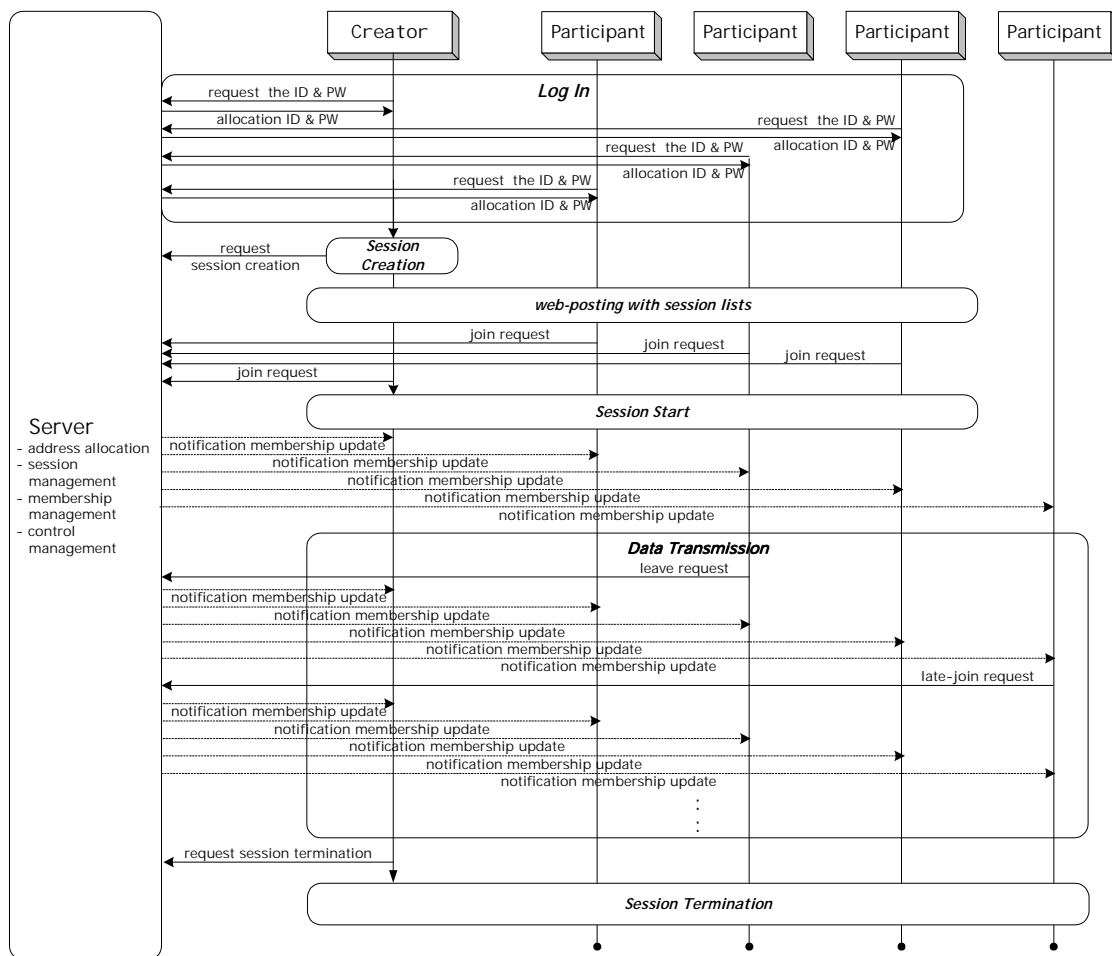
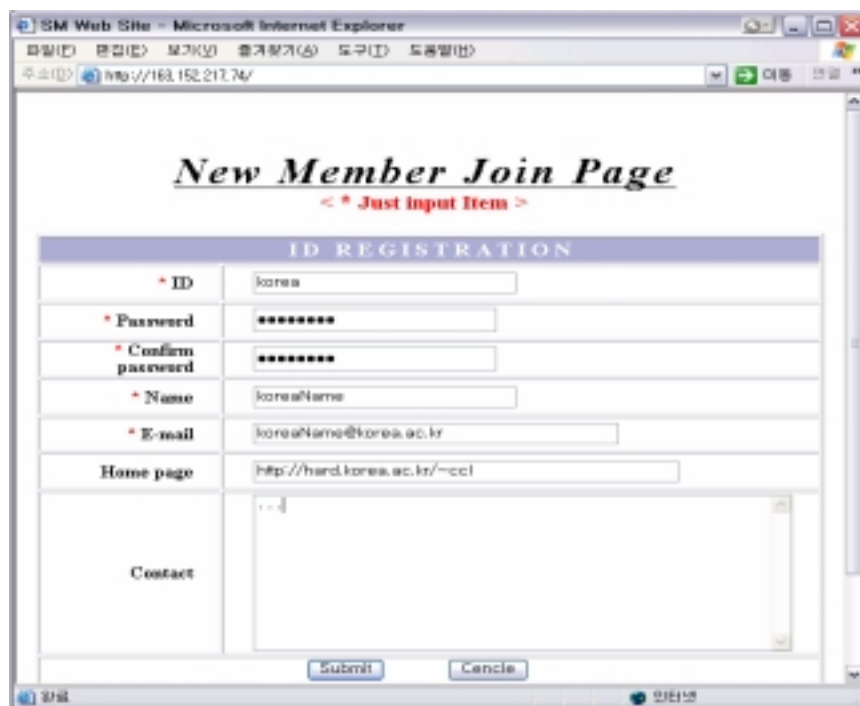


Figure 2. GMP control information and data flow diagram among clients and server.

2.1 Session Management Operation

The session management is implemented to web-based.[11]. The client should register to the SM server beforehand to use this SM server and get ID and password for the connection to the SM server. Only the registered participant in the SM server can create the session, check the session list, and join the session. Figure 3 illustrates the registration procedure to use the SM server.



The screenshot shows a web browser window titled "SM Web Site - Microsoft Internet Explorer". The address bar displays "http://163.152.212.74/". The main content area features the heading "New Member Join Page" in a large, bold, italicized font, followed by the sub-heading "< * Just Input Item >". Below this is a form titled "ID REGISTRATION" with a blue header. The form contains several input fields: "ID" (containing "korea"), "Password" (containing "*****"), "Confirm password" (containing "*****"), "Name" (containing "koreaName"), "E-mail" (containing "koreaName@korea.ac.kr"), "Home page" (containing "http://hard.korea.ac.kr/~cccl"), and "Contact" (containing "..."). At the bottom of the form are "Submit" and "Cancel" buttons.

Figure 3. Registration

2.1.1 Session Creation

The session initialization is the stage that the session creator creates the session for

the multicast communication. The session creator should determine the characteristics of session to create the session. The creator provides a detailed session description to a server and additionally determines a media type and an application, etc. according to the characteristics of session. Figure 4 shows all information about creating session.

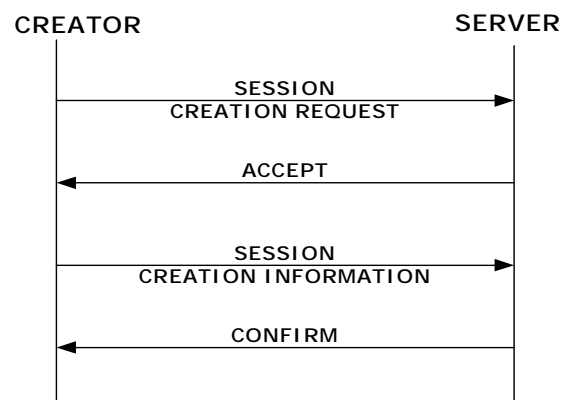


Figure 4. Session Creation Procedure

2.1.2 Session Advertisement

The SM typically advertises the information for the session over the web. All clients can show the information for opened session after login using own ID and password.

2.1.3 Session Registration

All clients can see a session list from the web page. After a participant sees the list, the participant selects the session to want to participate, and sends the **session join**

request to the server. The server who receives the **session join request** from the participant(client) adds the information of participant to the session information. In the case of the active session already, all members can see the active member list from the web page through updated active group member list periodically.

3. Membership Management

The Membership Management (MM) manages and maintains information of all participants according to the status of the member. That is, all members are managed according to their status by the session server.

3.1 Membership Management Architecture

The membership management (MM) module consists of the MM server and the MM client. The MM server is running with the SM server and waits the MM client's connection. The MM client starts action with the application when a participant joins the session. The Figure 5 shows the implementation module of the Membership Management. The MM server is consisted of a main module and a session module. When the client sends the request to the MM server to join the session, the main module delivers the session information file to client and passes the information to connect in a relevant session module to the client. If new session is begun, the

main module calls session module and updates the session list.

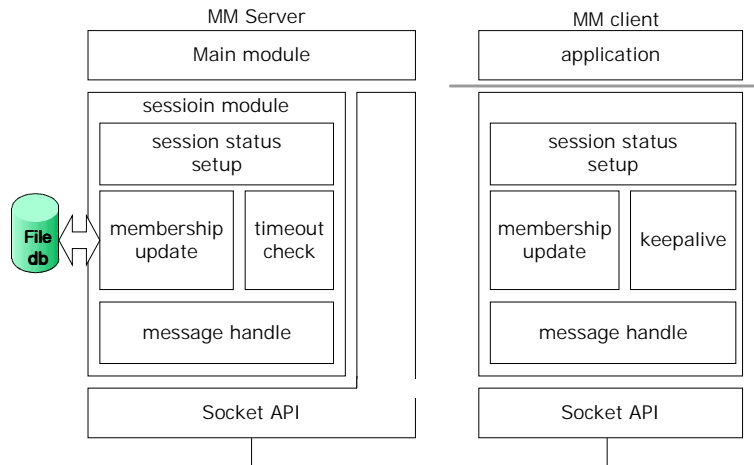


Figure 5. The MM module

The session module as module that operates original MM action sets the session state according to the information that comes over in the server (SM) and operates the membership administration action. Also, the MM server stores all information updated dynamically while session is processed to the database file. The MM client operates by the call of application, and the application operates in the client side at the same time that new participant to join the session. The MM client sends the **session join** message to the MM server and requests the session information file if the session is started. And operates original action after sets own state according to the received session information.

3.2 Membership List

The membership management mechanism that implemented in this paper has two membership lists such as the registered group member list and the active group member list. The MM server maintains and manages these membership lists to manage client's membership.

3.2.1 Registered group member list

The registered group member list is made out with information that received from the session management (SM) server as the membership list about participants that want to join the session before the session starts. The MM server updates the registered group member list when the session is started or the MM server receives the request for the registration.

3.2.2 Active group member list

The active group member list as the list about member that sends the **Keepalive** message to the MM server after the session is started is updated dynamically while the session is active. The client can see the active group member list from the web page in the client side.

3.3 Membership Check

The clients report the **Keepalive** message to server periodically to check the membership with Figure 6. If the server doesn't receive the **Keepalive** message from a client for a periodical time, the server removes the client from the active group member list and updates the membership according to this client's report.

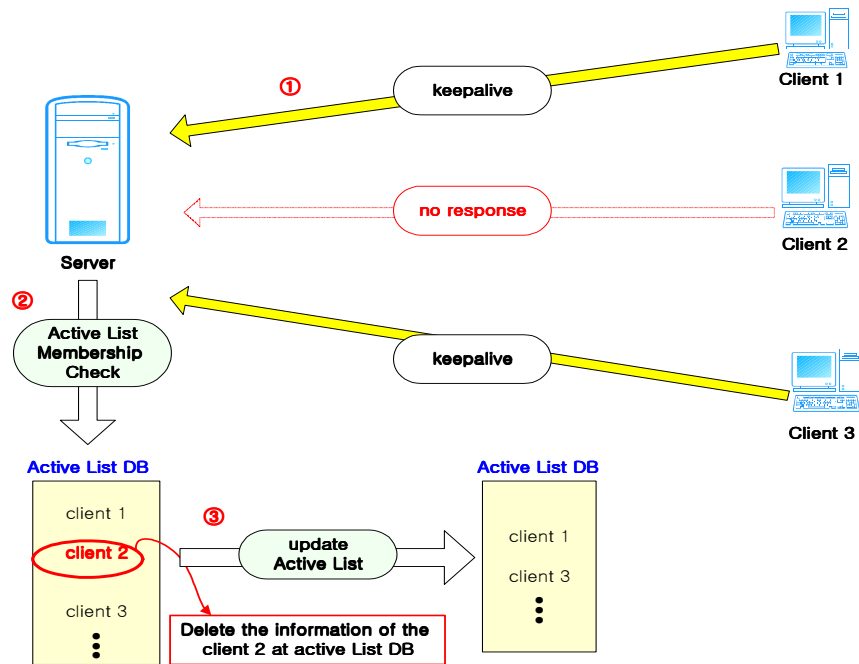


Figure 6. Membership Check

Figure 7 illustrates the membership check algorithm in the server side when the server received the **Keepalive** report from the client. If the MM server receives the **Keepalive** report from the client, the server examines whether the client is the

member in the registered group member list. If the client is in the registered group member list, then the server examines the active group member list again. If the client is in the active group member list, updates the time-out value for the member, and if the client is not in the active group member list, initializes the time-out value for the member after adding to the active group member list and updates the list.

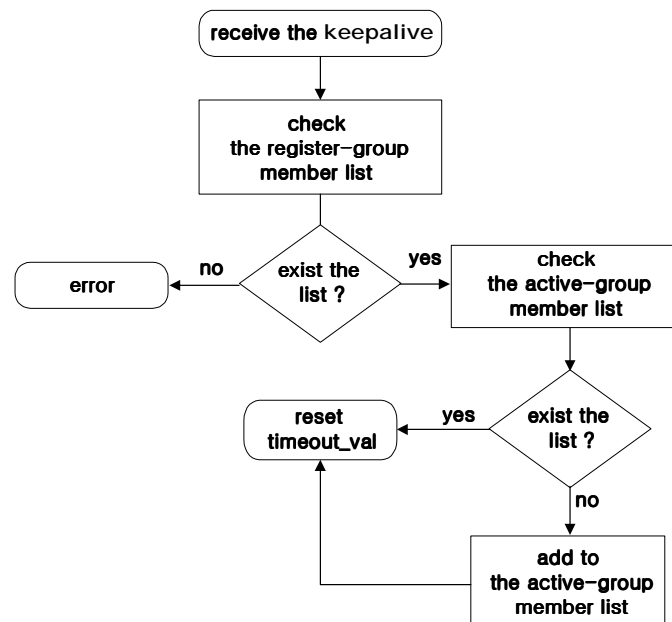


Figure 7. The algorithm of the membership check in the server

4. Test environment and Implementation

In this paper, we experimented using 4 PCs that have the operating system of the Red Hat 7.3. Also, used the Mplayer used in ECTP distribution as the application program for the MM client's test. ECTP in each client who tests is operating basically.

When new participant joins to the session, the MM starts, and if a sender (session creator) sends the data using "Mplayer server" over ECTP, "Mplayer" which is a application in the client starts in each clients' computer which participates. If receives all data through "Mplayer" in the client side, the client sends the **secession leave** message to server and the GMP server who receives the **secession leave** message removes the client from the active group member list and updates the active group member list.

4.1 Example of the application for MM (Join the Session)

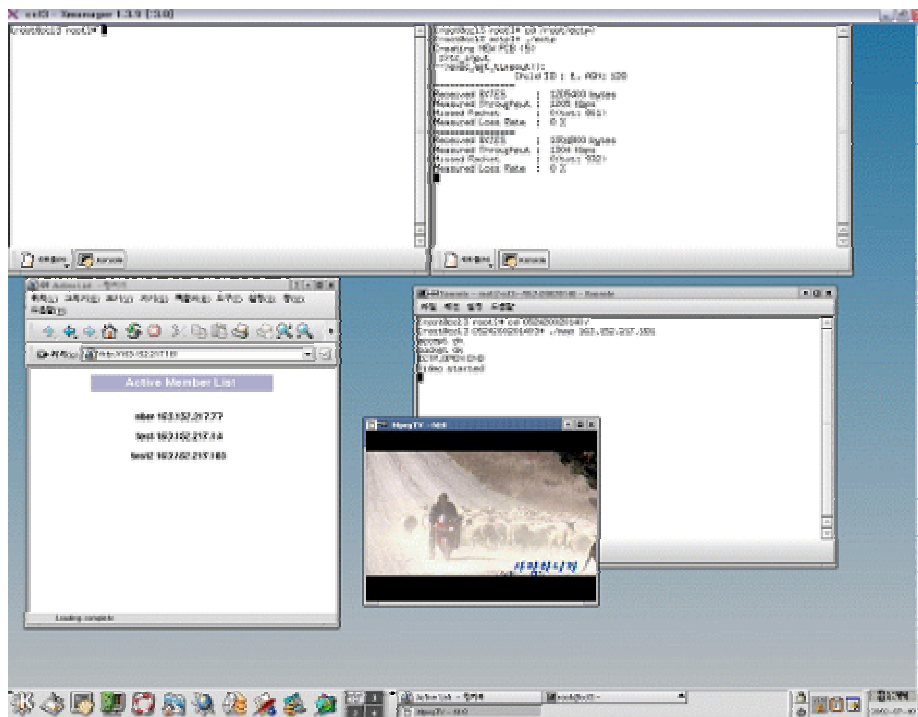


Figure 8. Mplayer on the client side

In the Figure 8, the sender and receivers are joining the session and sender is sending the data to clients over ECTP. When the sender starts to send the data, the Mplayer window begins to appear, and receives the data with above Figure 8. Also, when the client joins the session, the client can see the active group member list. This active group member list is updated periodically.

4.2 Example of the application for MM (Leave the Session)

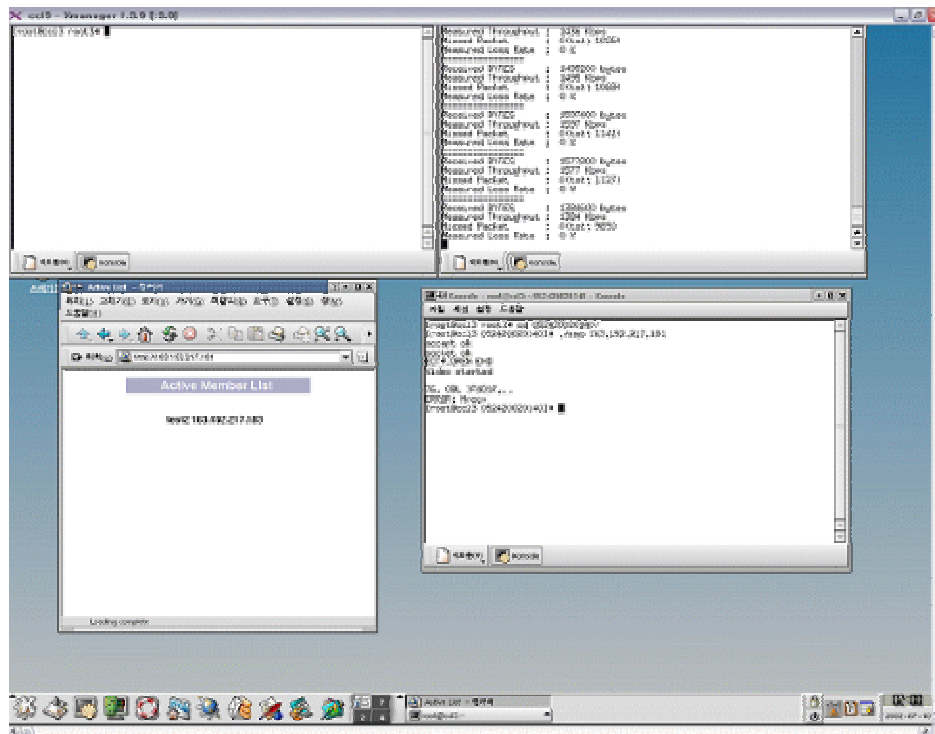


Figure 9. After the client leaves the session.

If receiver been joining to the session with Figure 9 receives all data from the sender,

the application window (Mplayer) is ended automatically. The receiver sends the **session leave** message to the MM server at the same time that application window is ended, and the MM server who receives this message removes the receiver from the active group member list. With Figure 9, we can confirm the active group member list that is updated from the active list group.

5. Conclusions and Future Work

In this paper we design and implement the efficient membership management protocol for multicast communication. Conventional multicast transport protocols have not included dynamic mechanism for the group management according to the membership information. It is difficult to operate the function properly according as the number of participants to join the session becomes much and the membership information amount is increased. So, we made an efficient membership management by a separate membership management function and made the membership management server manage the membership.

The membership management function that implements in this paper as a server/client model, the membership management server manages participants' membership by each session. Also, the client sends a **keepalive** message to server periodically, and

the server examines the membership list according to client's **keepalive** message and updates the membership information. And if sender finishes sending data transmission, the MM client sends the **session leave** message to server and leave in the session automatically. We used the Mplayer that is included in ECTP distribution to test an efficient membership management for multicast communication.

In future works, we will calculate the cycle that checks the membership exactly. As that this cycle is fixed in this paper, must look for a suitable cycle in multicast communications through many experiment. Also, we will implement the management program for server and can be used in actuality multicast communication.

Reference

- [1] M. Handley and S. Floyd, et al. "The Reliable Multicast Design Space for Bulk Data Transfer", RFC 2887
- [2] B.Whetten, L.Vicisano. et al., "Reliable Multicast Transport Building Blocks for One-to-Many Bulk-Data Transfer", IETF Internet Draft, draft-ietf-rmt-buildingblocks-03.txt, October, 2000.
- [3] ITU-T Rec. X.605, ISO/IEC 13252, "Enhanced Communications Transport Protocol", 1998

- [4] K. McCloghrie, et al. "Internet Group Management Protocol MIB", RFC2933, October, 2000.
- [5] Fenner, W., "Internet Group Management Protocol, Version 2", RFC2236, November 1997.
- [6] Rose, M. and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, RFC1155, May 1990.
- [7] M. Handley, V. Jacobson, "RFC2327: SDP: Session Description Protocol", 1998
- [8] M. Handley, C. Perkins, E. Whelan, "RFC2974: SAP: Session Announcement Protocol", 2000
- [9] M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg, "RFC2543: SIP: Session Initiation Protocol", 1999
- [10] ITU-T Rec. X.605 | ISO/IEC 13252, "Enhanced Communication Transport Service", 1999
- [11] Su-Jin Lee, Jae-Eun Kim, Eunsook Kim, Shin Gak Kang, Hyun-Kook Kahng, "Implementation of the Web-based Session Management Protocol for the Multicast Communication", Proceedings of the 28th KISS Fall Conference, Vol. 28
- [12] ITU-T Temporary Document 2051/Rev.1, "Group Management Protocol", 2002.