MULTICAST MOBILITY SUPPORT IN SESSION LAYER USING SIP

Xiuxia Yang

A Thesis

In

The Department

Of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Applied Science

(Electrical and Computer Engineering)

at

Concordia University

Montreal, Quebec, Canada

September 2004

©Xiuxia Yang

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-94716-5
Our file Notre référence
ISBN: 0-612-94716-5

The author has granted a non-exclusive license allowing the Library and Archives Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

ABSTRACT

MULTICAST MOBILITY SUPPORT IN SESSION LAYER USING SIP

Xiuxia Yang

Due to the development of computer hardware technology, the use of portable computers has become more popular. Research workers have become more and more interested in wireless mobility related to these new hardware technologies. Much work focusing on Mobile IP has been done. However, there exist some issues such as "Triangle Routing" in Mobile IP. It has been found that SIP-based (Session Initiation Protocol) mobility has advantages that can be used to avoid the issues in Mobile IP. Based on this idea, this thesis first introduces the previous work related to mobility as well as mobility with multicast, and presents some issues that exist in IP mobility with multicast. Then we provide an approach called "Multicast Mobility Support in Session Layer Using SIP", which is independent of the IP layer and automatically avoids the issues in IP layer. The structure of the SIP mobility with multicast is presented and a proposal to extend SIP functions for supporting SIP multicast mobility is provided. Based on the system model built by using SDL (ITU Specification and Description Language), we used ObjectGeode tool set to simulate the proposed idea presented in this thesis. This work may be helpful to shift the concept of multicasting and mobility from IP layer to session and application layers for real time multimedia.

ACKNOWLEDGEMENTS

I wish to express my deep gratitude to my supervisor Dr. Anjali Agarwal for her invaluable guidance, advice and financial support throughout the course of this work.

I would like to thank my colleague Kangbing Wang. Smart suggestion and helpful discussions are unforgettable and meant a lot during my study and research.

Finally, I would like to express my thanks to my husband and my daughter for their love and encouragement during these years.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF ACRONYMS

AMT	Automatic Multicast Tunneling
ALM	Application Layer Multicast
AS	Administrator Server
CBT	Core-Based Trees
CH	Correspondent Host
DHCP	Dynamic Host Configuration Protocol
DVMRP	Distance Vector Multicast Routing Protocol
DNS	Domain Name System
FSM	Finite-state Machines
FA	Foreign Agent
FN	Foreign Network
3GPP	3rd Generation Partnership Project
3GPP2	3rd Generation Partnership Project 2
GPRS	The General Packet Radio Service
HA	Home Agent
HN	Home Network
ITU-T	International Telecommunication Union-Telecommunication
ISP	Internet Service Provider
IGMP	Internet Group Management Protocol
IETF	Internet Engineering Task Force
Mbone	Multicast Backbone
MH	Mobile Host
MOSPF	Multicast Open Shortest Path First
MWIF	The Mobile Wireless Internet Forum
MSC	Message Sequence Charts
PSTN	Public Switched Telephone Network
PIM	Protocol Independent Multicast
RS	Root Server
RTP/RTCP	The real-time transport protocol/the RTP control protocol
RPF	Reverse Path Forwarding
SIP	Session Initiation Protocol
SDL	Specification and Description Language
TCP	Transmission Control Protocol
UMTP	UDP Multicast Tunneling Protocol
UML	The Unified Modeling Language
URL	Universal Resource Locators
UDP	User Datagram Protocol

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The hardware technology development has made an explosive growth in the number of portable computers sold. This has created an explosion of interest in problems related to the mobile computing. Multicast operation with mobile hosts is not common today, but it may become widespread in the future. Much more work has been done on mobility support multicast. It also is a new era for the real-time multimedia that is receiving increasing interest.

Multicasting services are increasing in popularity as service providers take advantage of multicasting solutions to efficiently distribute content to a large number of users. For example, multicasting can be used to provide streaming content such as news or video to many subscribers. Additionally, multicast services could be used to provide location-based information such as traffic report and advertisement tailored for users in a specific geographical area. While these application gain performance when the underlying network supporting them have multicasting capabilities, the networks are not consistent in this capability across the entire infrastructure reaching the user. This is especially true when IP multicasting is not ubiquitous to all networks. The quality of multicast services also becomes problematic when service providers consider the wireless

network environment and the maintenance of multicast sessions to users moving through various access network types.

Multicast services are supported currently by the deployment of techniques such as IP multicasting in the network to efficiently handle these applications. IP-layer multicast solutions for wired networks have been thoroughly investigated for non-mobile users [1-6]. Although multicast-related work has been done in wired networks, wireless networks introduce other considerations. It is desirable from the users' point of view to maintain multicast services from any point of attachment to the network. For example, users in cars moving through different access networks will desire the capability to continuously receive multicast streams and location-specific information.

Research relevant to support multicasting for mobile host has been done specifically for Mobile IP [7]. The bi-directional tunneling solution for Mobile IP puts the burden of forwarding the multicast packets to mobile users on the Home Agent (HA). However, when an HA has a number of users in the same multicast group visiting the same foreign network (FN), tunneling multiple multicast packets to the foreign network is inefficient. To avoid the duplication of multicast packets, remote subscription has been proposed whereby a user desiring to join a multicast group will do so in each visited network through the Foreign Agent (FA). However, this requires that after every handoff the user must rejoin the multicast group and the multicast tree used to route multicast packets will be updated after every handoff to track multicast group membership. In addition, the user may lose their identities [8] [9] [10].

For IP multicast, service providers require the ability to efficiently multicast to mobile users through various networks. From the service provider's point of view, this

requires some understanding of the multicasting capabilities in the various access networks over which service is provided. In most cases, the service provider will have limited knowledge and control over the networks. Besides the issues discussed above Mobile IP is still struggling with the problems of triangular routing, triangle registration, encapsulation overhead and need for permanent home address for unicast and with the problems of the tunnel convergence problem and multicast router requirement on home network or foreign network for multicast as will be discussed in the following section.

Current multicast solutions rely on knowledge and control of the network routers to perform multicast routing. However, multicast deployment has not been completed and is not ubiquitous to all wired and wireless networks. Tunneling techniques, such as Automatic Multicast Tunneling (AMT)[11] and UDP Multicast Tunneling Protocol (UMTP) [12], have been proposed to route IP multicast packets to users across non-multicast-enabled networks. For UMTP, there are some advantages such as: no triangle routing inefficiency, lower handoff latency, and no additional infrastructure. Although UMTP enables a host to establish a connection to the Mbone (Multicast Backbone) by tunneling multicast UDP datagram inside unicast UDP datagram, the end subnet still requires multicast routers for multicasting datagram to local group members. In addition, the multicast package must all be UDP. This means it does not support TCP. The UMTP implementation requires to know each (group, port) that the multicast-based application uses. However, under the mobility situation the group of multicast-based application are often created and destroyed dynamically. The (group, port) is not specified in advance, but instead is determined by the application itself. In this case, UMTP could not be used. Besides, the act of tunneling a multicast packet by using UMTP changes its source

address, so the application, which must use source addresses to identify these original data sources, could not use UMTP. Because of these disadvantages we prefer other approaches to UMTP like multicasting in application layer. A straightforward advantage of using multicast in the application layer is that multicast applications can be executed in networks that do not support IP layer multicast. Application Layer Multicast (ALM) offers accelerated deployment, simplified configuration and better access control at the cost of additional (albeit small) traffic load in the network, because in an ALM session, only the source and the destination nodes of the multicast session can sit on the end points of the paths. All the participants are connected via a virtual multicast tree, that is, a tree that consists of unicast connections between end hosts [13].

In order to solve the problems appearing in Mobile IP and find a better way to cooperate the development of mobile communication, we propose a solution that provides an infrastructure to evolve with the multicast capabilities of the network based on application-layer SIP (Session Initiation Protocol) multicast. SIP is the application layer protocol that already supports the personal and terminal mobility. Also, application layer mobility does not require any changes to the operating system of any of participants and can be deployed widely much easier than mobile IP.

Traditional SIP multicast support applies multicasting at IP layer. So, all problems that exist in IP multicasting also appear in this situation such as the multicast routers requirement, the lack of wide scale multicast network deployment and the issue of how to track group membership. Also the main drawback of IP multicast protocols is that they are developed for multicast parties whose members are topologically stationary and they do not consider extra requirements to support either topologically mobile senders or

mobile sources. In order to address these concerns, Application Layer Multicast [14] [15] has already been proposed as another multicast approach. Under this background, we propose a new approach to implement SIP Mobility Multicast in the application layer.

1.2 REVIEW OF MOBILE IP

Mobile IP support has been standardized by the IETF (Internet Engineering Task Force). Mobile IP introduces the following functional entities [7][16]: *Mobile host (MH)* is a host or router that changes its point of attachment from one network or sub-network to another, without changing its IP address and can continue to communicate with other Internet nodes at any location using its permanent IP address. *Correspondent host (CH)* is a peer with which a mobile host is communicating. *Home agent (HA)* is a router on a mobile host's home network that delivers datagram to departed mobile hosts, and keeps track of the current location of the MH. *Foreign agent (FA)* is a router on a MH's visited network that cooperates with the home agent to complete the delivery of datagram to the mobile host while it is away from home network. *Home address* is an IP address that is assigned for an extended period of time to a mobile host and remains unchanged regardless of where the node is attached to the Internet. *Care-of address* is the termination point of a tunnel toward a mobile node for datagram forwarded to the mobile host while it is away from home. There are two different kinds of care-of address: a foreign agent care-of address is an address of a foreign agent with which the mobile host is registered and a collocated care-of address is an externally obtained local address that the mobile host has associated with one of its own network interface. *Home network (HN)* is a network (possibly virtual) having a network prefix matching that of a mobile

host's home address. Foreign network (FN) is a network other than a mobile host's home network to which the mobile host is currently connected.

Mobile IP normally performs three related operations that are Agent Discovery, Registration and Datagram tunneling. Mobility agent (HA/FA) make themselves known by sending agent advertisement messages. After receiving an agent advertisement message, a MH determines whether it is on its home network or a foreign network. A mobile host basically works like any other node on its home network when it is at home. When a mobile host moves away from its home network, it obtains a care-of address on the foreign network by soliciting or listening for agent advertisements, or contacting Dynamic Host Configuration Protocol (DHCP). The mobile host registers each new care-of address with its home agent, possibly by way of a foreign agent. The datagram sent to the mobile host's home address by correspondent host are intercepted by its home agent, tunneled by its home agent to the care-of address, received at the tunnel endpoint (at either a foreign agent or the mobile host itself), and finally delivered to the mobile host. On the other hand, datagram sent by the mobile host are generally delivered to their destination using standard IP routing mechanisms, not passing through the home agent. Figure 1-1 illustrates the routing of datagram to and from a mobile host away from home network, while the mobile host has registered with its home agent. Here, the mobile host is presumed to be using a care-of address provided by the foreign agent. A datagram to the mobile host arrives on the home network via standard IP routing. The datagram is intercepted by the home agent and is tunneled to the care-of address, as depicted by the arrow going through the tube. The datagram is de-tunneled and delivered to the mobile host. For datagram sent by the mobile host, standard IP routing delivers each to its

destination. In the Figure 1-1, the foreign agent is the mobile host's default router through which it sends datagram to CH.

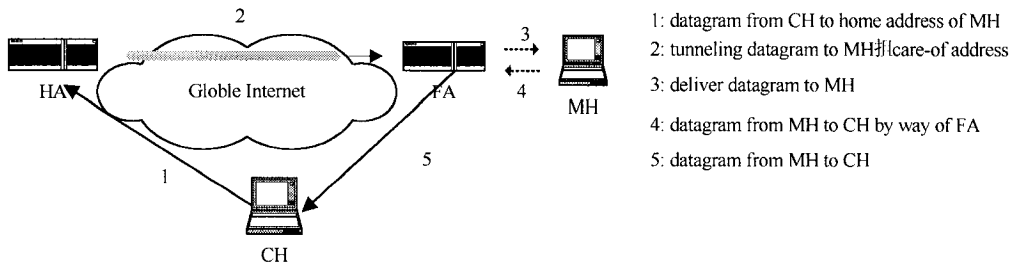

Figure 1-1 Mobile IP

1.2.1 Limitation of Mobile IP with Unicast

As noted in Figure 1-1, datagram going to the mobile host have to travel through the home agent when the mobile host is away from home network, but datagram from home host to other stationary Internet hosts can instead be routed directly to their destinations (Figure 1-2). This asymmetric routing, called *triangle routing,* is far from optimal especially in cases when the CH is very close the MH.

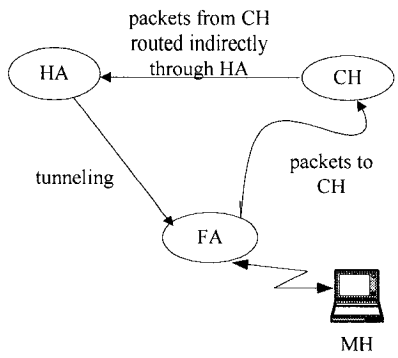

Figure 1-2 Triangle Routing

Route Optimization solves the triangle routing problem. The basic idea underlying route optimization is that the routes to mobile hosts from their correspondent nodes can be improved if the correspondent host has an up-to-date mobility binding for the mobile host in its routing table. Here, *Mobility Binding* is the association of a home address with a care-of address along with the remaining lifetime of that association. The basic operation of route optimization is: (1) The Correspondent host may send a *binding request.* (2) The home agent may send an authenticated binding update containing the mobile host's current care-of address. (3) For smooth handoffs, the mobile host transmits a binding update and has to be certain that the update was received. Thus it can request a *binding acknowledgment* from the recipient.

Although route optimization shortcuts the data path, but it still requires binding updates tunneled through the home agent, adding the handoff delays. Besides, they require changes in operating system of the correspondent host, including authentication mechanism.

Regardless of the problem of data path, Mobile IP encapsulation adds between 8 and 20 bytes [17] [18] of overhead. Packet header overhead is significant for low bit-rate packet voice where payloads tend to be very short.

In order to obtain the benefits of Mobile IP, a user has to have a permanent home IP address and needs to convince his ISP to offer home agent services. The fact is most customer devices do not have a fixed IP address but rather acquire one dynamically via DHCP only when log in. Thus, a customer is at the mercy of his ISP to obtain mobility services.

In summary, the disadvantages of Mobile IP are:

- Need two addresses: home and visiting.
- Triangle routing.
- Encapsulation overhead for voice: 8-20bytes/packet for a 50-byte payload.

1.2.2 Mobile IP with Multicast

IP multicast is a technology that saves bandwidth and reduces traffic by delivering a single stream to thousands of recipients (Figure 1-3). In order to perform multicasting, the basic requirement is to add multicast router whose function is to forward incoming IP multicast packets out to all interfaces that lead to members of the multicast group. Every member can dynamically join or leave the group by using IGMP (Internet Group Management Protocol). Multicast routing protocols, such as DVMRP (distance vector multicast routing protocol), PIM (Protocol Independent Multicast), CBT (Core-Based

Trees) and MOSPF (multicast open shortest path first), exist to implement the multicast service [19] [20].

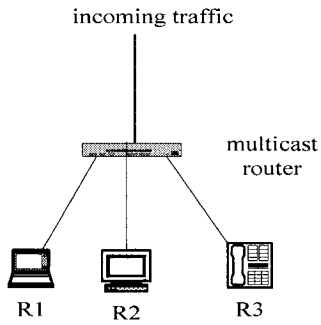

Figure 1-3 Multicasting

Mobile IP multicast uses the IGMP Join message (Host Membership Report) to join the group and IGMP Leave message to leave the group as is done in IP multicast communication. The current version of Mobile IP proposes two approaches to support mobile multicast [8] [21], called *remote subscription* and *bi-directional tunneling.*

In remote subscription, the mobile host re-subscribes to its desired multicast groups while at a foreign network. The remote subscription approach is simple and works well if the mobile host spends a relatively long time at each foreign network. It has the advantage of offering good routes for delivery of multicast datagram to mobile hosts. However, the approach implicitly presumes that mobile hosts are only recipients of multicast messages or that they have a co-located address on the foreign network. If the mobile host sends the multicast datagram with its home address as the source, the incoming interface check (Reverse Path Forwarding (RPF) check) [19] [22] of most multicast routing algorithms may discard datagram intended for the multicast group. Finally, the approach assumes the existence of a multicast router at the foreign network.

The fact is that this assumption may not always hold in an IP inter-network. On the other hand, without such a multicast router, multicast message delivery can be achieved only by using some form of tunneling.

With bi-directional tunneling, mobile hosts send and receive all multicast datagram by way of their home network using unicast mobile IP tunnels from their Home Agents. This approach in fact hides host mobility from all other members of the group. The disadvantages, however, are two-fold. First, the routing path for multicast delivery can be far from optimal in the worst case that the source and the recipient can be on the same network while all multicast messages between two hosts have to traverse the entire inter-network twice. Second, the approach offers limited scalability that means home agents with multiple mobile host group members away from home must replicate and deliver tunneled multicast datagram to each of them regardless of at which foreign networks they reside which may result in convergence problem.

Normally there are two ways each for Mobile IP multicast to receive and to send multicast traffic [23] [24]. The following is the summary of two ways as well as their limitations:

- **Receiving Multicasts:**
    1. The mobile node can tunnel the IGMP Host Membership Report packet to it's HA to tell the HA that it wants to join this multicast group from which to receive the multicast packets. Then HA will intercept the multicast packets and tunnel them to the mobile node. The problem is that if there are many (more than one) mobile nodes subscribing to the same group and registering with the same FA, all the tunnels from the different HA will

carry the same multicast packets and result in *Tunnel Convergence* (Figure 1-4) problem.

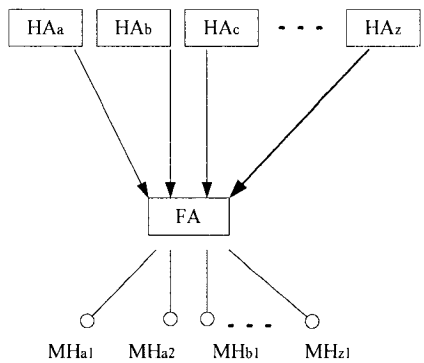

Figure 1-4 Tunnel Convergence Problem

2. The mobile node can also send the IGMP report to a multicast router on the foreign network. This router will add the mobile node to the multicast tree and route the multicast packets to it by using multicast routing protocols. Here the requirement is that the *multicast router* be present on the foreign network.

- **Sending Multicast:** the mobile node that wants to send the multicast packets need not be a member of a multicast group, and may not join the multicast group.
    1. The mobile node can tunnel the multicast packets to the HA directly. This case has a requirement that the HA should be a multicast router.
    2. The mobile node can also send the multicast packets to the multicast group on the foreign network through the local multicast router. In this case, if the mobile node uses its home address as source address, the RPF check (reverse path forwarding) would fail; if the mobile node uses its care-of

address as the IP source address, the RPF check would succeed, but mobile node would lose its identity. At the same time there must be a multicast router on the foreign network.

Because of such disadvantages for mobile IP and mobile IP supporting multicast, researchers hope to improve mobile IP protocol or to find another better way to implement mobility.

1.2.2.1 Limitation of Mobile IP with Multicast

The summary for the issues of the Mobile IP with multicast is presented in the following:

- Tunnel Convergence problem.
- Requirement for multicast router in home and foreign network.
- RPF check failing

1.3 WHY USING SIP

Several wireless technical forums, *e.g.,* 3GPP, 3GPP2, and MWIF, have agreed on SIP as the basis of the session management of signally protocol on the mobile Internet. It seems that SIP will certainly be an integral part of the mobile Internet's protocol architecture. Thus, it would be desirable to use SIP to provide means of terminal, service as well as personal mobility for all applications. The underlying rationales for seeking such a solution are:

i. The expected growth of Internet multimedia services.

ii. Strong likelihood of using SIP for supporting service and personal mobility, and

iii. The beliefs that SIP can also support terminal mobility with minimal extensions.

The advantages of using SIP for supporting mobility are that it

- Allows users to depend on their appliances rather than the network for supporting mobility on an end-to-end basis without reliance on and knowledge about abilities of network elements for packet interception and forwarding, i.e., mobile users can roam across SIP environments without concern about whether they support network layer mobility or not,

- Provides a means of route optimization and improved performance for real-time services via SIP signaling messages for address binding, registration, *etc.*, and
- Allows dealing with mobility at a semantic level above IP terminals.

In the short term, supporting mobility with SIP at the application layer can either partially replace or complement other approaches that rely on network layer mobility protocols (*e.g.*, mobile IP, GPRS).

1.4 THESIS OBJECTIVE

The objectives of this thesis are:

- Present architecture, i.e., SIP mobility supporting multicast, for supporting multicast roaming users in a mobile wireless internet whose signaling system is built upon SIP.

- Develop the detailed messages for supporting the new protocol and demonstrate its performance.
- Use SDL to simulate.

1.5 THESIS OUTLINE

The content of this thesis is organized as follows:

Chapter 2 introduces the basic ideas for SIP mobility. It includes the basic concept and network structure as well as the measures of dealing with unicast and multicast mobility.

Chapter 3 presents our approach for SIP mobility with multicast. System structure and detailed messages are given in this chapter. UML (Unified Modeling Language) sequence diagram are given here to guide to understand its performance.

Chapter 4 presents the procedures for MHs' handoff. In this chapter, we discuss various kinds of situation under which handoff may happen: whether moving happens before calling or during calling; whether the new domain where MH is moving to has group member/members or not.

Chapter 5 provides the results of simulations by means of an example. At the beginning of the chapter, it simply introduces the SDL (ITU Specification and Description Language), and then gives the system model as well as the configuration of the simulation model. The results are shown thereafter.

Chapter 6 gives the summary of the thesis. The issues and the suggestions for future work on this topic are presented.

CHAPTER 2

SIP MOBILITY

The Session Initiation Protocol (SIP) is an Internet standard that was published as RFC 2543 in March 1999 (the latest issue is RFC 3261 in June 2002). It is a text-based client-server signaling protocol used for creating, changing and tearing down multimedia sessions between two or more participants [25] [26]. In addition to being able to set up, change and release sessions, SIP can also be used to build advanced telephony service like call waiting, call forwarding, and so on. SIP also allows two or more participants to establish a session consisting of multiple media streams such as audio, video or any other Internet-based communication mechanism. The media stream for a single user can be distributed across a set of devices, e.g., specialized audio and video network appliances in addition to a workstation. The protocol is standardized by the IETF and is being implemented by a number of vendors, primarily for Internet telephony.

2.1 Review of SIP

SIP [26] [27] defines a number of logical entities, namely *user agents* and *network servers*. A *user agent* is an end system that contains both a protocol client used to initiate a call and a protocol server used to answer a call. It is used to originate and terminate requests. Examples include conferencing software or gateways to the PSTN (Public Switched Telephone Network), but also voice mail systems. Generally, user

agents are the only elements where media and signaling converge. SIP end points are addressed by SIP URLs (Universal Resource Locators) that have the form of email addresses like xx-yang@ece.concordia.ca. SIP network servers include *proxy server*, *redirect server*, and *registrar*. P*roxy server* acts as both a server and a client for the purpose of making requests on behalf of other clients. It ensures that a request is sent to another entity "closer" to the targeted user. It receives a request, determines which server to send it to, and then forwards the request, possibly after modifying some of the header fields. A SIP proxy has no way of knowing whether the next server to receive the request is another proxy server, a redirect server, or a user agent server. For this reason, SIP request can traverse many servers on their way from user agent client to user agent server. Response to a request always travels along the same set of servers the request followed, but in reverse order. *Redirect server* receives request, but instead of forwarding them to the nest hop server, it just generates responses to requests it receives and indicates where the request should be sent next. It answers the client's request using a redirect response that contains the address of the next hop server. For example, a redirect server may keep track of the user's location and then return a response indicating that location as a list of one or more SIP URLs. Typically, a SIP server implements a redirect and proxy server with information provided by a built-in registrar. *Registrar* is a server that accepts REGISTER request and stores the information it received in the request into the location service for the domain it handles. Normally the registrar is collocated with proxy server/redirect server. Depending on configuration and the specific request, the server acts as either a proxy or redirect server or a registrar.

A SIP request consists of a request line, header fields, and a message body. The various header fields contain information on call services, addresses, and protocol features. The body can contain anything. SIP defines six different method types, including INVITE, ACK, CANCEL, OPTIONS, REGISTER, and BYE. INVITE method is used to invite user(s) to a session and establish a new connection. The header fields of an INVITE request contain the address of the caller and callee, subject of the call, call priority, and call routing requests, caller preferences for user location and so on. The body of the request contains a description of the media content of the session. Usually, this body is an object described by the session description protocol (SDL), a textual syntax for describing unicast and multicast multimedia sessions. It contains information on codec, ports, and protocols to be used for sending media to the caller. REGISTER is used to transfer the information about a user's location to a SIP server. REGISTER allows a user to tell a SIP server how to map an incoming address into an outgoing address that will reach that user (or another proxy that knows how to reach that user). The body of a REGISTER message can be anything. Currently, researchers are investigating the use of simple scripts to describe the more complex programmatic name translations. Furthermore, the body of a REGISTER response can contain configuration information useful to the user agent. Such information may include additional addresses that allow for private branch exchange, whereby the server chooses the address used by each client. Beyond INVITE and REGISTER described in the above, the SIP methods also include: ACK, an acknowledgement of an INVITE and is used to confirm reliable message exchanges; BYE used to terminate a session; OPTIONS used to query SIP servers' capability, but do not set up a call; and CANCEL is used to terminate a pending request,

but does not undo a completed call. When received at a user agent server (UAS), CANCEL has no effect if the UAS has already answered the call. If the UAS has not answered, CANCEL indicates that it should not respond because the call has effectively been cancelled. This does not, however, prevent a UAS from answering the call request. It is simply an optimization. Figure 2-1 shows an example of SIP session setup and termination [16].

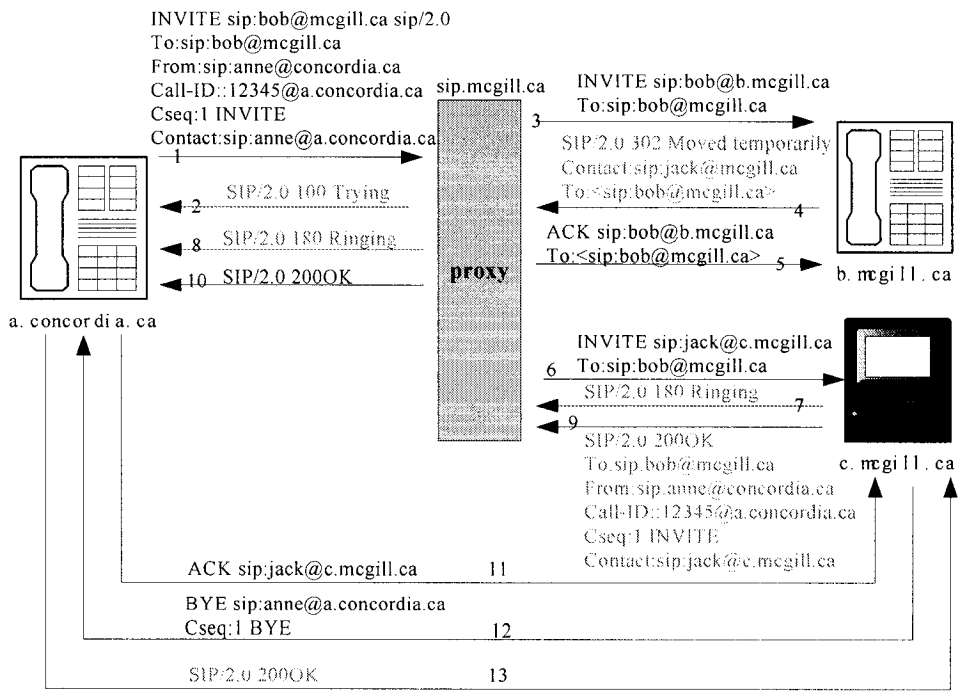

Figure 2-1 Example of SIP Session Setup

2.2 SIP Mobility

From Chapter 1, we see that Mobile IP is struggling with the problem of triangular routing. The route optimization solves this by sending binding updates to inform the sending host about the actual location of the mobile host. This solution still has several problems like the requirement of changes in the IP stack of the correspondent host in order to encapsulate IP packets and store care-of addresses of the foreign agent or mobile host, and the extra delay added before the correspondent host finds out where to send the packets. Therefore, there is a need to introduce mobility awareness on a higher layer. The application layer protocol SIP already supports personal mobility, and the changes needed to support device mobility are minor. In the following, it is discussed how mobility support in SIP can improve the performance for real-time services in wireless networks. Also, application layer mobility does not require any changes to the operating system of the participants and thus can be deployed widely much easier than Mobile IP.

2.2.1 SIP Personal Mobility

Personal mobility is the ability of an end user to originate and receive calls and access subscribed telecommunication services on any terminal in any location, and the ability of the network to identify end users as they move [3][28][29]. Personal mobility is based on the use of a unique personal identity. SIP support advanced personal mobility services, an example of which is shown in Figure 2-2 [29]. A user of the system, John, maintains an office at Laurent Technologies location. In addition, John is a researcher at

Concordia University, where he has another office. John publishes a single IP telephony phone address for himself: john@laurent.com. When John is at Concordia University, he sends a REGISTER message to the Laurent SIP server (1), listing his Concordia address-john@concordia.ca as forwarding address. Once at Concordia, John registers his office-john@office.concordia.ca with the Concordia registration server (2). Later in the day, jack@mcgill.ca places a call to john@laurent.com. Using DNS (Domain Name System), the caller resolves Laurent.com to the address of the Laurent SIP server, which receives the call request (3). The server checks its registration databases (4) and decides to forward the request to john@concordia.ca. To do so, it looks up concordia.ca in DNS and obtains the address of the main Concordia server. It then forwards the request there (5). As soon as the request arrives, the Concordia server looks up john@concordia.ca in the registration database (6) and determines that he is at the office of Concordia University. The server then sends a call request to the office, causing the office phone to ring (7). John then answers the phone in his office, sending and acceptance response back to the Concordia server (8). Having now received the response, the Concordia server forwards the call acceptance back to the Laurent server (9), which forwards the request to the original caller (10). At this point, call transactions may proceed directly between the caller and John without passing through the intermediate server (11).

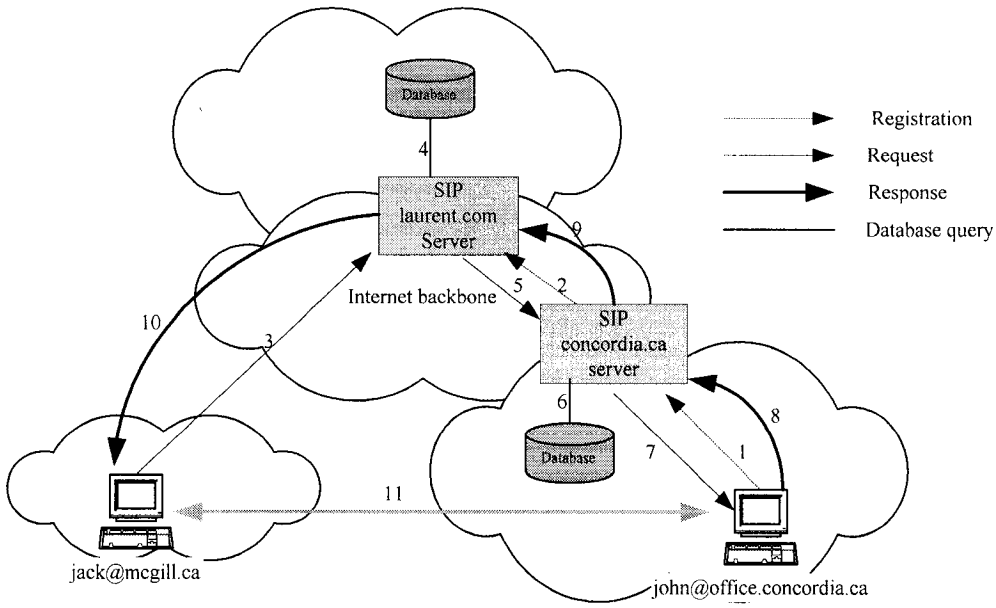

Figure 2-2 An Example of SIP Personal Mobility

2.2.2 SIP Terminal Mobility

Terminal mobility allows a device to move between IP subnets, while continuing to be reachable for incoming requests and maintaining sessions across subnet change. A subset of terminal mobility, being able to be reached for new sessions after subnet changes, requires only DHCP (Dynamic Host Configuration Protocol) and dynamic DNS. Using SIP for mobility trades generality for ease deployment. SIP-based mobility is less suitable for TCP (Transmission Control Protocol) based applications, but does not require adding capability to existing operating system nor the installation of home agents or dynamic DNS update [30] in the user's ISP (Internet Service Provider). Here, SIP mobility means SIP terminal mobility. SIP mobility happens at two stages: pre-call and mid-call [16] [31]. Pre-call means that moving happens before session is setup. Mid-call

means that moving happens in the middle of communication or say after session is setup. Figure 2-3 and Figure 2-4 are the examples of SIP mobility with its basic messages and entities for unicast application.

The pre-call mobility is the easiest part of SIP mobility. Within the pre-call mobility a MH has already moved to a foreign network before receiving or making a call. The MH re-registers with its home registrar each time it moves. Figure 2-3 shows the working of the SIP pre-call mobility. A correspondent host (CH) sends INVITE MH message (1) to its home network. MH has moved but has registered with its home network. Redirect server responses the INVITE message providing the new server address of MH (2). CH sends INVITE MH message (3) to foreign network where MH is located. Upon receiving INVITE, MH sends 200OK message to CH (4). They begin media communication (6) after CH confirms with ACK (5).

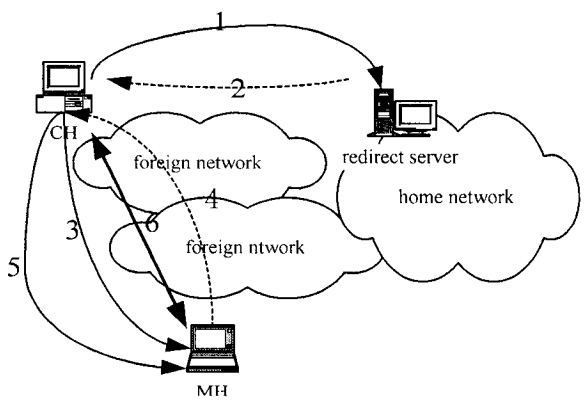

Figure 2-3 SIP based Pre-call Mobility

SIP based mid-call mobility is shown in Figure 2-4. Within mid-call mobility a MH moves after it receives or makes a call. As soon as MH knows it has moved, it sends a new INVITE message (1) to CH with its new location information, without going through any intermediate SIP proxies (a SIP proxy will be traversed if it has requested to be part of future signaling messages by inserting a Record-Route header during the initial call setup). CH accepts it by responding with 200OK message (2). MH confirms with ACK (3) and they continue to communicate with each other (4). After setting up communication with CH, the MH registers with its home registrar (5). If new foreign local proxy server cannot send REGISTER message to home registrar directly, it has to send REGISTER message to home server through the outbound proxy server (Figure 2-4 step 5′ and step 6′). An outbound proxy can be manually configured or learned through auto-configuration protocols. A UAC (user agent client) is recommended to send messages to the location indicated in the first Route header field value instead of sending all messages to the outbound proxy. It allows endpoints that cannot resolve the first Route URI to delegate that task to an outbound proxy [26].

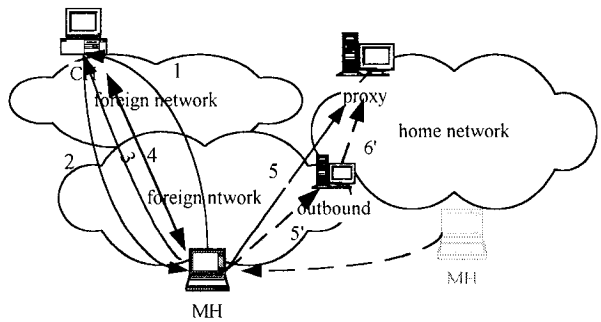

Figure 2-4 SIP based Mid-call Mobility

The occasion discussed above is focused on the situation under which CH is static. If CH moves, it also sends REGISTER message to its home proxy server. MH simply sends INVITE request, which the SIP server infrastructure then routes to the SIP server of the CH's home link following the standard SIP procedures. Assuming that the SIP server on the home link is a redirect server, it returns the address that the CH has registered with. The MH then redirects the INVITE request to the CH directly. If the CH accepts the invitation, it exchanges future SIP messages as well as any other data directly with the MH from that moment on.

2.2.3 Hierarchical Registration

By default, registrations are sent to the "home" registrar, as explained normally. Thus, any location change causes a SIP REGISTER request and response to be sent. Although SIP signaling is likely to use a higher-speed network, the signaling traffic is still undesirable. Within SIP, registrations can be forwarded just like other requests as shown in Figure 2-5 [16]. In the figure, Anne, with a home in Quebec, visits Montreal. Each time she moves, she sends REGISTER request towards her home registrar through the outbound proxy in Montreal. For the REGISTER message originating in Laval, the outbound proxy makes a note of the registration and then forwards the request to the normal home registrar after modifying Contact in the registration to point to it. After Anne moves to Dorval, the REGISTER update hits the same registrar (MTL). It recognizes that Anne is already in Montreal and does not forward the request. It updates the Contact field to point to Dorval. A call from anywhere first reaches the QC proxy

server that forwards the request to the MTL proxy server that in turn forwards it to Anne's correct location.

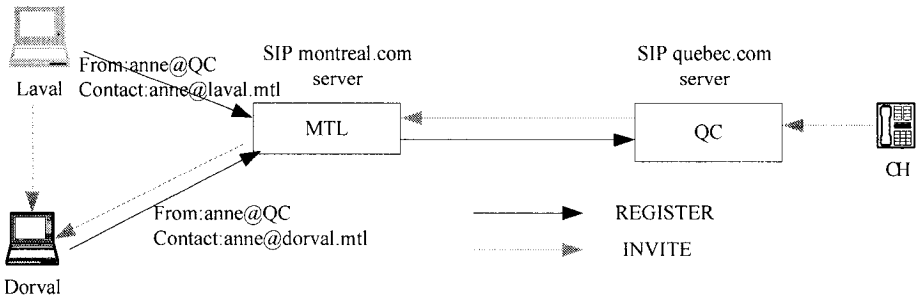

Figure 2-5 Hierarchical Registrations in SIP

Here the example is focusing on the personal (Anne) mobility. For the terminal mobility, the concept of hierarchical registration is the same as above. So we do not repeat it again.

2.2.4 Issues of SIP Multicast Support

Much work has been done for SIP mobility unicast support. This section identifies problems that may exist with multicast communication.

We assume that the mobile host and the correspondent host communicate via an IP multicast group of which both hosts are the members. When the mobile host moves to another link, it gets a new IP address. This means that it will have to leave the multicast group with its old IP address and join the group with its new IP address. However when

mobile host detects that it has moved to another link using router advertisements [32], it will have no way to contact its old multicast router and deliver the IGMP (Internet Group Management Protocol) unsubscribe message to it. This means that multicast router on the mobile host's old link keeps on multicasting message destined for the mobile host, if there are other mobile host on the old link that happen to be a member/members of the same multicast group. However, when the mobile host is the last one on the old link that was receiving the multicast message from the multicast group, the multicast router on the old link will keep on sending the multicast message on the old link while there are no hosts that want to receive them. This is a waste of the bandwidth, especially on the wireless link. This sending will not stop until the multicast router on the old link finds out that there are no nodes on the link that are interested in receiving packets from the multicast group. To solve the above problem, the mobile node should be able to send the IGMP unsubscribe message for its old IP address to the old multicast router. This requires the mobile node to record its old IP address as well as the IP address of the old multicast router. The mobile host should be able to send this IGMP message, which only carries IP multicasts address, to the old multicast router by a tunnel. Finally, the old multicast router needs to be able to act as a representative of the mobile host. It should be able to take the unsubscribe message from tunnel and transmit them onto the mobile link on behalf of the mobile host.

When it is a problem to gracefully leave a multicast group on the old link, it is easy to complete the handoff from the old link to the new link. All the mobile host has to do is to join the same multicast group by using its new IP address on its new link. Here we assume that the router of the new link is a multicast router. Unlike the unicast

situation, the mobile host does not need to send an INVITE message to the correspondent host, because the correspondent host is not aware of the identity of the host that it is communicating with anyway.

2.3 SUMMARY

In this chapter, we introduce the work that has been done recently in the literature related to SIP. It includes the basic SIP concept, the SIP structure and how it works. At the same time we provide the issues about the SIP mobility with multicast and explain the reason why propose an approach of application layer multicast. Based on these ideas, we develop a new method for SIP based mobility with multicast. This will be introduced in the following chapter.

CHAPTER 3

SIP-BASED MOBILITY WITH MULTICAST

We see in Chapter 1 that there exist problems such as triangular routing for IP unicast mobility. Though route optimization is adopted to solve the triangular routing problem, it brings new problems such as the binding update delay and the changes required in the operating system of the correspondent host. For the IP multicast mobility, the problems are that the multicast router is required and of tunneling convergence. Another problem is which address (home address or care-of address) should be used as source address when MH wants to send multicast packets to CHs. If mobile host uses home address as the source address, mobile host cannot pass the RPC checking and the packets would be dropped. If care-of address is used as source address, the mobile host may lose its identity. That results in research work that has been done in this thesis on the application layer to provide SIP based multicast mobility. SIP already supports personal mobility. The difference between the personal mobility and the terminal mobility is the frequency with which MHs register at different locations [23]. SIP works in application layer, it therefore does not need any changes in the operating system of correspondent host, and is thus easier to deploy in the system. As seen in Chapter 2, the traditional SIP multicast mobility support applies the multicast protocol in IP layer to implement multicasting. There still exist the problems as discussed in Section 2.2.4 as well as the problem of multicast router requirement.

The work presented in this Chapter focuses on the SIP multicast mobility support to avoid the traditional multicast protocol of IP layer, and to move the multicast functionality to application layer. This avoids the issues discussed in Chapter 1 and Chapter 2.

In this Chapter, we first propose the structure of SIP-based Multicast Mobility. We add the new fields "join" and "leave" to REGISTER message and discuss the detailed messages such as REGISTER_J, REGISTER_L to register, INVITE for session setup and BYE for session termination. In order to provide a clear presentation, we also provide network diagram as well as UML diagram.

3.1 PROPOSED SIP-BASED MULTICAST MOBILITY STRUCTURE

To support SIP Mobility for multicast applications, we define a new logical entity named "Root Server (RS)" that is responsible for forming a multicast tree in the local domain and to multicast packets to the group members. Figure 3-1 shows the network structure in order to provide SIP Mobility with Multicast. Here the basic entities are the same as SIP Mobility with unicast such as proxy server, redirect server and registrar. In addition, there are RSs to implement the multicast service. Every MH who wants to join a multicast group for receiving multicast packets must register with RS through its local proxy server by using REGISTER_J message. We extend REGISTER method to REGISTER_J and REGISTER_L method by adding "join" and "leave" fields for joining and leaving a multicast group as described in the Section 3.2.

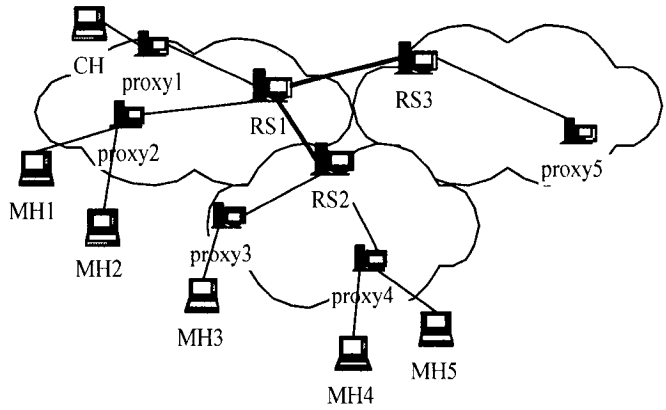

Figure 3-1 SIP Multicast Mobility Structure

3.1.1 Forming the RSs Multicast Tree

An RS is active if there is a member of the multicast group in its domain. When a root server is activated, it will join the RS multicast group by sending the RS_REGISTER_J request with "join" field to the Administrator Server (AS) that is assigned by an ISP to administer the RSs multicast tree forming. If the RS is the first server to send the RS_REGISTER_J request with "join" to AS, the AS informs it to be the source of the RS multicast tree. If the RS is not the first one to send the RS_REGISTER_J request to AS, The AS will inform it about the source of the multicast tree and redirects it to the source server. When a RS has no group members in its domain, it sends the RS_REGISTER_L request with "leave" field to the source root server for leaving the multicast tree. If the source root server has no group member in its domain and other domains, it will send RS_REGISTER_L request to leave the AS. Figure 3-2 is an example of the root server multicast tree. In this figure, there are four domains and

each domain has its local group members (mobile hosts) and it is up to the source RS to apply any multicast routing protocol to obtain its multicast tree. Figure 3-3 provides the sequence diagram for forming multicast tree of root servers. Here is only the general idea. The detail about forming the multicast tree of root servers does not belong to the scope of this research.

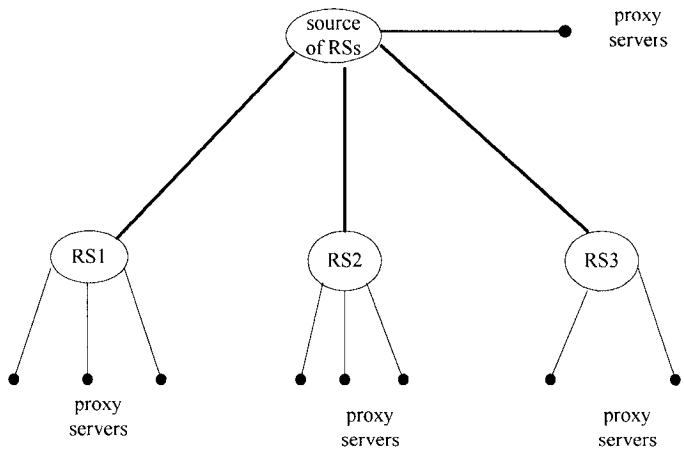

Figure 3-2 Sample of RS Multicast Tree

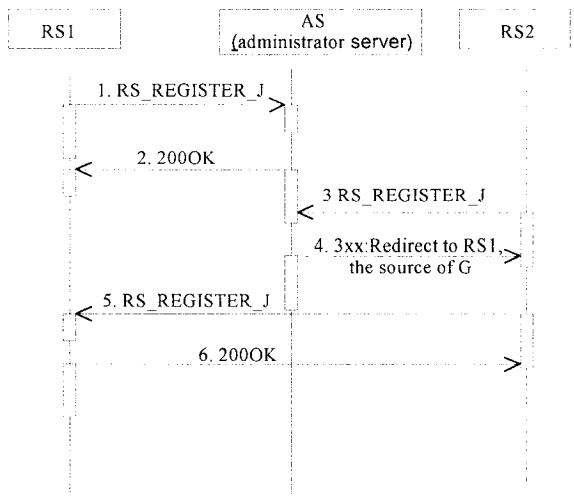

Figure 3-3 Sequence Diagram for Forming RS Multicast

We assume that all domain RSs exist and every RS knows the location of it's AS assigned by ISP. All local servers know the location of its RS in its domain either manually or auto-configured. The first RS that has the first group member will be the source of the multicast group.

3.2 MOBILE HOST REGISTRATION PROCESS

When a mobile host (MH) moves to a foreign network (FN) from its home network (HN), first thing it needs to do is to register with local registrar that is normally collocated with proxy server. The purpose of registration is to "join" the multicast group by registering with RS through the local proxy server for multicast mobility and to inform the home server where it is now by registering with home server through outbound proxy server for unicast mobility. Figure 3-4 shows the registration procedure and Figure 3-5 is the sequence diagram for registration.

In Figure 3-4, MH1 moves to FN from its home. As soon as MH reaches the FN, it sends REGISTER_J message to local proxy server (proxy2). After finishing the registration, proxy2 sends this REGISTER_J message to RS2. When RS2 receives the REGISTER_J message from MH1 through its proxy server proxy2, it first checks if the domain name shown in the "leave" field is the same as the RS2's domain name. This REGISTER_J message could be from a mobile host that has moved from a different domain or from the same domain. If the REGISTER_J message is for multicast and the domain name in the Leave field is different from its domain, RS2 adds MH1 in its domain group member's table that contains the entries as [proxy2, MH1]. It then creates a REGISTER_L message and transfers this REGISTER_L message to the pre-root server

RS1 of the domain shown in the "Leave" field of REGISTER_J message. RS1 deletes its [proxy1, MH1] entry from its domain group member's table and transfers this REGISTER_L message to the pre-proxy server (proxy1) mentioned in the "Leave" field of REGISTER_L message. Proxy1 deletes MH1 from its list of mobile hosts supported. Similarly, proxy2 adds MH1 in its list of mobile hosts supported. If mobility happens in the same domain such as moving from proxy2 to proxy3, the REGISTER_J message is for multicast and the domain name in the "Leave" field is the same as RS2's domain, RS2 updates its entry in its domain group member's table from [proxy2, MH1] to [proxy3, MH1]. Proxy3 adds MH1 in its list of mobile hosts supported and proxy2 deletes MH1 from its list of mobile hosts supported. Here, REGISTER_L and REGISTER_J are the same. We gave the different names like REGISTER_J for joining and REGISTER_L for leaving just for readers to understand it easily. Every server will send back 200OK message for the successful registration.

The detailed REGISTER message is as shown in the following:

- **Registering with foreign proxy server and RS**

MH1 sends REGISTER_J message to proxy2 for registering and joining the multicast group.

REGISTER_J MH1→ proxy 2

REGISTER_J sip: registrar.foreign.com SIP/2.0

Via: SIP/2.0/UDP laptop.foreign.com

From: sip: MH1@home.com

To: sip: MH1@home.com

Contact: sip: MH1@10.12.3.6

Join: sip: 224.10.20.5 (multicast group G)

Leave: sip: pre-proxy@pre-foreign.com

<other sip header>

…

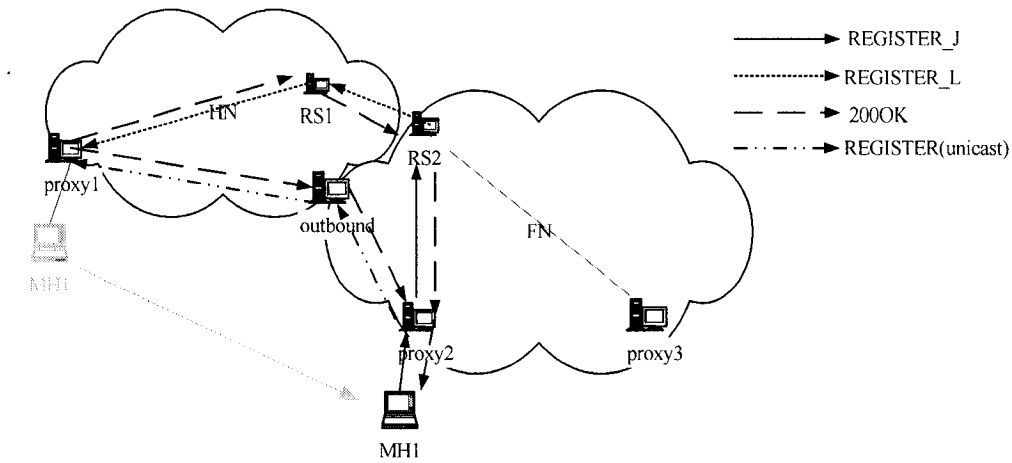

Figure 3-4 Registration of Mobile Host Moved From a Different Domain

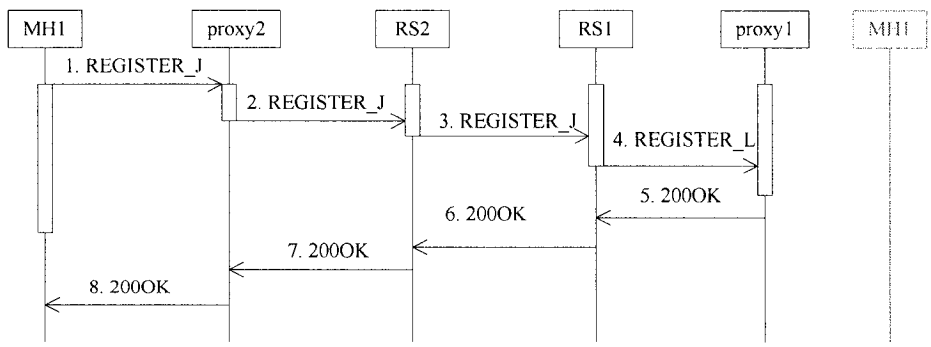

Figure 3-5 Sequence Diagram for Registration

Proxy2 processes the registration request from MH1 and finds that MH1 needs to join the multicast group G. Proxy2 transfers this REGISTER_J message to RS2 for MH1 to join the group G and release the path to former IP address of MH1 from pre-proxy server.

REGISTER_J proxy2→ RS2

REGISTER_J sip: rootserver.foreign.com SIP/2.0

Via: SIP/2.0/UDP proxy2.foreign.com

Via: SIP/2.0/UDP laptop.foreign.com

From: sip: MH1@home.com

To: sip: MH1@home.com

Contact: sip: MH1@10.12.3.6

Join: sip: 224.10.20.5 (multicast group G)

Leave: sip: pre-proxy@pre-foreign.com

<other sip header>

...

RS2 finds the pre-RS (RS1) for MH1 by checking its RSs routing table according to the information it gets from "leave" field of REGISTER_J. After RS2 registers for MH1, it creates the REGISTER_L message and then forwards it to RS1.

REGISTER_L RS2→ RS1

REGISTER_L sip: rootserver.pre-foreign.com SIP/2.0

Via: SIP/2.0/UDP RS2.foreign.com

Via: SIP/2.0/UDP proxy2.foreign.com

Via: SIP/2.0/UDP laptop.foreign.com

From: sip: MH1@home.com

To: sip: MH1@home.com

Contact: sip: MH1@10.12.3.6

Join: sip: 224.10.20.5 (multicast group G)

Leave: sip: pre-proxy@pre-foreign.com

<other sip header>

...

RS1 updates its routing table for multicast group G, and forwards this message to pre-proxy proxy1.

REGISTER_L RS1→ proxy1

REGISTER_L sip: proxy1.pre-foreign.com SIP/2.0

Via: SIP/2.0/UDP RS1.pre-foreign.com

Via: SIP/2.0/UDP RS2.foreign.com

Via: SIP/2.0/UDP proxy2.foreign.com

Via: SIP/2.0/UDP laptop.foreign.com

From: sip: MH1@home.com

To: sip: MH1@home.com

Contact: sip: MH1@10.12.3.6

Join: sip: 224.10.20.5 (multicast group G)

Leave: sip: pre-proxy@pre-foreign.com

<other sip header>

...

After updating the information about MH1 for the multicast group G, proxy1 sends back 200OK message for successful REGISTER_L to RS1. 200OK message reaches the same servers as mentioned above through the same path until MH1.

- **Registering with home server**

For multicast mobility, no matter where MH moves and no matter where it joins the multicast group, it can always obtain the packets sent to the multicast group by other CHs. That is why MH need not register with its home server after registering with foreign proxy server and RS for multicast communication. On the other hand, for unicast mobility each time MH moves to a new network or say obtains a new IP address, it should register with its home server to inform home server where it is now. After MH1 registers with local proxy server, it sends a REGISTER message to its home server through local proxy as well as outbound proxy. If MH does not register with home server for unicast communication, CH may not find it and does not know where it should send messages. It will result in losing connection between CHs and MH.

REGISTER MH1→ proxy2

REGISTER sip: registrar.home.com SIP/2.0

Via: SIP/2.0/UDP laptop.foreign.com

From: sip: MH1@home.com

To: sip: MH1@home.com

Contact: sip: MH1@10.12.3.6

<other sip header>

...

Proxy2 receives this REGISTER message and checks its database or location server to know where the outbound proxy server is. In order to reach MH1's home server, then proxy2 forwards REGISTER message to this outbound proxy server.

REGISTER proxy2→ outbound proxy

REGISTER sip: registrar.home.com SIP/2.0

Via: SIP/2.0/UDP proxy2.foreign.com

Via: SIP/2.0/UDP laptop.foreign.com

From: sip: MH1@home.com

To: sip: MH1@home.com

Contact: sip: MH1@10.12.3.6

<other sip header>

...

According to the Hierarchical Registration, outbound proxy server keeps the information about MH1 in its database and routes REGISTER message to MH1's home server across other proxy server when it is the first time for outbound proxy server to route REGISTER message to MH1's home server from MH1. When it is not the first time, outbound proxy server should check if MH1 locates in the same domain as before by obtaining information from its database. If yes, outbound proxy server updates the information related to MH1 in its database and need not route REGISTER message to MH1's home server. If not, it follows the same steps as if it is first time.

REGISTER outbound proxy → home server

REGISTER sip: registrar.home.com SIP/2.0

Via: SIP/2.0/UDP outbound_proxy.foreign.com

Via: SIP/2.0/UDP proxy1.foreign.com

Via: SIP/2.0/UDP laptop.foreign.com

From: sip: MH1@home.com

To: sip: MH1@home.com

Contact: sip: MH1@10.12.3.6

<other sip header>

...

Home server would send back 200OK message to MH1 through the entities mentioned above if it successfully accepts this message. If the last entity receiving 200OK message is outbound server, it would be the source of sending the 200OK message. Here these 200OK messages are omitted. Readers who are interested in it can refer to the reference [27].

3.3 SESSION SETUP PROCESS

When a correspondent host (CH) wants to set up a media session with a multicast group G, it should have itself registered with RS to join the multicast group and be part of the RS multicast tree. It sends INVITE message to its local proxy server (proxy1) and informs it to set up a media session to the group G (Figure 3-6). Local proxy server checks its database or location server to obtain the information about RS of group G. It then directs this INVITE message to RS1. If RS1 has local group members registered with it, it forwards INVITE message to all local group members. RS1 also sends INVITE

message to other RSs in other domain that have group G members. RS1 routes INVITE message to RS2 that has group members including MH1 and MH2. After receiving INVITE message from RS1, RS2 multicasts INVITE message to group members MH1 and MH2. MH1 and/or MH2 will send back 200OK message to CH if it / they accept the INVITE. After CH receives 200OK, it responds with ACK message, and the session is set up. RTP/RTCP media can now be sent and session is established. Figure 3-7 is a sequence diagram that gives the detailed steps for session set-up.

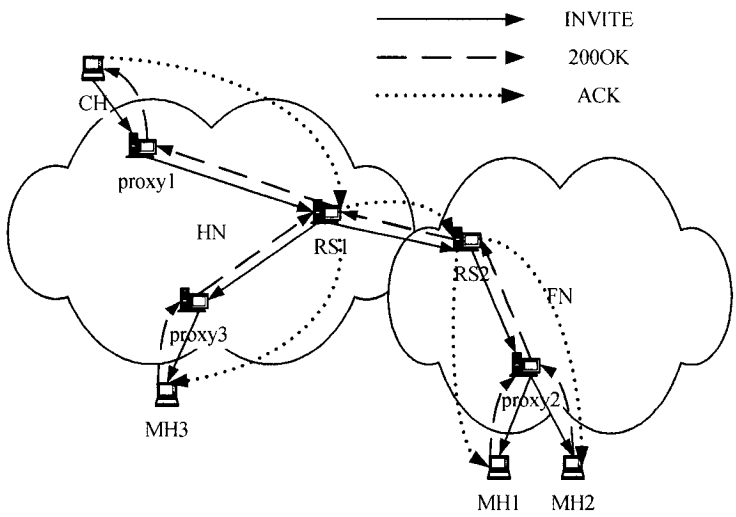

Figure 3-6 Example of Session Setup

The detailed message is following:

INVITE CH → proxy1

INVITE groupmembers@224.10.20.5 (multicast address) SIP/2.0

Via: SIP/2.0/UDP laptop.home.com

From: CH@home.com

To: groupmembers@224.10.20.5

<other SIP header>

...

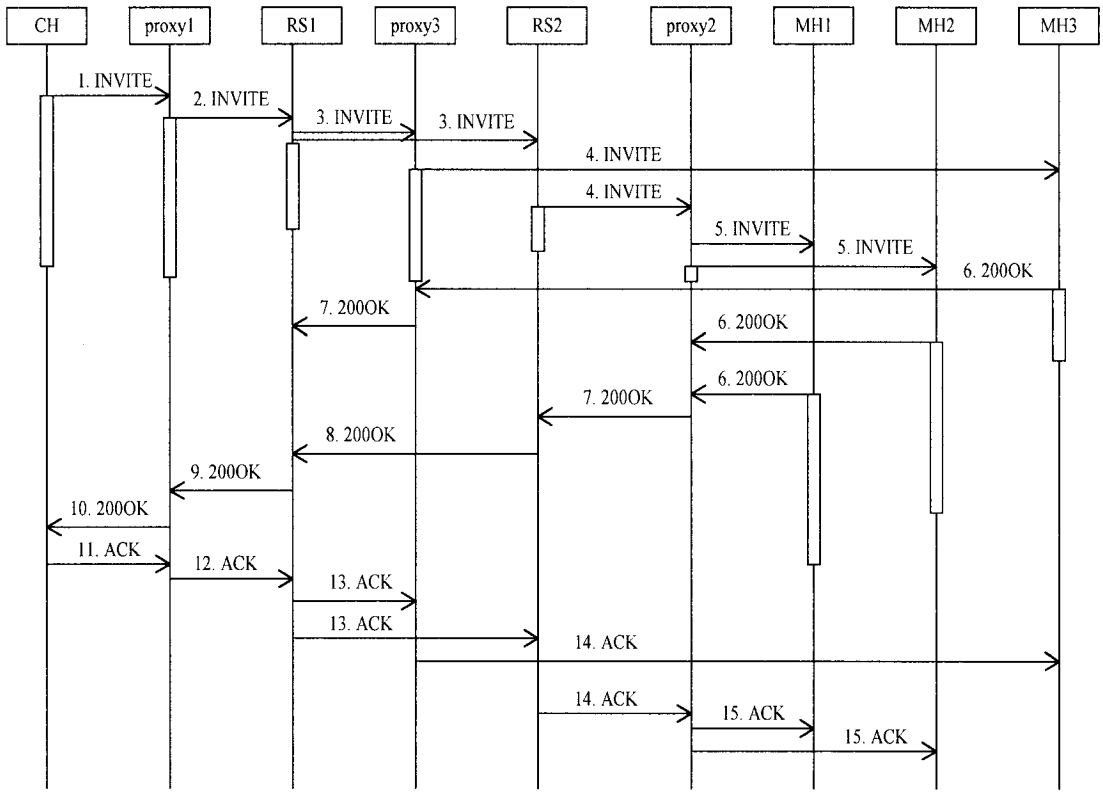

Figure 3-7 Sequence Diagram for Session Set-up (RS1 having local members)

INVITE proxy1 → RS1

INVITE groupmembers@224.10.20.5 (multicast address) SIP/2.0

Via: SIP/2.0/UDP proxy1@home.com

Via: SIP/2.0/UDP laptop.home.com

From: CH@home.com

To: groupmembers@224.10.20.5

<other SIP header>

...

INVITE RS1 → RS2 and RS1 → proxy3

INVITE groupmembers@224.10.20.5 (multicast address) SIP/2.0

Via: SIP/2.0/UDP RS1@home.com

Via: SIP/2.0/UDP proxy1@home.com

Via: SIP/2.0/UDP laptop.home.com

From: CH@home.com

To: groupmembers@224.10.20.5

<other SIP header>

...

INVITE proxy3 → MH1

INVITE groupmembers@224.10.20.5 (multicast address) SIP/2.0

Via: SIP/2.0/UDP poxy3@home.com

Via: SIP/2.0/UDP RS1@home.com

Via: SIP/2.0/UDP proxy1@home.com

Via: SIP/2.0/UDP laptop.home.com

From: CH@home.com

To: groupmembers@224.10.20.5

<other SIP header>...

As soon as RS2 gets the INVITE message, it multicasts INVITE to all local proxy servers that have the same group members. The message is similar to the above and is not repeated again. For 200OK and ACK message, the difference is the first line that is changed from INVITE groupmembers@224.10.20.5 (multicast address) SIP/2.0 to SIP/2.0 200OK and SIP/2.0 ACK. 200OK message comes from MHs and follows the INVITE path in the reverse direction, and ACK message come from CH and follows the same path direction of INVITE. Here one 200OK and ACK message is shown.

200OK MH1 → proxy3

SIP/2.0 200OK

Via: SIP/2.0/UDP RS1@home.com

Via: SIP/2.0/UDP proxy1@home.com

Via: SIP/2.0/UDP laptop.home.com

From: CH@home.com

To: groupmembers@224.10.20.5

<other SIP header>

...

ACK CH → proxy1

SIP/2.0 ACK

Via: SIP/2.0/UDP laptop.home.com

From: CH@home.com

To: groupmembers@224.10.20.5

<other SIP header>

...

3.4 SESSION TERMINATION

When CH wants to terminate the session between CH and group G, it sends BYE message to root server RS1 through the local proxy server proxy1. Then RS transfers the BYE message to its local group members as well as to RS2 in other domain that has the same group members. On the other hand, if any member (MH) does not want to continue this communication, it sends BYE message to its local proxy server, then the server will terminate this session to it. If no group members belong to this local server, this server sends BYE message to RS in the same domain and the RS will terminate the session to it. Figure 3-8 is the sequence diagram of session termination corresponding to the Figure 3-6 when CH initiates to terminate the communication.

The detailed message is following:

BYE CH → proxy1

BYE sip: groupmembers@224.10.20.5 (multicast address) SIP/2.0

Route: <proxy1.home.com>

Route: <RS1.home.com>

<other SIP header>

...

BYE proxy1 → RS1

BYE sip: groupmembers@224.10.20.5 (multicast address) SIP/2.0

Route: <RS1.home.com>

<other SIP header>

...

RS1 receives BYE message and removes the first Route header field. It reformats the request to be:

BYE RS1 → proxy3

BYE sip: groupmembers@224.10.20.5 (multicast address) SIP/2.0

Route: <proxy3.home.com>

<other SIP header>

...

BYE RS1→ RS2

BYE sip: groupmembers@224.10.20.5 (multicast address) SIP/2.0

Route: <RS2.foreign.com>

<other SIP header>

...

Proxy3 multicasts BYE request to its local group members and forwards 200OK message coming from group members to RS1. RS2 multicasts BYE request to its local group members through its local proxy servers. RS2 also forwards 200OK message that are from local proxy servers. The session is terminated finally after CH gets 200OK message through proxy1 and RS1.

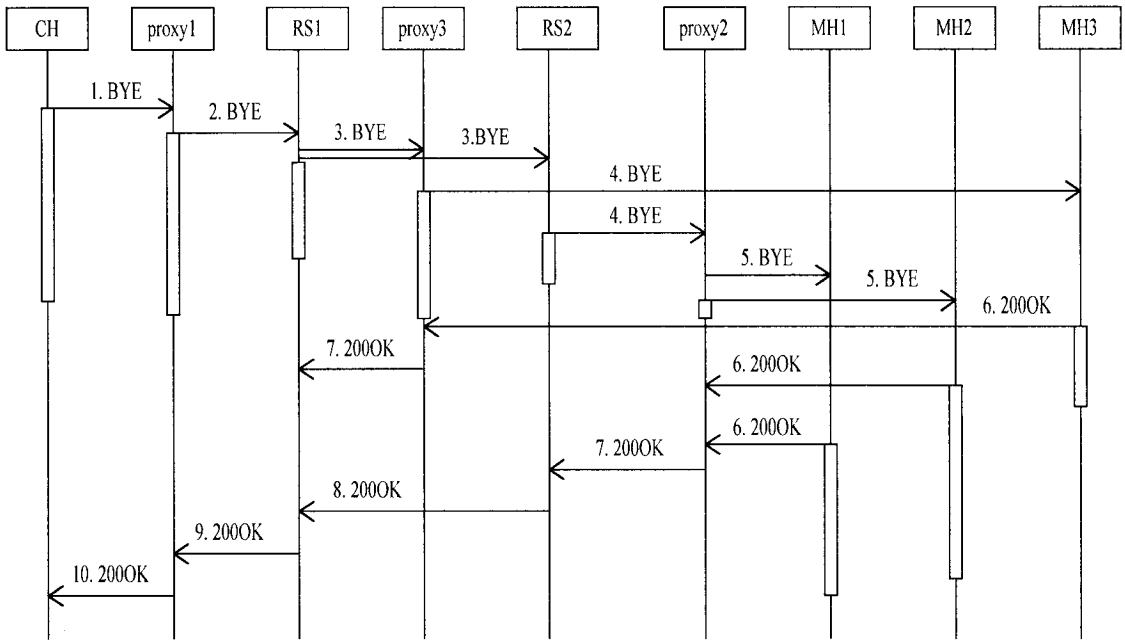

Figure 3-8 Sequence Diagram for Session Termination

3.5 SUMMARY

In this chapter we propose a new architecture to support multicast mobility in SIP based networks based on the work introduced in Chapter 2. We define a new server, RS, to control the multicast session messages. In order to support the multicast functionality, the REGISTER message field was added with new fields: "From" and "To". The detailed message format for REGISTER messages are shown for MHs to join and leave the multicast sessions with mobility. Finally we apply the UML diagram to explain how the new proposal works during registration, session setup and session termination. The following chapter will focus on the handoff issue for SIP mobility with multicast.

CHAPTER 4

SIP-BASED MULTICAST MOBILITY HANDOFF

In this chapter, we introduce two kinds of handoff: subnet handoff and domain handoff [33]. Upon every cell handoff, the DHCP client in the mobile station (MS) initiates a reconfiguration process. Its DHCP client requests for new IP address and domain name, etc. The MS examines the DHCP response according to the following rules to identify the handoff process [33].

- If the new IP address is the same as the current one, then a cell handoff has occurred
- If the new IP address is different from the current one though the domain name is the same as the current one, the subnet handoff happens.
- If the new IP address and new domain name both differ from current ones, it is a domain handoff process.

4.1 SUBNET HANDOFF

For multicast mobility, the basic mobility idea is the same as unicast mobility in Section 2.2. We also discuss MH mobility subnet handoff in two different stages: pre-call moving and mid-call moving.

4.1.1 Pre-call Moving

In Figure 4-1 and Figure 4-2, a MH moves and gets a new IP address. It moves in the same domain as the current domain. It registers with local proxy2 (1). Proxy2 transfers REGISTER_J to RS1 for MH to join the multicast group (2). RS1 updates the information related to MH and forwards REGISTER_L to proxy1 for leaving group G from proxy1 (3). MH will hit the same outbound proxy when registering with home server (proxy_H) (4). The outbound server will find MH is still in the same domain, so it will update all information related to the MH, but not forward the request to the home server.

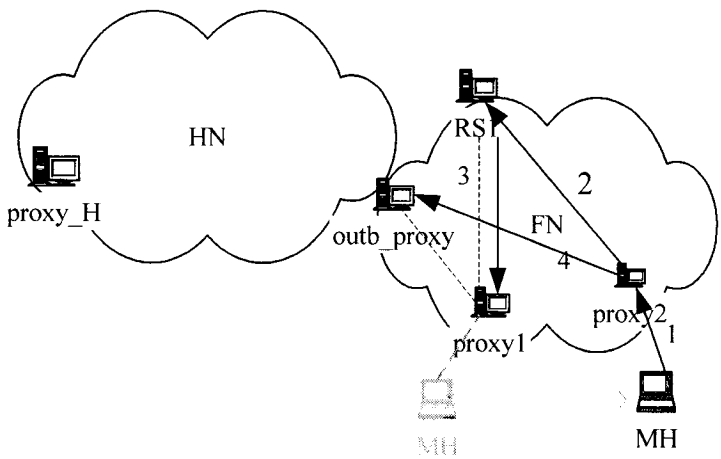

Figure 4-1 Pre-call Moving

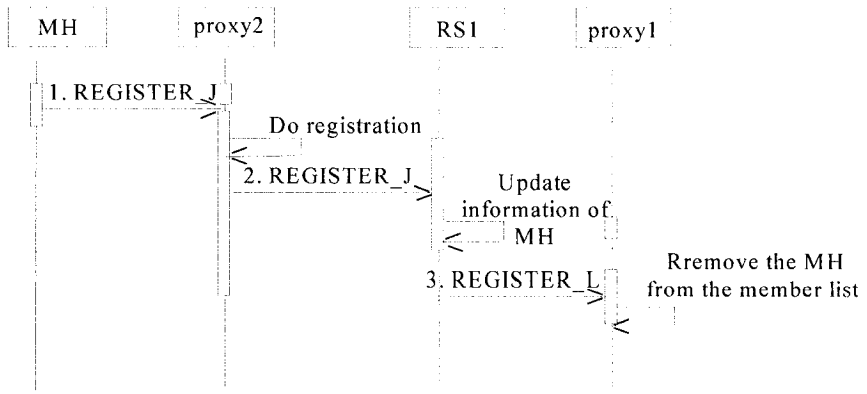

Figure 4-2 Sequence Diagram for Pre-call Moving

4.1.2 Mid-Call Moving

In Figure 4-3, MH1 moves as it is getting signal/data packets from the multicast group. We use the lighter line to represent the multicast communication after the multicast session is setup. MH1 registers with new local server (proxy2) and informs the local server to join the multicast group G (1). The new local server does registration for MH1 and forwards REGISTER_J to RS2 (2). RS2 adds MH1 to its local multicast tree and forwards REGISTER_L to the pre-proxy server (proxy1) for informing it to release the path to MH1 (3). MH1 registers with home server by sending REGISTER message to the outbound server (4). If the new proxy server proxy2 already has group members before MH1 registered with it, proxy2 would setup session with MH1 directly. If there no members exist, RS2 would setup session with MH1 through proxy2. After receiving the response to REGISTER_L, it would terminate session with proxy1 if RS2 knows that there is no group member in proxy1. The sequence diagram for Figure 4-3 is similar to Figure 4-2.

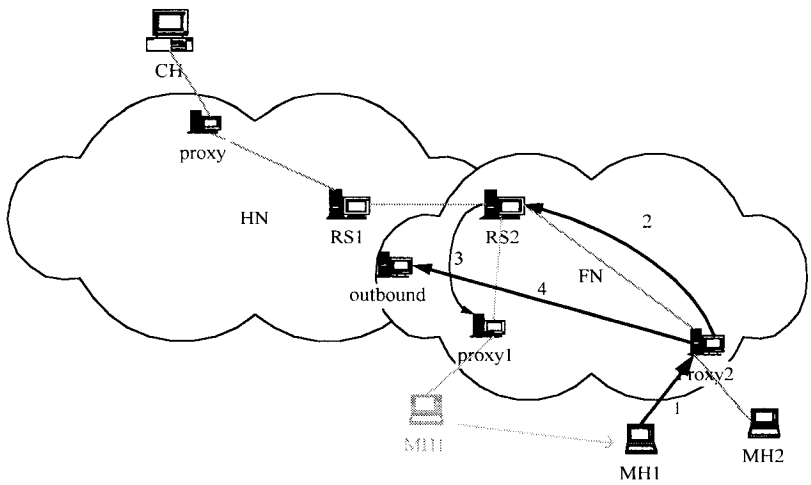

Figure 4-3 Mid-call Moving

4.2 DOMAIN HANDOFF

For domain handoff, we discuss two different situations in which there still are pre-call and mid-call cases. It is the same idea as subnet pre-call and mid-call moving. So in the following we will not discuss pre-call and mid-call again and only focus on if there are members in the new domain.

4.2.1 Destination Foreign Domain with Group Members

Figure 4-4 and Figure 4-5 show handoff when a MH moves to a domain with already existing group members. When MH2 moves to a domain where root server RS2 has been activated and has the same group members in this domain, it sends

REGISTER_J message to local proxy server (proxy2) with "join" and "leave" fields (1). Proxy2 does registration for MH2 after receiving the message. Then proxy2 sends REGISTER_J with "join" and "leave" field to RS2 in this domain (2). RS2 adds MH2 in its multicast tree, then forwards REGISTER_L message to the pre-root server RS3 (3). RS3 updates the information related to MH2 and transfers the message to pre-proxy server proxy3 (4). Proxy3 removes MH2 from its members list. After that, MH2 sends unicast REGISTER message to its home server through the outbound proxy server. Outbound proxy server puts MH2 in its contact list. Outbound server transfers the REGISTER message to the MH2's home server. The home server updates MH2's information. Then it will send back 200OK message to MH2 following the same path as REGISTER message comes. The outbound proxy and unicast registration messages are however omitted from the diagram for clarity purposes.

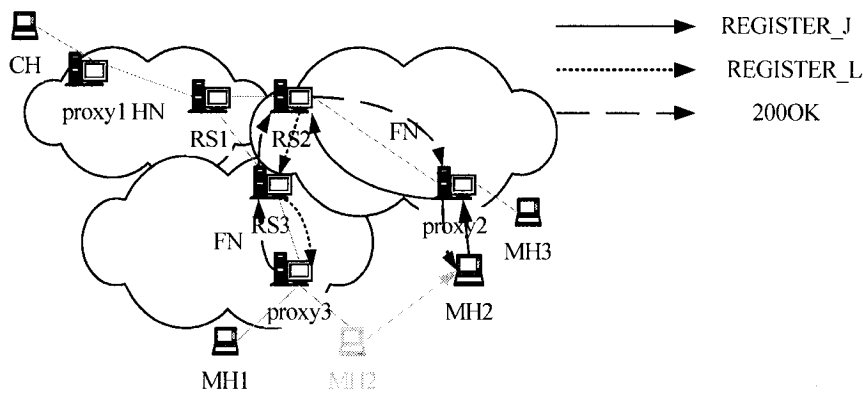

Figure 4-4 Domain Handoff with group members

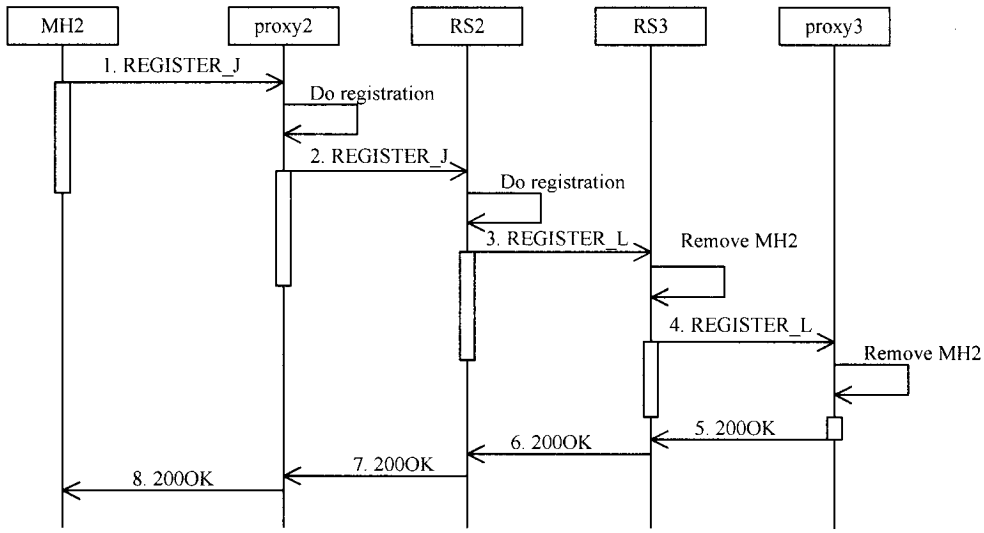

Figure 4-5 Sequence Diagram of Domain Handoff with group members

4.2.2 Destination Foreign Domain without Group Members

In Figure 4-6, when MH2 moves to a domain where root server has not been activated and does not have the same group members in this domain, sending RS_REGISTER_J message to AS for joining the RS multicast group G will first activate the RS of this domain. After successfully joining the RS multicast group, RS adds the MH to its local multicast tree and sends REGISTER_L request to the pre-root server of MH to inform it that MH has already moved to a new domain and to release the path to MH in order not to send message to the end system where no member exists. Figure 4-6 is the structure about MH moving to a domain without the group members. The following is the sequence diagram in Figure 4-7. MH2 registers with local proxy server (proxy2) with "join" and "leave" fields (1). Proxy2 does registration for MH2. Proxy2 sends

REGISTER_J with "join" and "leave" to the root server (RS2) in this domain (2). RS2 sends join request RS_REGISTER_J to its multicast tree AS (Administrator Server) (3). Then it adds the MH2 in its delivery tree. RS2 forwards REGISTER message as REGISTER_L to the pre-root-server RS3 (4). RS3 removes MH2 from its multicast group. RS3 transfers REGISTER_L to the local proxy server (proxy3) and informs it to release the path to MH2 (5). Proxy3 removes MH2 from its member list and then sends back 200OK message. For unicast registration, it is same as the unicast registration in Section 3.2 of Chapter 3.

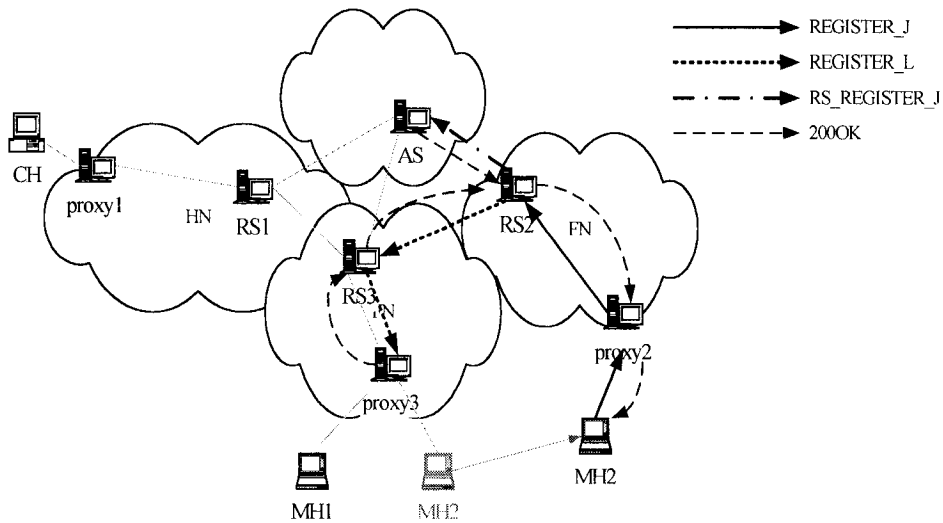

Figure 4-6 Domain Handoff without group members

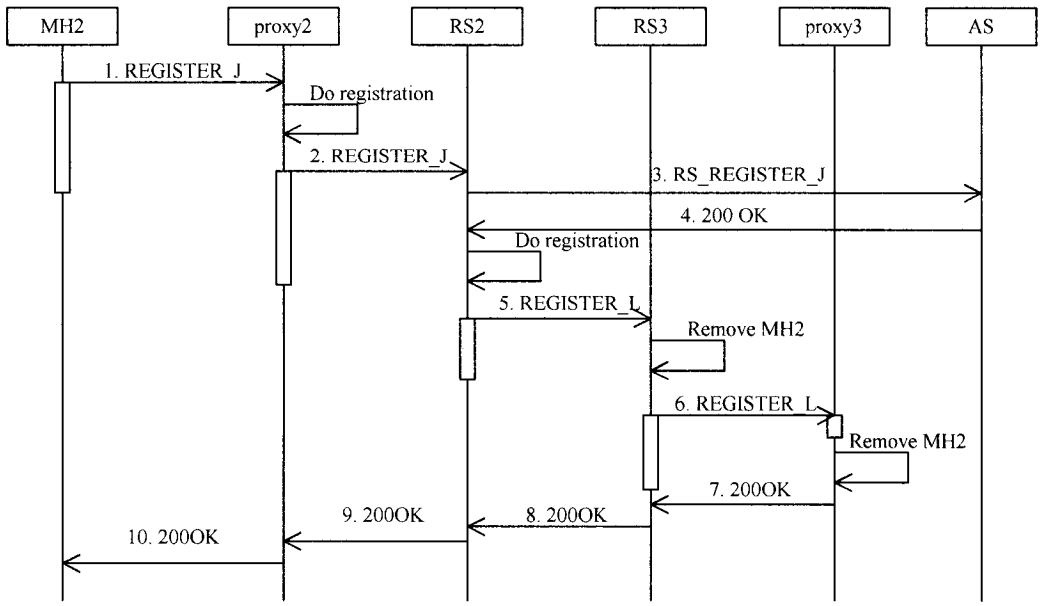

Figure 4-7 Sequence Diagram for the Domain Handoff without the Group Members

4.3 SUMMARY

Dealing with handoff is the most difficult problem in SIP mobility. In this chapter we introduce the working of the new proposal when handoff takes place. We discuss it according to the different situations:

1. When the MH moves within the same domain (subnet handoff):
    - Pre-call moving.
    - Mid-call moving.
2. When the MH moves to a different domain (domain handoff):
    - The new domain with the same group member/members.

- The new domain without the same group member/members.

We also employed the UML diagram to express the theory of how it works when handoff happens.

CHAPTER 5

SIMULATIONS

We have used ObjectGeode [34] as a simulation tool on the system model built using SDL (ITU Specification and Description Language) [35] [36]. ObjectGeode is a formal toolset dedicated to analysis, design, and verify through simulation and testing of real time and distributed applications. ObjectGeode can verify that the system works as expected with limited number of nominal cases described by MSC (Message Sequence Charts) [37].

SDL is an internationally standardized language, developed and maintained by ITU-T (International Telecommunication Union-Telecommunication) since 1976. The language is well suited for describing stimuli-response behavior since it is based on experience of describing telecommunication systems functions as communicating state machines. It has evolved from an informal drawing technique to a formal description technique. Due to its simple conceptual basis and its choice of graphical or textual representation it is very intuitive and helps the designer in visualizing relationships. It is well received in the telecommunication community and has been extensively used both in standard and product development. MSC (Message Sequence Charts) is a language that has been informally used in ITU and in industry for visualization of selected message trace within communication system. The MSC language is now standardized by ITU-T in recommendation Z.120, and is used most frequently together with SDL. MSCs and SDL

descriptions should be regarded as two different but complementary system representations. SDL on one hand provides a complete behavior description of individual processes (communicating entities), but there is no direct description of communication between several entities. On the other hand, MSC provides a clear description of system traces in the form of message flow diagrams.

SDL and MSC are just two of many languages. They combine the power of expression and communication of a graphical presentation with the formality required to carry out formal analysis. MSC enables the formalization of scenario use sequence that informally describes observed behavior in stimulus-response terms. MSC also allows the interaction between the entities in a system to be shown. SDL bridges the gap between specification and implementation, as it supports modeling at an abstract level as well as detailed description of the implementation. Compared with MSC, an SDL description describes the entities in the system as a number of well-defined machines that determine the response for each stimulus. SDL also allows the relationships between entities and classes in the system to be shown. Together the two languages provide a complete solution for specification and design, and integrate well with ASN.1 for protocols. The combination fits the requirements of specification languages: unambiguous, clear, precise, easy to communicate, learn and use; supports analysis, modeling and product development; allows abstraction.

5.1 STRUCTURAL DEFINITION

A system specification in SDL is divided into two parts: the system and its environment. A system specification is a formal model that defines the relevant

properties of an existing system in the real word. Everything outside the system belongs to the environment. The SDL specification defines how the system reacts to events in the environment that are communicated by messages, called signals, sent to the system. The behavior of a system is the joint behavior of all the process instances contained in the system, which communicate with each other and the environment via signals. The process instances exist in parallel with equal rights. A process may create other processes, but once created, these new process instances have equal rights. SDL process instances are extended finite-state machines (FSM). An extended FSM is based on an FSM with the addition of variables that represent additional state information and signal parameters. The union of the FSM-state and additional state variables represent the complete state space of the process [38]. The relationship between modeled entities, their interfaces, and attributes are considered parts of the structural definition. A SDL system represents static interactions between SIP entities. The channel connected between various block instances specify the signals or SIP message that are sent between user agent and/or proxy/RS servers. Block and process types are used to represent SIP entity types such as user agents, proxies and root servers. SIP messages are defined as SDL signals in the SIP message package. We have defined only the main header fields of the SIP header because we are interested in only the operation and the endpoints of the session. These fields in a SIP message are represented by the corresponding signal parameters.

In our SDL model, we use different SDL systems to represent different structural bindings between SIP entities and to simulate a particular set of call scenarios. The first system in our model realizes the concept of registration after moving. The most complex

system in our models realizes the concept of originating and terminating the session. The originating block contains the entire user agent process instances that originate SIP requests while the terminating block contains all user agent process instances that receive these requests. Upon receiving a request, a terminating user agent would reply with the corresponding response messages.

5.2 SYSTEM MODEL

We design and define the internal structure and behavior of three main components by SDL/MSC: SIP user agent (UA), SIP proxy server (PS) and SIP root server (RS). In our system modeling, we provide two configurations for the registration, session setup and session termination scenarios. One configuration is for registration as shown in Figure 5-1 that includes user agent (UA), new proxy server (PS2), pre-proxy server (PS1), new RS (RS1) and pre-RS (RS2). The other configuration as shown in Figure 5-2 is session setup and session termination that contains one caller (UA1), three PS (PS1, PS2, PS3), three RS (RS1, RS2, RS3) and four cal lees (UA2, UA3, UA4, UA5) distributing the three different domains. Figure 5-3 and Figure 5-4 show the systems models described by SDL for registration and session setup respectively. Session termination system description (Figure 5-5 and Figure 5-6) is similar to session setup except for the different SIP messages used.

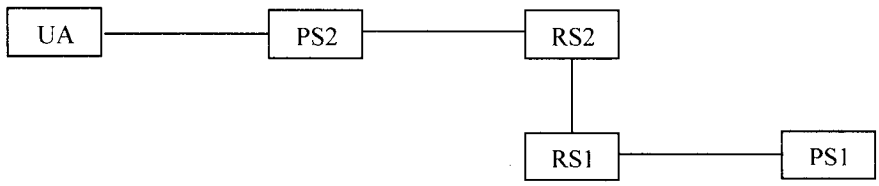

Figure 5-1 Configuration for Registration

Fig 5-1 shows the configuration of registration for general situation. Every time UA moves to a new area, it is supposed to register with new RS (RS2) through the new local proxy server (PS2). The new RS is responsible to tell its pre-RS (RS1) and pre-proxy server (PS1) to release the path to its old location.

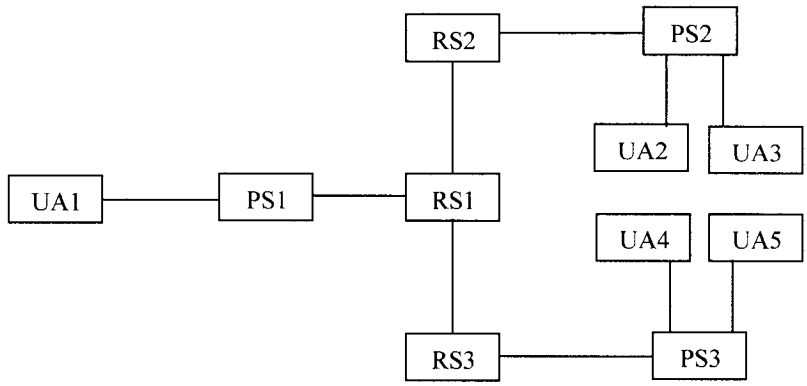

Figure 5-2 Configuration for Session Setup and Termination

Figure 5-2 shows the configuration of session setup and session termination. In the case of session setup, UA1 sends INVITE message to multicast address to setup the session with multicast group. They response it with 200OK messages as soon as the invitees accept this message. Then UA1 sends ACK message to all invitees that response with 200OK. The configuration for session termination is same as shown in Figure 5-2. UA terminates its session by sending BYE message.

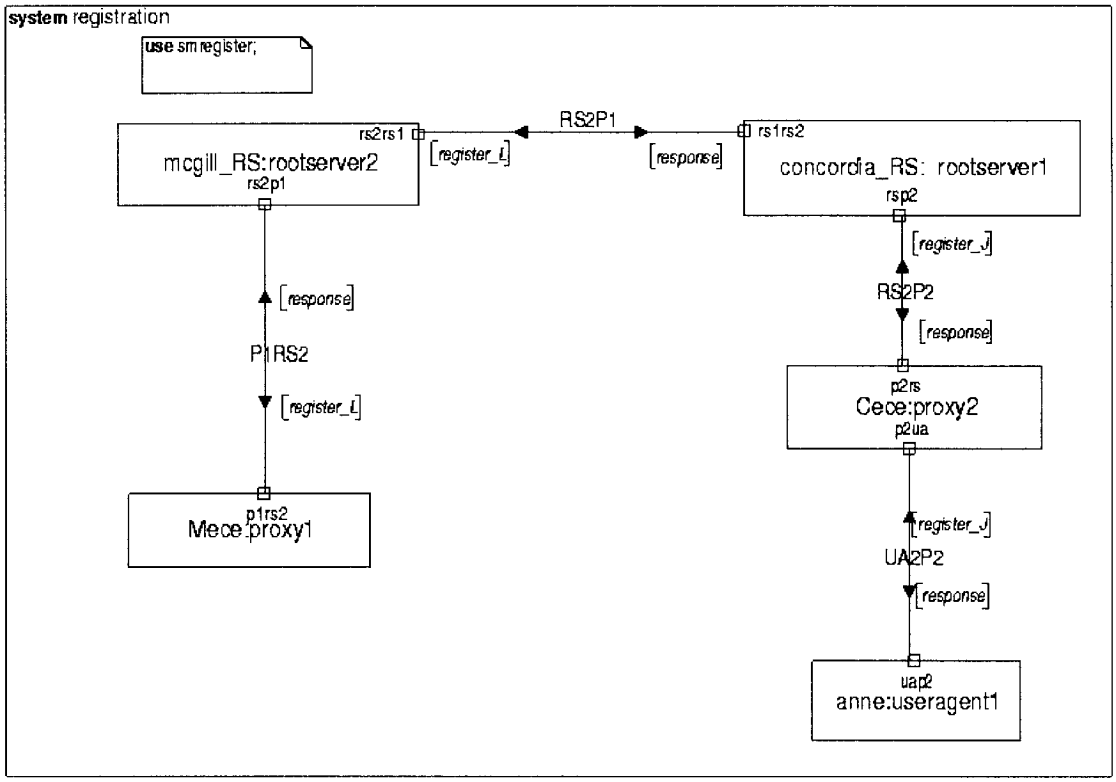

Figure 5-3 SDL System Structure for Registration

Figure 5-3 includes five blocks that are proxy1 ("Mece"), rootserver2 ("mcgill_RS"), rootserver1 ("concordia_RS") and proxy2 ("Cece") and useragent1 ("anne"). Here "anne" moves from "mcgill" domain to "concordia" domain. She registers by sending "register_J" to its new RS ("concordia_RS") through its new local proxy server ("Cece"). "concordia_RS" sends "register_L" message to its previous RS and proxy server ("mcgill_RS" and "Mece") for release the path to the old location of "anne" after it finishes the registration for "anne".

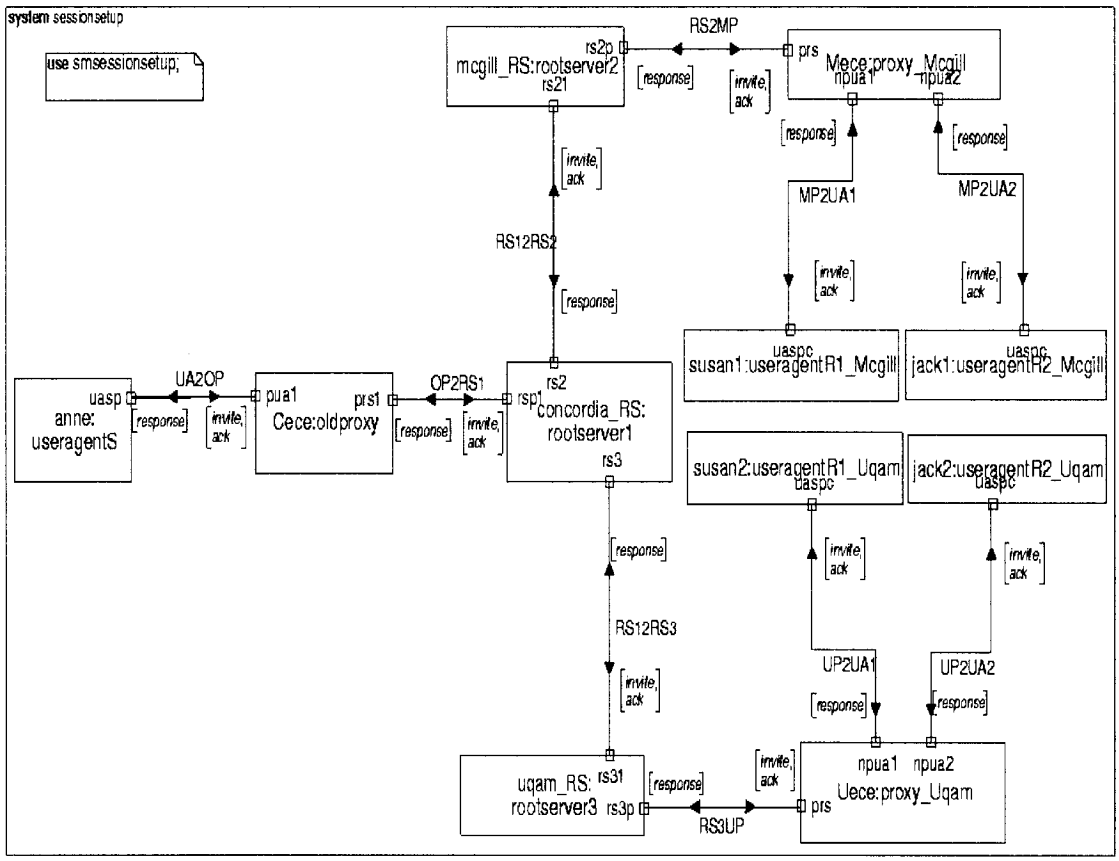

Figure 5-4 SDL System Structure for Session Setup

Figure 5-4 shows the system model for session setup. There are eleven blocks in this system. They include five user agents ("anne", "susan1", "jack1", "susan2" and "jack2"), three proxy servers ("Cece", "Uece" and "Mece") and three root servers ("uqam_RS", "concordia_RS" and "mcgill_RS"). When the caller wants to setup session to multicast group, it sends INVITE message to the group through its local proxy server and RS.

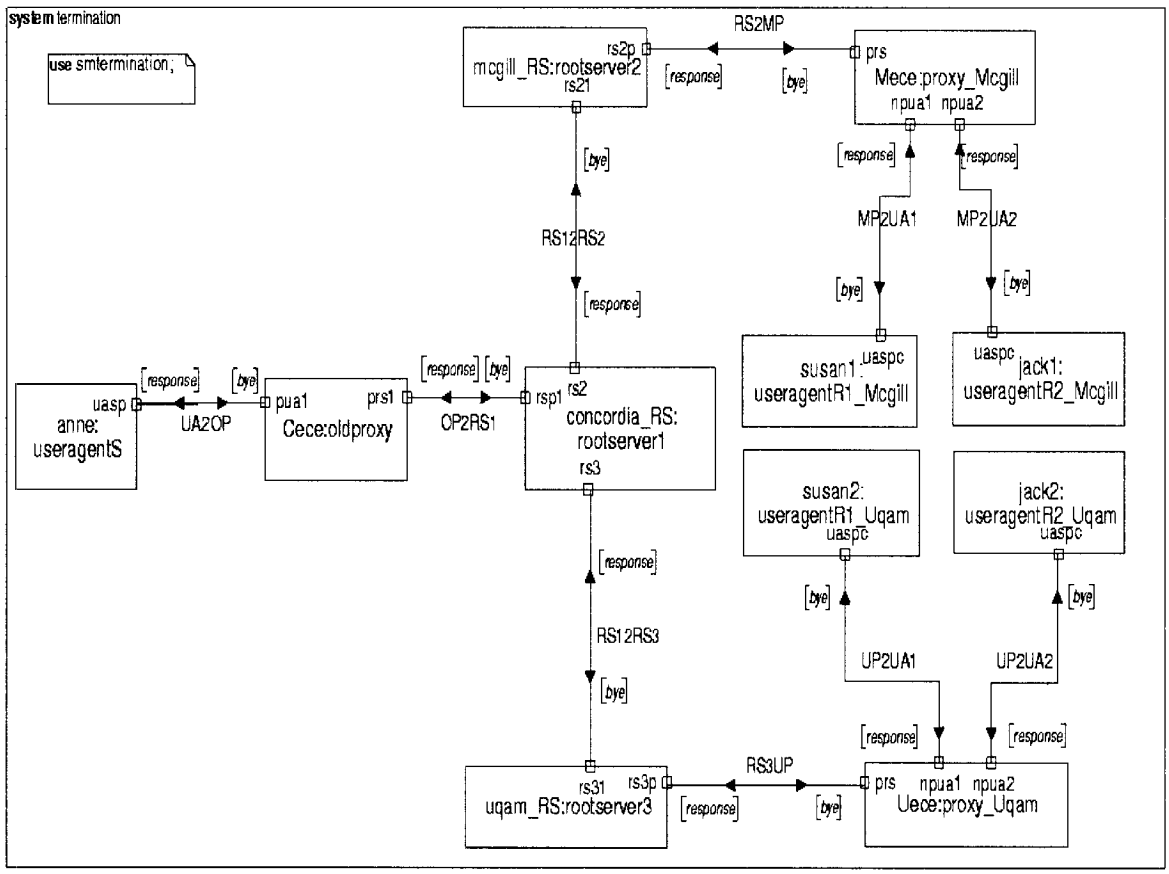

Figure 5-5 SDL System Structure for Session Termination (caller to callees)

The termination follows session setup. The structure of session termination is the same as the session setup. Figure 5-5 represents the session termination in the case that the request is sent from caller to all callees. The caller wants to quit the session and initiates the termination by sending BYE message to all callees.

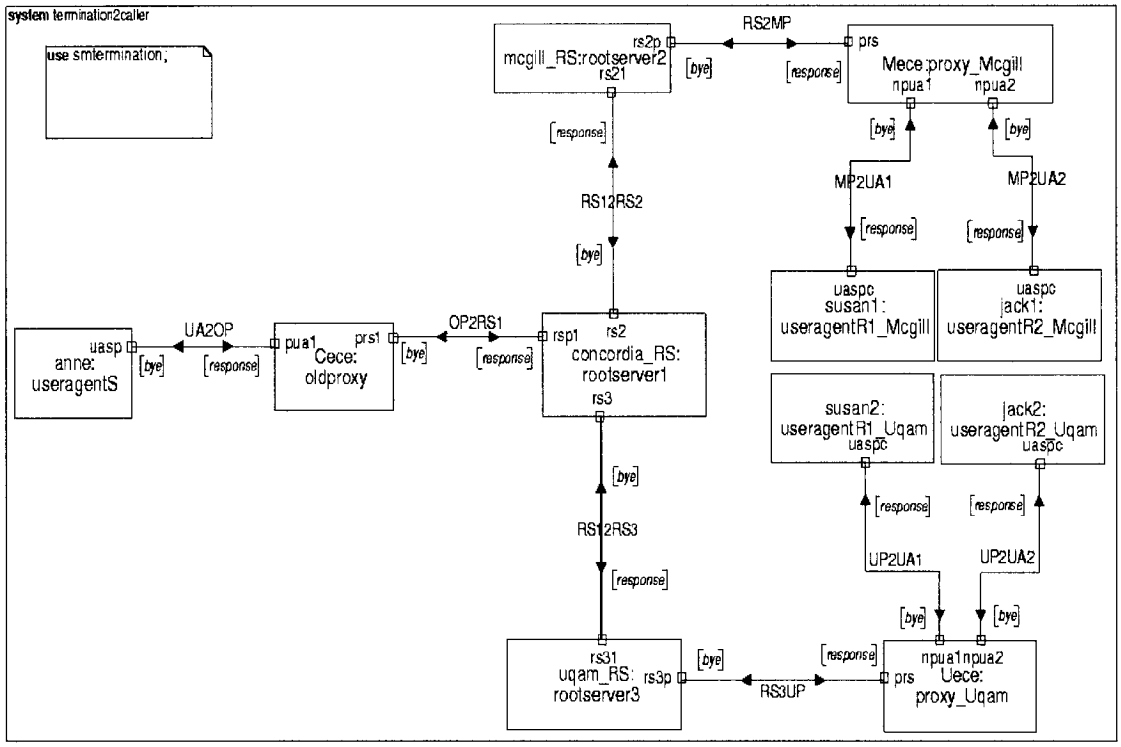

Figure 5-6 SDL System Structure for Session Termination (Callee to Caller)

Figure 5-6 represents the termination situation that the request is sent from one or some callee/callees to the caller direction. This figure 5-6 is the same as Figure 5-5. The only difference is the message flow direction.

5.3 SIMULATIONS

Using ObjectGeode tool set we have built our SDL model of SIP Multicast Mobility to simulate whether the model would be able to realize specific interaction scenarios described informally in Unified Modeling Language (UML) [39] in Chapter 3, and to simulate if our SDL system was able to generate the same MSC as the expected

UML. We separate the simulation into three parts: 1) registration, 2) session setup, and 3) session termination. Each part simulates different scenarios. In each part, the simulation results and SIP package are given as following.

5.3.1 Registration

The registration package is shown in Figure 5-7. All data and data types used by signals are defined here. All the signals used in registration simulation such as "register_J", "register_L" and "response" are included in this package. Here "sm" in "smregistration" means "SIP message".

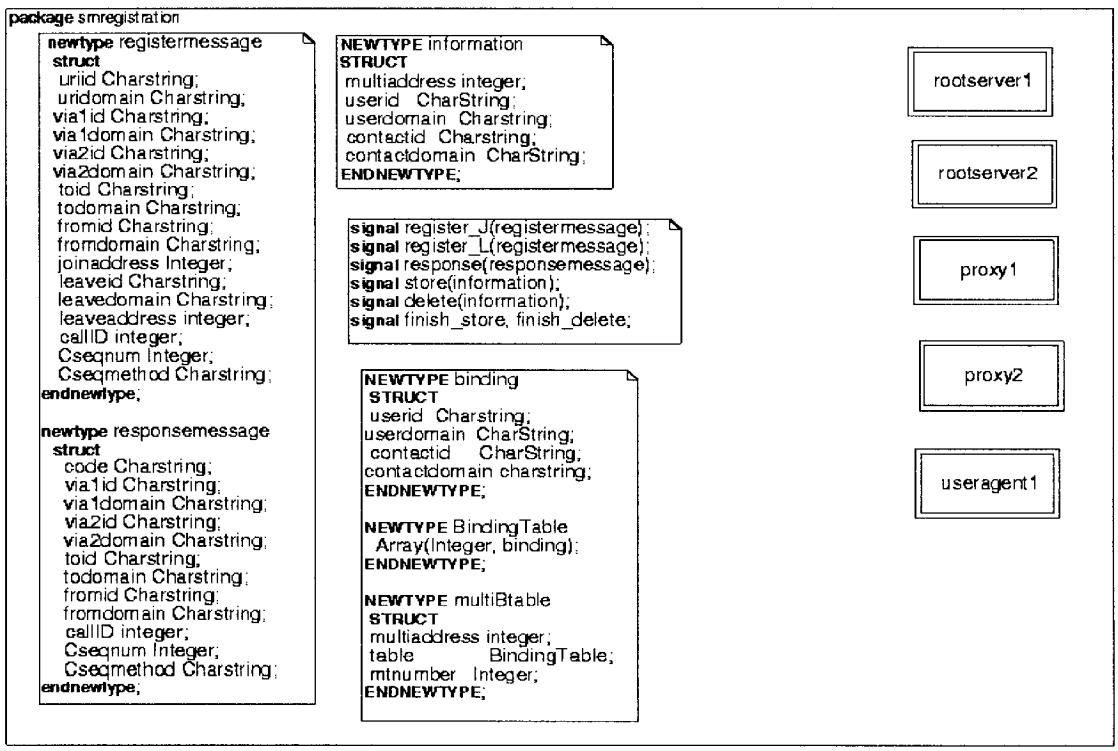

Figure 5-7 Registration Package

The system includes a user who has moved from its proxy server "Mece" of its "mcgill_RS" to proxy server "Cece" of "concordia_RS". The user registers itself to

multicast group through the new local proxy server "Cece" server and "concordia_RS". The RS "concordia_RS" checks the "register_J" message and finds that the pre-root server and the pre-proxy server the user wants to leave are in the "mcgill" domain. It creates a message "register_L" and forwards to "mcgill_RS". The "mcgill_RS" and the pre-proxy server "Mece" remove the user's information from their database. The database may have one of the three scenarios: 1) No members left in both pre-RS database and pre-proxy database (shown in Figure 5-8). The "concordia_RS" removes the information of "mcgill_RS" from its database. 2) No member left only in pre-proxy database but has members left in pre-RS database (shown in Figure 5-9). The "concordia_RS" does not remove the information about "mcgill_RS", because there still are other members who wish to receive multicast message. The "mcgill_RS" does not have information about pre-proxy server who has not members left. 3) Members left in both pre-RS database and pre-proxy database (shown in Figure 5-10). In this situation, new RS "concordia_RS" keeps the information about pre-RS "mcgill_RS" and the pre-RS keeps the information about the pre-proxy "Mece".

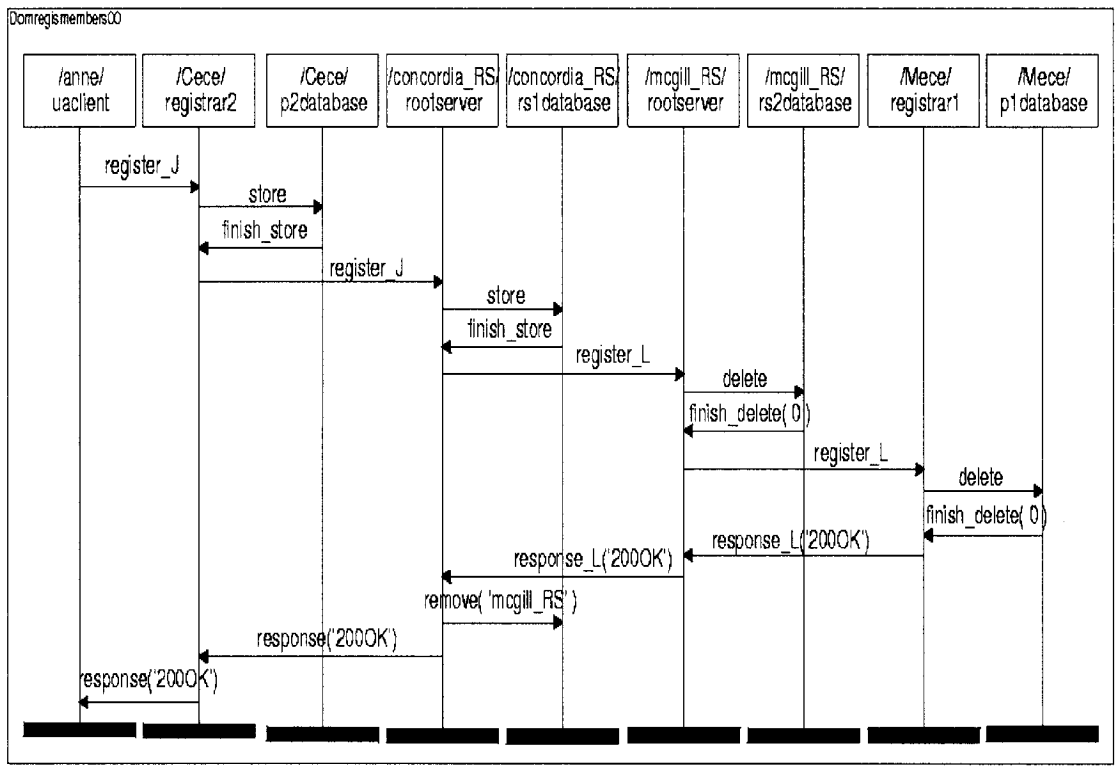

Figure 5-8 MSC for registration (no members in pre-RS and pre-proxy)

In the situation shown in figure 5-8, there is no group member left in the domain of pre-RS and pre-proxy after the MH moves, so the response for the REGISTER_L message includes this information (the number of group member left is 0) to the present RS in order not to forward the message to group members that do not exist there.

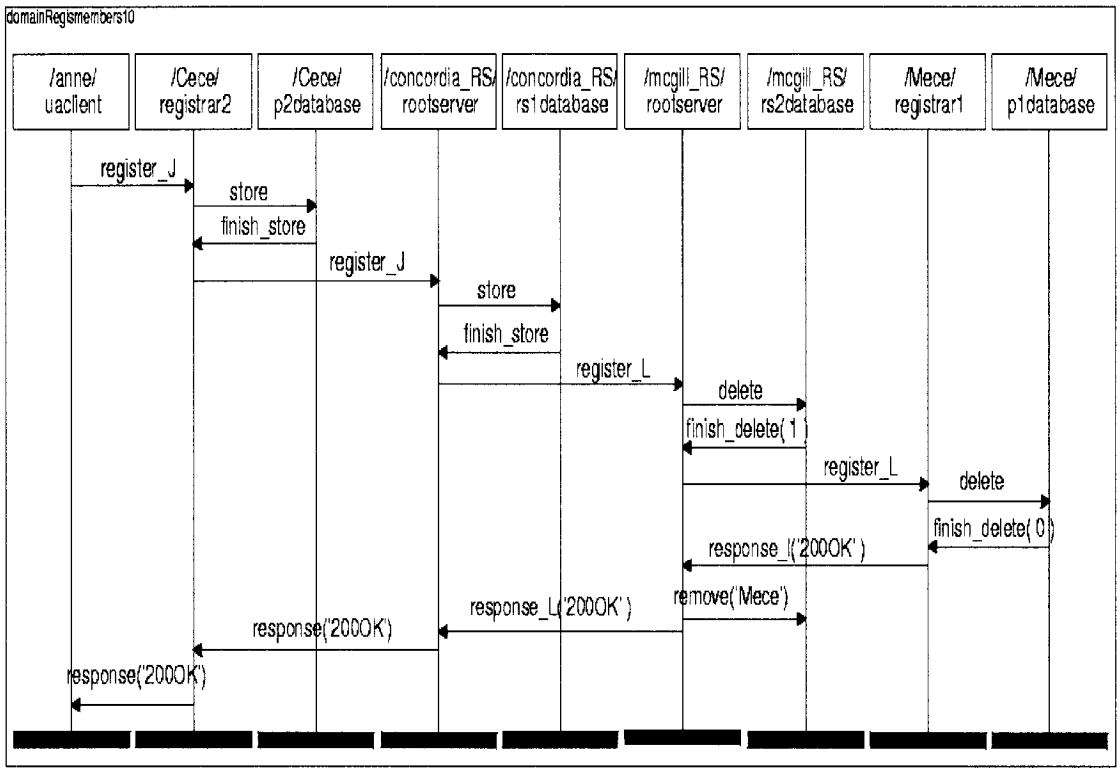

Figure 5-9 MSC for Registration (no members in pre-proxy)

Figure 5-9 shows the situation when there is no member left that belongs to pre-proxy, but there still exists member that belongs to pre-RS. The response for the REGISTER_L message includes this information to the present RS in order to receive the future message to group members that exist in this domain.

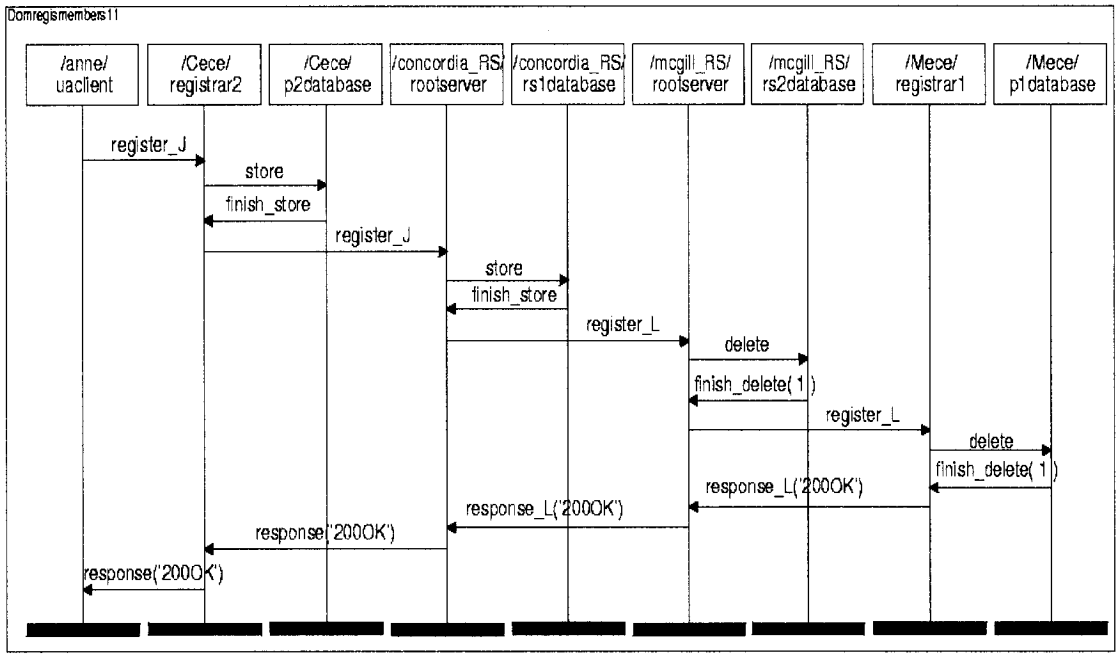

Figure 5-10 MSC for Registration (members in both pre-RS and pre-proxy)

This situation shown in figure 5-10 is similar to the situation shown in Figure 5-9. Although the MH moves, there is still group member existing in this domain. They would like receiving the future message for the multicast group members.

5.3.2 Session Setup

The configuration shown in Figure 5-2 has three domains: concordia, mcgill and uqam with group members "anne", "susan1" and "jack1", "susan2" and "jack2" respectively. Caller "anne" located in the "concordia" domain initiates session setup. Anne sends INVITE message to multicast group members through its local proxy server "Cece" and local RS "concordia_RS". The "concordia_RS" forwards the INVITE message to "mcgill_RS" and "uqam_RS" that has multicast group members. These two RSs deliver the message to its local members ("susan1", "jack1", "susan2" and "jack2"

respectively) through its local proxy server ("Mece" and "Uece"). All the signals/signal lists with their data and data types are defined in package "smsessionsetup" shown in Figure 5-11. Figure 5-12 shows the situation when all four members accept the INVITE. Session will be setup to all members. Figure 5-13 simulates the situation when two members accept the INVITE message. Other two members send 4xxfail message. The caller does not response with ACK message to these two members. The situation when three members accept the INVITE and only one MH refuses invitation by sending back 4xxfail message is similar to Figure 5-13. There is no ACK message to member who refused the INVITE. Figure 5-14 simulates the situation when session setup fails. This happens when all MHs send back 4xxfails message and therefore in this case the caller does not response by ACK message.

In Figure 5-11, we define data structure "sipRegularMessage" that is used by signal "invite" and "ack" as well as data structure "sipResponseMessage" that is used by signal "response". Database checking uses the "userdata" and "groupdata". We also provide data for signal routing. The signal we define here are "invite", "ack", "response", "members_inquiry" and "resp_inquiry". In the package we show all the block types we used in simulation that include all the user agent types, root server types and proxy server types. The following Figure 5-11 shows the details.

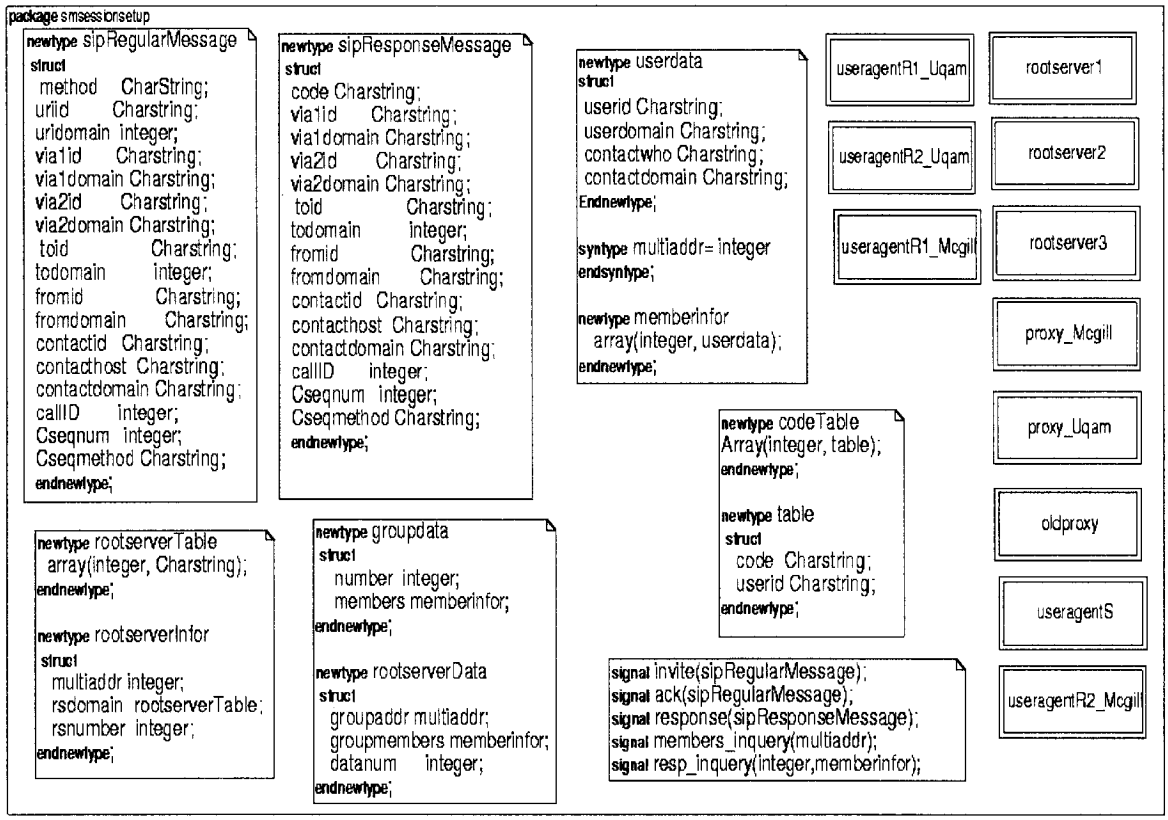

Figure 5-11 Packages for Session Setup

Figure 5-12 shows the situation that four invitees all accept the invitation by sending back 200OK message to inviter. The inviter "anne" sends the INVITE message to multicast group. The group members "susan1", "jack1", "susan2" and "jack2" accept the invitation by sending 200OK to "anne". The inviter "anne" finishes the session setup by responding them ACK message. After receiving ACK, RTP/RTCP media follows up.

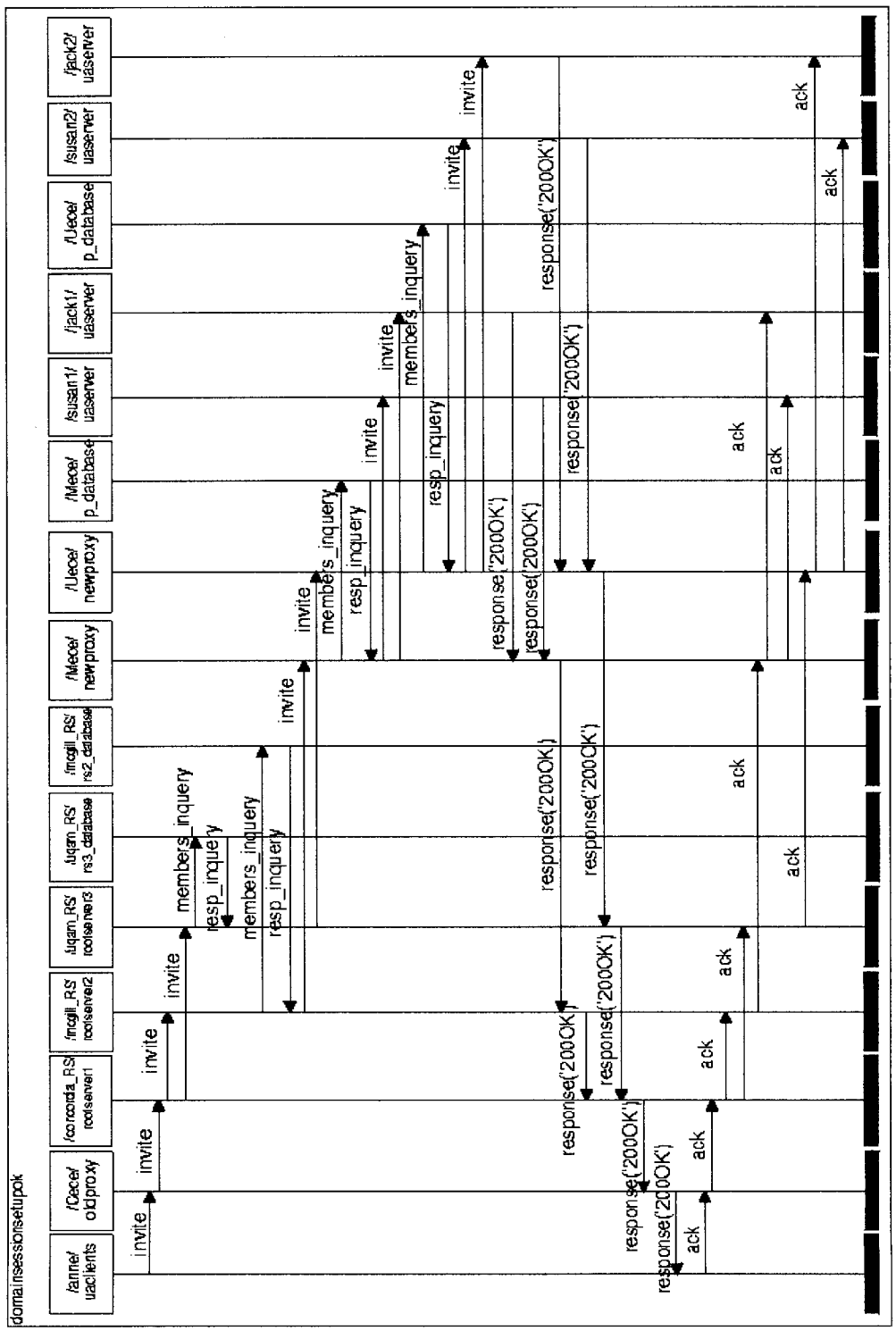

Figure 5-12 Session Setup with 200OK from all MHs

Figure 5-13 shows two out of four group members accepting the invitation. Others refuse the invitation by responding with the "4xxfail" message to inviter.

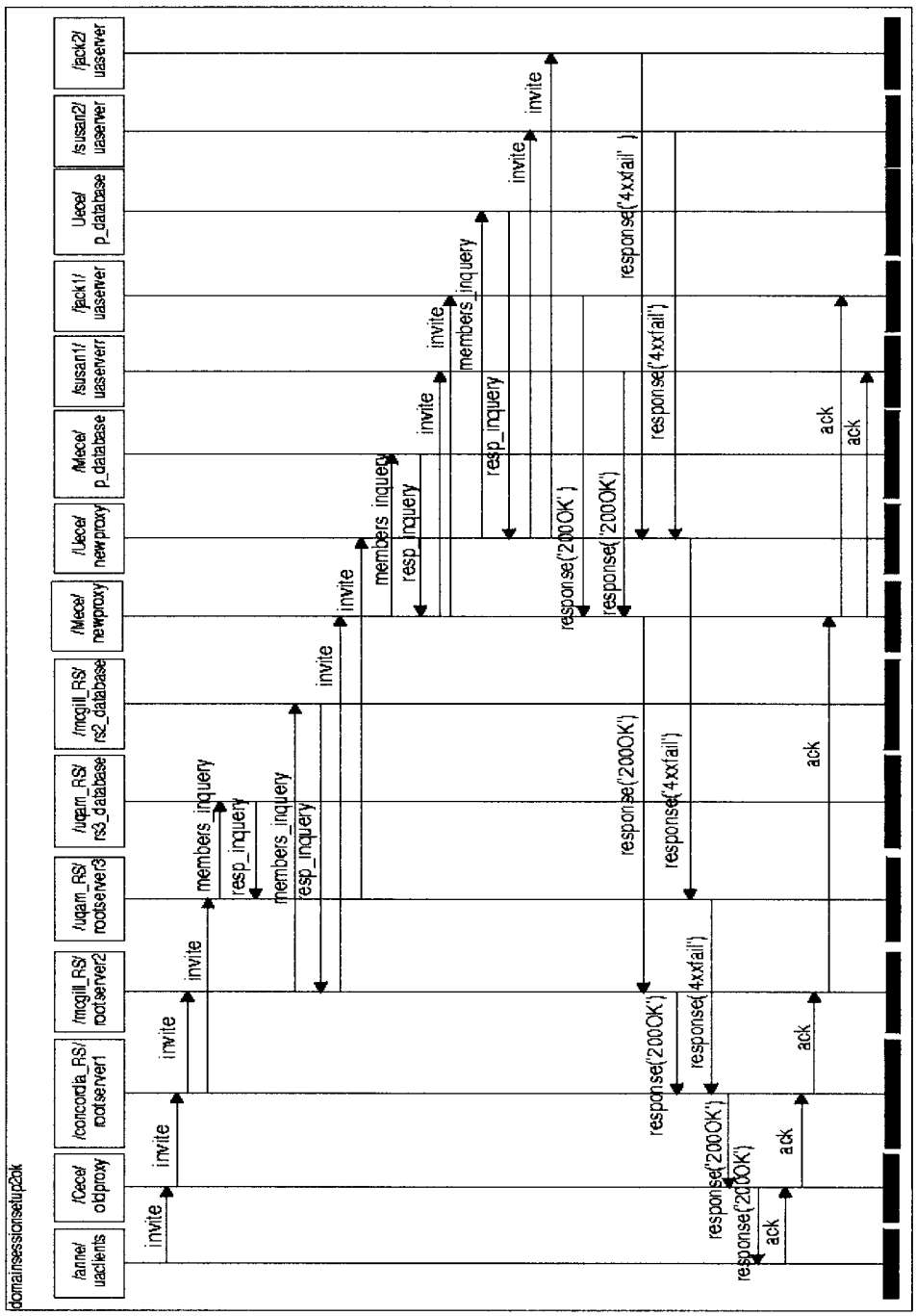

Figure 5-13 Session Setup with 200OK from two out of four MHs

Figure 5-14 provides the result of the situation that all group members refuse the invitation and response with "4xxfail" message to INVITE message from inviter.

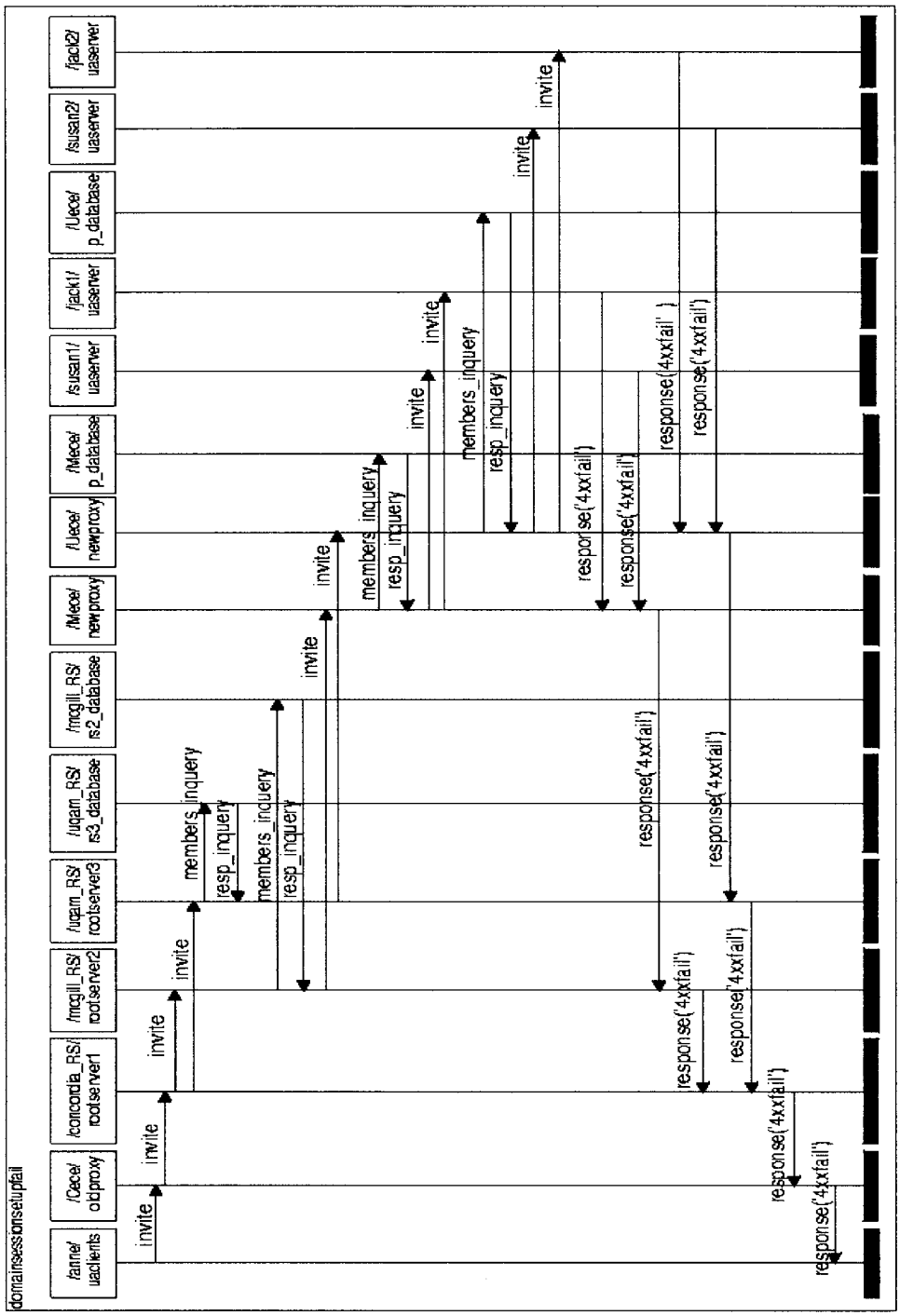

Figure 5-14 Session Setup fail with 4xxfail from all four MHs

5.3.3 Session Termination

The caller or one/more callees who want to terminate their sessions may terminate session. If the caller wants to terminate the session it sends BYE message to all members who are in the session. Members/callees response to the caller through their local proxy servers and RSs. In case of callee/callees terminating the session, only member/members who want to quit the session initiate the termination by sending BYE message to their local proxy server. If the proxy server has one or more than one group member, it terminates the session to this member by sending 200OK message. If all members of this proxy server send it BYE messages, it sends BYE message to its local RS after terminating the session to all members by sending them 200OK messages. The RS follows the same procedures as proxy server.

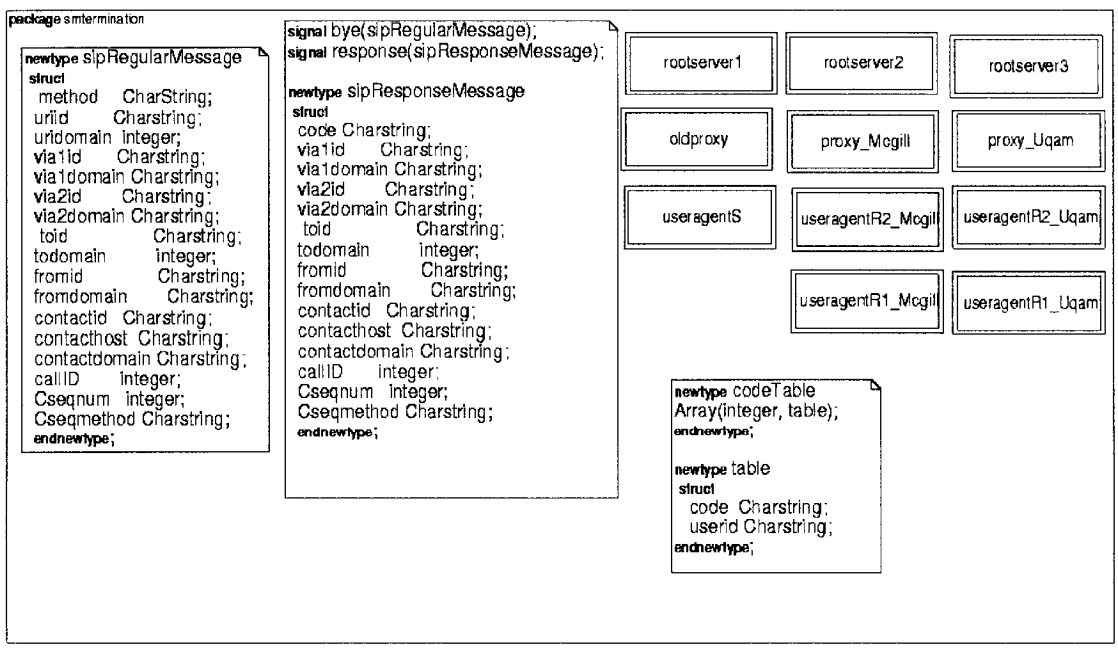

Figure 5-15 Package for Session Termination

Figure 5-15 shows the package used to simulate the termination. Figure 5-16 simulates the session termination from caller that initiates termination. The case when only one MH wants to quit session is shown in Figure 5-17. Figure 5-18 and Figure 5-19 present the situation when termination is initiated by two MHs that are in the same domain and in the different domain respectively. Figure 5-20 shows the case when all MHs want to terminate the session.

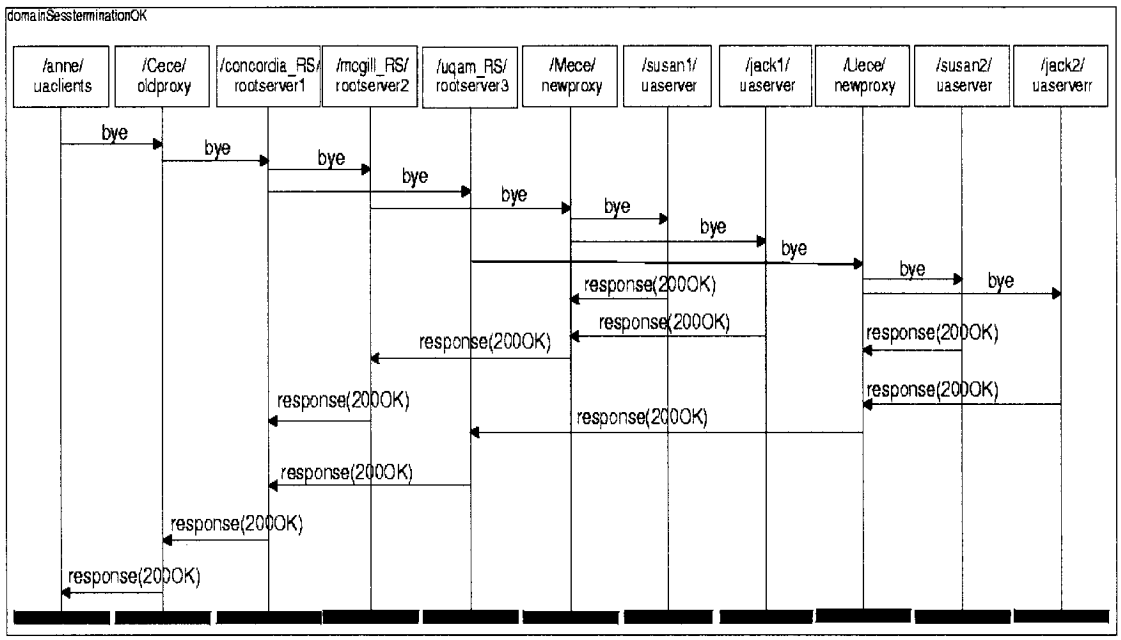

Figure 5-16 Session Termination from caller to callees

Figure 5-17 shows the case that one member "jack1" wants to quit the session. "jack1" sends "bye" message to its local proxy server "Mece". "Mece" terminates the session to "jack1" by sending back 200OK. "Mece" still has other group member/members in its area, so it keeps the session and does not send "bye" message to its RS.

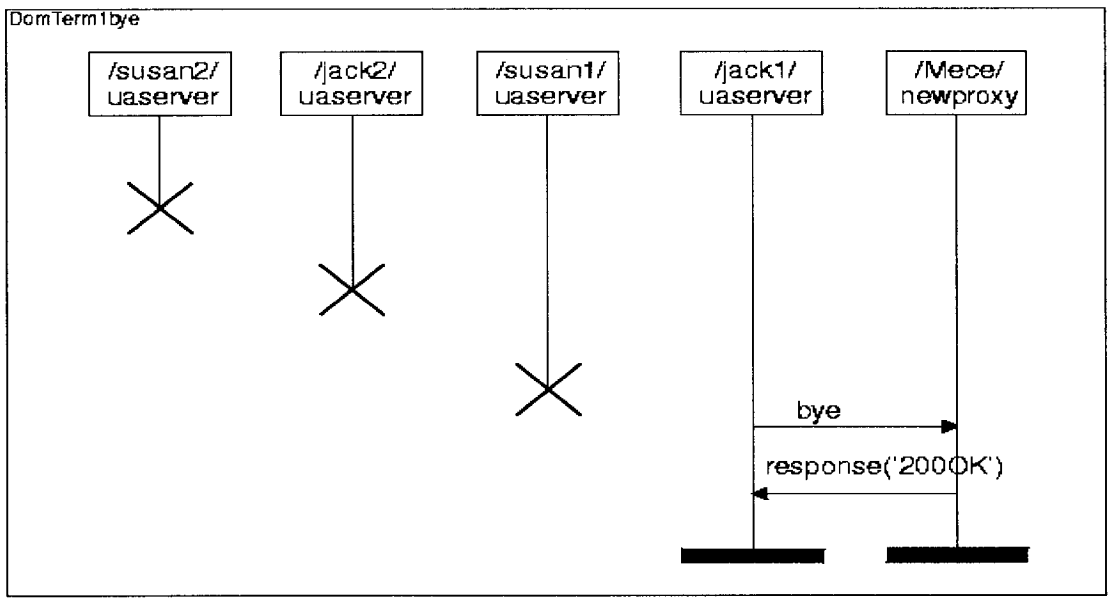

Figure 5-17 Session Termination from one callee

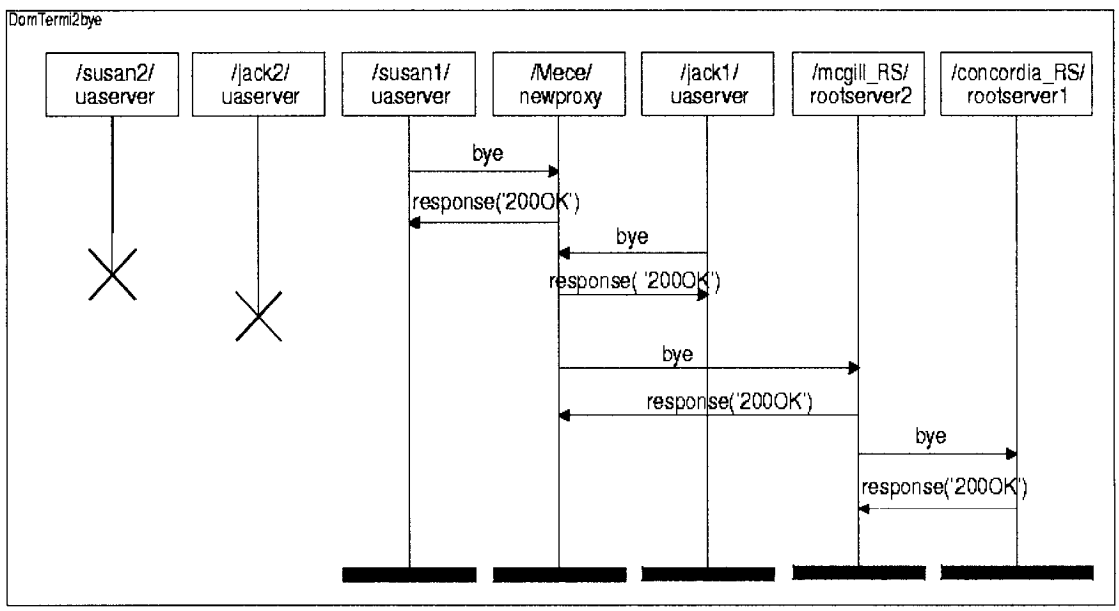

Figure 5-18 Session Termination from two callees belonging to the same domain

In the situation shown in Figure 5-18, "Mece" receives "bye" message form all its local group members. It sends "bye" message to RS "mcgill_RS" after it terminate the session to all members by sending 200OK messages to them.

The case as shown Figure 5-19 is similar to the case shown in Figure 5-17. The difference is that two group members that want to quit the session are belong to different domain respectively.

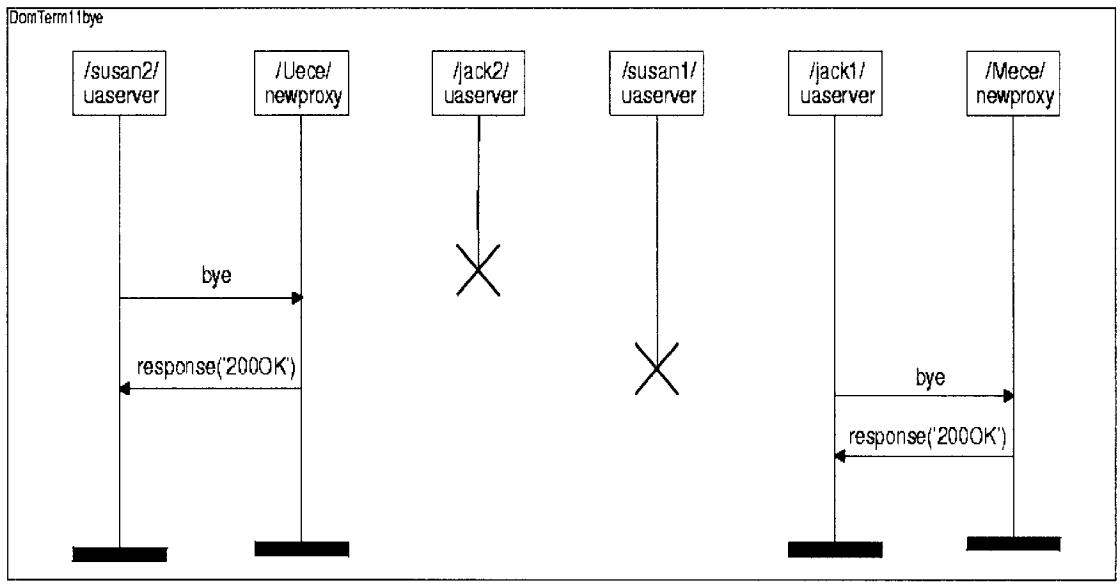

Figure 5-19 Session Termination from two callees belonging to different domains

Figure 5-20 shows the situation when all group members want to quit the session. The local proxy servers ("Uece" and "Mece") send "bye" message to their RSs ("uqam_RS" and "mcgill_RS") after terminating the session to their local group members. Then RSs follow the same procedure to terminate the session to them and request "concordia_RS" to terminate the session. The termination finishes until the caller "anne" receives "bye" message and responds with 200OK.

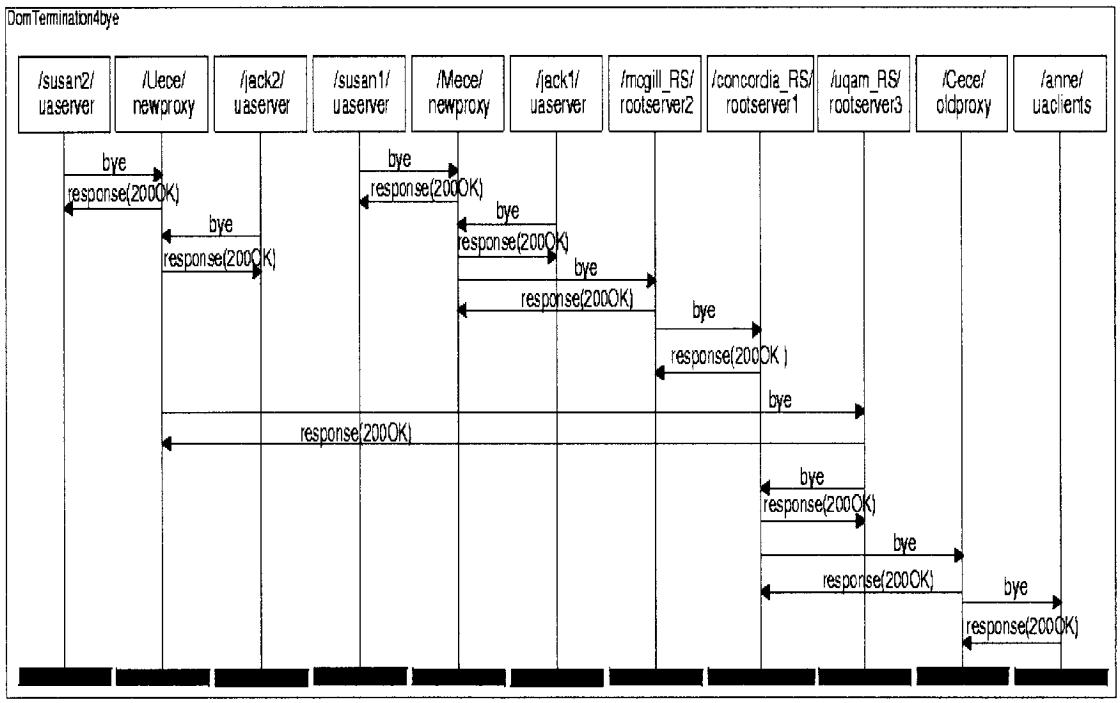

Figure 5-20 Session Termination from all callees

The MSC diagrams from Figure 5-16 to Figure 5-20 show all the situations of session termination that includes BYE message from caller and callee/callees.

5.4 SUMMARY

We present the theory of our new proposal in Chapter 4. All the work to simulate and verify the proposal was discussed in this chapter. We used SDL to build the system model, then used ObjctGeode toolset to simulate and verify them if it works as we described it in Chapter 4. All the results were shown using MSC diagrams. According to the MSCs, the proposal was shown to work well in the normal situation. All the simulations followed the same sequences and steps as in Chapter 4.

CHAPTER 6

CONCLUSIONS

This thesis first presents the problems existing in the IP mobile multicasting, and proposes a solution by using SIP multicast mobility. It is well known that SIP already supports mobility and multicast depending on IP layer. Based on this basic theory, we address a new idea to implement the multicast mobility in application layer and so to be independent of the IP layer. This thesis shows the architecture of SIP multicast mobility in which we added a new entity RS and extended REGISTER message with "join" and "leave" fields. Here we focus on how to make multicast mobility work well. Regarding issues like message duplication and message lost, by comparing RTP/RTCP (The Real-Time Transport Protocol/ The RTP Control Protocol) sequence number, message duplication can be avoided and message lost can be ignored for real time media. We use SDL to build the system model and use ObjectGeode tool to build the system models and to simulate. We have found that our model is able to realize specific interaction scenarios for host registration, session setup and session termination described informally using UML. The work we have done proves that the idea is feasible. Up to how to deal with handoff, we have considered the cases of the destination foreign domain with group members and without members in the new domain.

6.1 CONTRIBUTIONS

The major contributions of this thesis are:

1. Propose an experimental topology for the Mobility with multicast in application layer that help to eliminate the issues existing in Mobile IP.
2. Develop the detailed message based on the SIP unicast message field to support the new behavior.
3. A laboratory prototype is built and simulation results show its feasibility.

6.2 OPEN ISSUES AND SUGGESTIONS FOR FUTURE WORK

6.2.1 Open Issue

During simulation of the proposed protocol, we find that there are some issues on dealing with the group members' responses. In our simulation, we chose to forward the 200OK messages after receiving all group members' responses. Actually this is not the best way to deal with this situation. Comparing to the method that 200OK responses is forwarded as soon as the first 200OK responses from one of the group member is received, the method we adopted use more time than the method above. But for the method we did not use, the problem occurs about how to deal with the responses from other group members that arrive later. For example, one RS has two group members (MH), RS forwards the response as soon as it receives the first 200OK for INVITE request from one of group members. But the second 200OK from another member may

arrive after ACK message. In this situation, The MH, that response with late INVITE request, will lose media.

6.2.2 SIP Mobility Multicast versus Mobile IP multicast

In the case of SIP-based multicast with mobility, multicast support is provided at the application layer. By running distributed algorithms, receivers of a multicast session organize themselves in an overlay network. All communications are then carried out using unicast between neighbors of this overlay network. This offers the advantage of possible immediate deployment since it can utilize all the flow/congestion control capabilities available for unicast. The disadvantage is higher delay. In the case of Mobile IP multicast, multicast support is provided at the network layer. Multicast routers are needed in this case. These routers are responsible for building and managing multicast distribution tree.

Addressing: The SIP mobility multicast approach to support mobility issues a single temporary address to a mobile host. This eliminates the need for explicit address translation as in Mobile IP.

Packet Forwarding: In Mobile IP multicast, since only the home agent is aware of the care-of address of the mobile host, all packets form CH to a MH suffer from triangular routing. In SIP mobility multicast, the RSs will forward the packets to local proxy servers that multicast the packet to local group members, and triangular routing disappears. Mobile IP Multicast applies tunneling techniques by using IP in IP encapsulation protocol and results in multicast packet header overhead. On the other hand, SIP Mobility Multicast does not have such packet overhead.

Location Management: In Mobile IP, CH uses the home address of a MH. Thus, it is sufficient for a MH to notify its home agent of its current care-of address. In SIP mobility multicast, there is no notion of a home agent. When a server wishes to forward a packet to a MH, it needs to locate the host by checking its registration database.

6.2.3 Future Work

The suggestions for the future work:

- A control methodology for fast and smooth handoff.
- Comparing the functionality of SIP mobility and Mobile IP for supporting multicast.
- Combining SIP mobility with Mobile IP to find a more efficient way for mobility supporting multicast.

REFERENCES

[1] S. Deering. *Host Extention for IP Multicasting.* IETF RFC 1112, August 1989.

[2] B. Quinn, K.Almeroth. *IP Multicast Applications: Challenges and Solution.* RFC 3170, September 2001.

[3] Andrew Adams,William Siadak. *Protocol Independent Multicast-Dense Mode (PIM-DM): Protocol Specification (Revised).* IETF Draft, draft-ietf-pim-dm-new-v2-05.txt, June 2004.

[4] Deering, D. Estrin, D. Farinacci, V. Jacobson, et al. *Protocol Independent Multicast - Sparse Mode(PIM-SM): Protocol Specification.* IETF RFC 2362, June 1998.

[5] R. Boivie, Y.Imai, W.Livens, D.Doms, O.Paridaens. *Explicit Multicast (Xcast) Basic Specifications.* IETF Draft, draft-ooms-xcast-basic-spec-06.txt, June 2004.

[6] G. Xylomenos and G. Polyzos. *IP multicasting for wireless Mobile Host.* IEEE MILCOM, Volume 3, pp. 933-937,1996.

[7] Charls E. Perkins, Sun Microsystems. *Mobile IP.* IEEE Communications Magazine, Volume 35, Number 5, pp. 66-82, May 1997.

[8] Tim G.Harrison, Carey L.Williamson et al. *Mobile Multicast (MoM) Protocol:Multicast Support for Mobile Host.* Proceedings of the Third Annual International Conference on Mobile Computing and Networking, pp. 151-160, September 26-30, 1997, Budapest, Hungary.

[9] C.L. Tan and S. Pink. *Mobicast: a Multicast Scheme for Wireless Networks.* Mobile Networks and Applications, pp. 259-271, Volume 5, Number 4, 2000.

[10] C.R. Lin and K-M. Wang. *Mobile Multicast Support in IP Networks.* IEEE INFOCOM, pp. 1664-1672, March 2000.

[11] D. Thaler, A.Aggarwal, C.Vicisano, D.Ooms. *IPv4 automatic Multicast without Explicit Tunnels (AMT).* IETF Draft, draft-ietf-mboned-auto-multicast-02.txt, February 2004.

[12] R. Finlayson. *The UDP Multicast Tunneling Protocol.* IETF Network Working Group, Internet-draft, draft-finlayson-umtp-09.txt, November 2003.

[13] D.Pendarakis, S.Shi, D.Verma, and M.Waldvogel. *AMLI: An Aplication Level Multicast Infrastructure.* In Proceedings of the 3rd USENIX Symposium on Internet Technologis and Systems (USITS), pp. 49-60, 2001.

[14] S.Banerje, B.Bhattacharje, and C.Kommareddy. *Scalable Application Layer Multicast.* Proceedings of ACM SIGCOMM, pp.205-217, August 2002.

[15] M.Castro, P.Druschel, A.M.Kermarrec, and A.Rowson. *SCRIBE: A Large-Scale and Decentralied Application-Level Mulicast Infrastructure.* IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, Vol. 20, Issue 8, pp. 100-110, October 2002.

[16] H.Schulzrinne, Elin Wedlund. *Application-Layer Mobility Using SIP.* ACM SIGMOBILE Mobile Computing and Communications Review, Volume 4, Number 3, pp. 47-55, July 2000.

[17] C. Perkins. *Minimal Encapsulation within IP.* RFC 2004, IETF, Oct. 1996.

[18] C. Perkins. *IP encapsulation within IP.* RFC 2003, IETF, Oct. 1996.

[19] Beau Williamson. *Developing IP Multicast Networks.* Published by Cisco Press, 2000. ISBN: 1-57870-077-9.

[20] Hrishikesh Gossain, Carlos Morais Cordeiro and Dharma P.Agrawal. *Multicast: Wired to Wireless.* IEEE Communications Magazine,Number 6, pp. 116-123, June 2002.

[21] I.Romdhani, M.Kellil, H-Y. Lach, A.Bouabdallah, H.Bettahar. *IP Mobile Multicast: Challenges and Soutions.* IEEE Communications Surveys, Volume 6, No.1, pp.18-41, 2004.

[22] S.Deering, D. Estrin, D. Farinacci, and V. Jacobson. *An Architecture for Wide-Area Multicast Routing.* Proceeding of the 1994 ACM SIGCOMM Confrence, London, UK, pp. 126-135, August 1994.

[23] Hamad el Allali and Cristian Hesselman. *Multicasting with Mobile IP &The Session Initiation Protocol.* Technical Report. Telematica Institute. 2000.

[24] Vineet Chikarmane, C.L.Willamson et al. *Multicast Support for Mobile Hosts Using Mobile IP: Design Issues and Proposed Architecture.* ACM Mibile Networking and Application, Volume 3, Issue 4, pp. 365-379, 1998.

[25] Bill Douskalis. *IP Telephony-the integration of roubust VOIP services.* Published by Prentice Hall PTR. 2000 by Hewlett- Packard Company. ISBN 0-13-014118-6.

[26] J.Rosenberg, H.Schulzrinne, G.Camarillo, A.Johnstom, J.Peterson, R.Sparks, M.Handly, E.Schooler. *SIP: Session Initiation Protocol.* RFC 3261, June 2002.

[27] Elin Wedlund, H.Schulzrinne. *Mobility Support using SIP.* ACM/IEEE International Conference Wireless and Mobile Multimedia (WoWMoM'99), pp. 76-82, August 20-20, 1999.

[28] R. Pandya. *Emerging mobile and personal communication system.* IEEE Communications Magazine, Vol. 33, pp 44-52, June 1995.

[29] H.G.Schulzrinne and J.D. Rosenberg. *The Session Initiation Protocol: Providing Advanced Telephony Services Across the Internet.* Bell Labs Technical Journal October-December, pp. 144-160, 1998.

[30] P. Vixie, Ed., S. Thomson, Y. Rekhter, and J. Bound. *Dynamic Updates in the Domain Name System (DNS UPDATE).* IETF, RFC 2136, Apr. 1997.

[31] Melody Moh, Gregorie Berquin, Yanjun Chen. *Mobile IP Telephony: Mobility Support of SIP.* Proc of ICCN, IEEE, pp. 554-559, August 1999

[32] J. Solomon. *Mobile IP-The Internet Unplugged.* Prentice Hall, 1998.

[33] F.Vakil, A.Dutta, M.Tauil, S.Baba, N.Nakajima, Y.Shobatake, H.Schulzrinne. *Supporting Mobility for Multimedia with SIP.* IETF Internet Draft, draft-itsumo-sipping-mobility-multimedia-01.txt, 2001.

[34] *Telelogic ObjectGeode: The Most Advanced Integrated Environment for the Development of Distributed Real-time Systems.* http://www.telelogic.com/products /additional/objectgeode/index.cfm.

[35] International Telecommunication Union. *ITU-TS Recommendation Z.100: Specification and Description Language (SDL).* ITU-TS, Geneva, Switzerland, 1999.

[36] Gerard J. Holmann. *Design and Validation of Computer Protocols.* Bell Laboratories, Murray Hill, New Jersey 07974, ISBN-0-13-539834-7, Prentice Hall,1991.

[37] International Telecommunication Union. *ITU-TS Recommendation Z.120: Message Sequence Chart (MSC).* ITU-TS, Geneva, Switzerland, 1999.

[38] J.Ellsberger, D. Hogrefe, and A. Sarma. *SDL – Formal Object-oriented Language for Communication Systems.* Prentice Hall Europe, ISBN 0-13-621384-7, 1997.

[39] Eric J. Braude. *Software Engineering: An Object-Oriented Perspective*. Boston University. ISBN 0-471-32208-3.