Telecommunications and Information Exchange Between Systems

ISO/IEC JTC 1/SC 6

Document Number:	N14012
Date:	2009-06-10
Replaces:	
Document Type:	Other document (Defined)
Document Title:	Revised text of ITU-T X.604.2 | ISO/IEC 24793-3, Mobile Multicast Communications(MMC) – Protocol over Overlay multicast networks
Document Source:	SC 6/WG 7 Tokyo meeting
Project Number:	
Document Status:	As per the SC 6 Tokyo resolution 6.7.5, SC 6 approves advance authorization for a second FCD of this project.
Action ID:	FYI
Due Date:	
No. of Pages:	86

ISO/IEC JTC1/SC6 Secretariat Ms. Jooran Lee, KSA (on behalf of KATS)

Korea Technology Center #701-7 Yeoksam-dong, Gangnam-gu, Seoul, 135-513, Republic of Korea ;

Telephone: +82 2 6009 4808 ; Facsimile: +82 2 6009 4819 ; Email : jooran@kisi.or.kr

6N14012
7TOK-27

Title: Revised text of MMC-3 (ISO/IEC 24793-3)

Source: ISO/IEC JTC 1/SC 6/WG 7 Meeting (Tokyo, June 2009)

Status: This document is an output document to the MMC-3 (ITU-T X.604.2 | ISO/IEC 24793-3) of June 2009 Tokyo ISO/IEC JTC 1/SC 6/WG 7 Meeting. This document is based on the output document of Jan. 2009 SC 6/WG 7 and ITU-T Q.15/11 joint meeting which is TD 96 of ITU-T.

Five contributions on MMC-3 have been submitted in the June 2009 Tokyo ISO/IEC JTC 1/SC 6/WG 7 Meeting. The submitted contributions are as follows:

- Enhance session and membership control for ISO/IEC 24793-3
- Enhancement in neighbor discovery and multicast session leave for ISO/IEC 24793-3
- Proposed Modification of Maintenance Procedure of MA operation to MMC-3 (ISO/IEC 24793-3)
- Proposal of additional function to MMA and MN for ISO/IEC 24793-3
- Proposed messages and parameters for ISO/IEC 24793-3

Major changes to this document are as follows:

- Enhanced MCS/MA session subscription and session initiation procedures
- Added failure detection and recovery mechanism for SM, MA, MMA, and MN.
- Enhanced leave procdure to MA and Added leave & report procedure for MMA and MN
- New messages added: mLEAVREQ, mLEAVANS, mTERMNOTI, FAILCHECK

The editors have added various “editor’s notes” to indicates works that need to be considered as an input for the next meeting.

Next, the editors have added an attachment at the end of this document to list open issues that needs to be resolved.

Summary

This Recommendation describes Mobile Multicast Communications (MMC) protocol which is an application-layer protocol constructing an one-to-many overlay multicast tree for data delivery from one sender to multiple receivers over fixed and mobile converged network environment. This specification describes detailed functions and procedures of the MMC-3 protocol. The defined protocol can be used as delivery service for applications that require one-to-many data delivery service over fixed and mobile converged network environment. Some examples of MMC-3 service include mobile IPTV service, mobile NEWS ticker service, file distribution, e-learning, etc.

삭제됨: , which

삭제됨: s

삭제됨: It is expected that MMC-3

삭제됨: ;

Contents

삭제됨: 1 Scope 8
2 Normative references 8
3 Definitions 8
3.1 Terms defined elsewhere 8
3.2 Terms defined in this Recommendation 9
4 Abbreviations 9
5 Conventions 9
6 Overview 10
6.1 MMC-3 services 10
6.2 MMC-3 entities 11
6.3 Protocol blocks 11
6.4 MMC-3 control model 14
6.5 MMC-3 data delivery model 14
6.6 Types of control messages 15
7 Protocol operation 15
7.1 Session manager's operation 15
7.1.1 Session initiation 15
7.1.2 Admission control 16
7.1.3 Session monitoring 16
7.1.4 Session termination 16
7.2 Multicast agent's operation 17
7.2.1 Session subscription 17
7.2.2 Neighbor discovery 17
7.2.2.1 Multicast area 18
7.2.2.2 Unicast area 19
7.2.3 Tree join 19
7.2.4 Leave 20
7.2.4.1 When MA leaves a session 21
7.2.4.2 When MA leaves from its PMA – for parent switching 22
7.2.4.3 When MA is kicked out 22
7.2.4.4 When SMA leaves the session 23
7.2.5 Maintenance 23
7.2.5.1 Heartbeat 23
7.2.5.2 Monitoring 23
7.2.5.3 Fault detection & recovery 24
7.2.5.4 Tree improvement 26
7.2.6 Termination 26
7.3 Mobile multicast agent's operation 27
7.3.1 MMA announcement 27
7.3.2 Support session join of MN 27
7.3.3 Support MN handover 27
... [1]

Introduction

The demands of multicast based multimedia services in the mobile environment seem to be grown enormously in the next generation network. Among the multimedia services can be considered, multimedia streaming services are expected the most influence services.

삭제됨: <Optional – This clause should appear only if it contains information different from Scope and Summary>

INTERNATIONAL STANDARD <Nbr>
ITU-T RECOMMENDATION <Nbr>

Information Technology – Mobile Multicast Communications: Protocol over Overlay Multicast Networks

1 Scope

This Recommendation | International Standard describes protocol over overlay multicast networks, which can be used to support a variety of multimedia multicasting services by constructing overlay multicast tree over the IP-based wireless mobile networks as well as the wired fixed networks. This protocol focuses on one-to-many data delivery service types.

삭제됨: path
삭제됨: ;
삭제됨: t

This Recommendation | International Standard specifies the protocol over overlay multicast, which describes the protocol operation and message types.

This Recommendation | International Standard specifies the followings:

a) Overview of protocol; which introduces protocol entities, protocol blocks and message types

b) Protocol operation for one-to-many data delivery

c) Detailed messages and parameters

2 Normative references

The following ITU-T Recommendations and International Standards contain provisions that, through references in the text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations | International Standards listed below. IEC and ISO members maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU-T maintains a list of currently valid ITU-T documents.

– ITU-T Recommendation X.603 (2004) | ISO/IEC 16512-1 : 2004, Information technology – Relayed multicast protocol: Framework

– ITU-T draft Recommendation X.603.1 | ISO/IEC 16512-2, Information technology – Relayed multicast protocol: Specification for for simplex group applications

3 Definitions

3.1 Terms defined elsewhere

This Recommendation|International Standard uses the following terms defined elsewhere:

3.1.1 RMCP session [ITU-T X.603]: A set of MAs that use the RMCP to configure the data delivery path.

3.1.2 Sender multicast agent (SMA) [ITU-T X.603]: The MA attached to the sending application in the same system or local network.

3.1.3 Session identification (SID) [ITU-T X.603]: Corresponds to group name and identifies RMCP session uniquely.

3.1.4 Session manager (SM) [ITU-T X.603]: An RMCP entity that is responsible for the overall RMCP operation; it may be located in the same system as the sending application or located separately from the sending application.

3.1.5 Simplex [ITU-T X.603]: Wherein only one sender is send only and all others are receive only.

3.1.6 Multicast agent (MA) [ITU-T X.603]: An intermediate RMCP entity used to support and manage a relayed multicast data transport over a unicast based Internet; an MA may be installed in the same system as a receiving application.

3.1.7 **MobileIP [IETF RFC 3344]**: An Internet Engineering Task Force (IETF) standard communications protocol that is designed to allow mobile device users to move from one network to another while maintaining a permanent IP address.

3.2 Terms defined in this Recommendation

For the purposes of this Recommendation | International Standard, the following definitions apply:

3.2.1 **Mobile multicast agent (MMA)**: A special MA used to support mobile nodes.

3.2.2 **Mobile node (MN)**: A leaf node of overlay multicast tree; whose location and point of attachment to the Internet may frequently be changed.

3.3.3 **Mobile node identifier (MNID)**: Unique value used to identify MN for a certain session.

3.3.4 **ROOTPATH**: Delivery path from MCS to each MA.

Editor's note: Definition for MMC-3 service, Group service, and Session service will be added.

삭제됨: **identification**

서식 있음: 글꼴: 기울임꼴, 강조

서식 있음: 글꼴: 기울임꼴, 강조

서식 있음: 글꼴: 기울임꼴

4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply:

CMA	Child Multicast Agent
HMA	Head Multicast Agent
IP-IP	IP in IP
KO	Kick-Out
MA	Multicast Agent
MAID	Multicast Agent Identifier
MCS	Multicast Contents Server
MMA	Mobile Multicast Agent
MMC	Mobile Multicast Communications
MN	Mobile Node
MNID	Mobile Node Identifier
PMA	Parent Multicast Agent
PoA	Point of Access
RMCP	Relayed Multicast Protocol
SDP	Session Description Protocol
SID	Session Identifier
SM	Session Manager
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
MN	Mobile Node
MMA	Mobile Multicast Agent
MMC	Mobile Multicast Communications

삭제됨: Identification

삭제됨: Communication

삭제됨: Identification

삭제됨: Identification

삭제됨: SMA Sender Multicast Agent

5 Conventions

<None>

6 Overview

6.1 MMC-3 services

MMC-3(Mobile Multicast Communications Part 3) is an application-level protocol that is an extension of RMCP-2 protocol defined in X.603.1 to support simplex group communication services over fixed and mobile converged network environment. MMC-3 defines two new types of node, which are MMA (Mobile Multicast Agent) and MN (Mobile Node), to support overlay multicast function to the mobile environment. The entities of MMC-3 protocol configure an efficient data delivery path for one-to-many group communications. MMC-3 entities forward group data to each participant along the constructed data delivery path. Some major application services for MMC-3 are mobile IPTV and mobile NEWS ticker service.

Figure 1 shows a typical service model of MMC-3 for supporting one-to-many group communications service over fixed and mobile converged network. The MMA in the MMC-3 protocol is used to support mobility for MN. MMA provides multicast service to MN by multicasting the multicast traffic needed by the MN to the wireless region, which is called a MMA region. MMA also provide mobility for the MN by forwarding multicast traffic to new MMA when MN has moved to new MMA region. The MMC-3 protocol can also be used in a wireless environment where MMA does not exist, called a non-MMA region. MN will need to join the multicast session to the previous MMA to be provided with the on-going multicast session.

As shown in Figure 1, the MMC-3 protocol can provide one-to-many group communications service in both MMA region and non-MMA region. The MMC-3 protocol utilizes the multicast capability inside the MMA region and, also, provides multicast capabilities to entities in the non-MMA region.

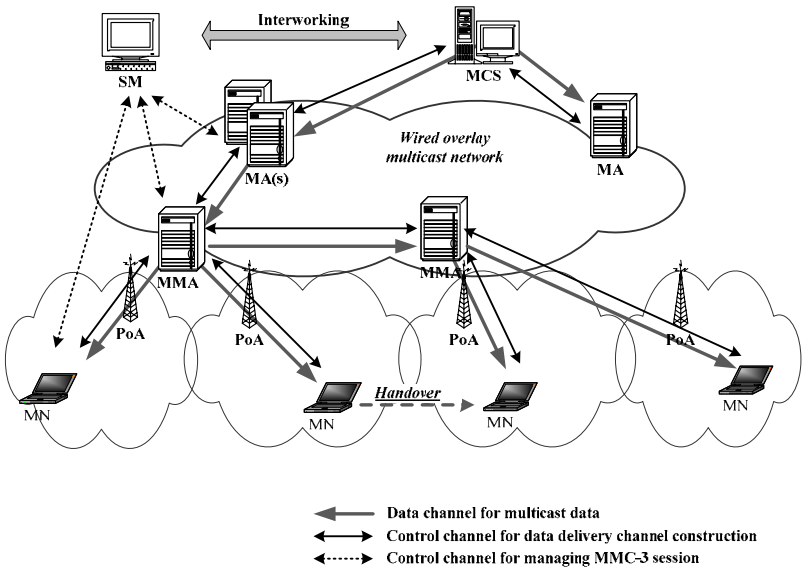


Figure 1 – MMC-3 service model

The entities of the MMC-3 protocol are session manager(SM), multicast contents server (MCS), multicast agent (MA), mobile multicast agent (MMA), and mobile node (MN). SM manages the multicast tree and multicast session. MA is an intermediate node that provides data delivery capability. MMA provides multicast service to MNs which are a user device.

The following features of MMC-3 support the one-to-many group communications over the fix and mobile converged network.

a) MMC-3 constructs a logical control tree by using one or more MAs and MMAs

b) MMC-3 control tree supports the transmission of multicast data in a reliable or real-time manner.

c) MMC-3 control tree consists of logial links between MAs and MMAs.

삭제됨: expanded from

삭제됨: To support both fixed and mobile converged network enviuronment, MMC-3 uses the MMA, Mobile multicast agent, to support mobile node.

삭제됨: MMC-3 can support various application services that require one-to-many group communications over fixed and mobile converged network environment such as

삭제됨:

삭제됨: or

삭제됨: . In MMC-3, the wireless environment where MMA does not exist is

삭제됨: main

삭제됨: and

삭제됨: and

삭제됨: The

삭제됨: The

삭제됨: The

삭제됨: and MN is

삭제됨: a)

서식 있음: 번호 매기기 + 수준:1 + 번호 스타일: a, b, c, … + 시작 번호: 1 + 맞춤: 왼쪽 + 맞춤 위치: 39.7 pt + 탭 간격: 0 pt + 들여쓰기 위치: 59.2 pt

삭제됨: ; the

d) MMC-3 has the capability of selecting optimal peers to configure logical links. The selection of optimal peers can be based on various metrics; example of such metrics includes hop count, delay, and/or bandwidth.

c) MMC-3 supports pure IP multicast, NAT/Firewall.

d) MMC-3 allows participants to join or leave at any time during a session.

e) MMC-3 manages the participants of a session which includes membership monitoring and expulsion of members.

f) MMC-3 provides an auto-configuration mechanism for the group communications path.

g) MMC-3 provides network fault detection and service recovery.

h) MMC-3 provides multicast service to MN through MMA.

i) MMC-3 supports seamless handover of MN.

j) MMC-3 has various ways of managing the session; e.g., tightly or loosely.

삭제됨: <#>
b)

삭제됨: ; because the MMC-3 control tree consists of logical links between MAs, and each MA

삭제됨: s a

삭제됨: based on the selected peers

삭제됨: , and different versions of IP

삭제됨: ; the capability of managing the session

6.2 MMC-3 entities

This clause provides description of five entities of MMC-3 protocol which are SM, MA, MCS, MMA, and MN. SM (Session Manager) manages group membership, multicast sessions. MA (Multicast Agent) is an entity that constructs multicast data delivery path from senders and receivers and forwards data along the constructed path. MCS (Multicast Contents Server) is a specific form of MA which is the sender of the multicast data and also the root of the logical control tree.MMA (Mobile Multicast Agent) is also a specific form of MA which assists MN (Mobile Node) in be provided with MMC-3service. As shown in Figure 1, one MMA can handle one or more wireless access network. MN (Mobile Node) is leaf node of overlay multicast tree with capability of changing the location and point of attachment.

삭제됨: The following figure shows an example network configuration of the MMC functional entities over relayed multicast; MCS, MA,

삭제됨: , and SM

삭제됨: MA can be categorized by MA in wireless access network which is called MMA and MA in relayed multicast backbone network.

It is noted in the figure that the network entities are hidden such as multicast routers and MA can be implemented as an end-system, server, or hardware set-top box; the ways of implementing MA are out of scope of this Recommendation.

삭제됨: document's scope

SM has the following functionality:

삭제됨: A

삭제됨: can provide

a) Session initialization;

b) Session termination;

c) Membership management;

d) Monitoring session status.

MA has the following functionality:

삭제됨: A

삭제됨: , which refers both MA and MMA, constructs a relayed multicast delivery path from one sender to many receivers and then forwards data along the constructed path, can

삭제됨: provide

a) Session initialization;

b) Session subscription;

c) Session tree join;

d) Session leave;

e) Session management;

f) Reporting session status;

g) Data delivery.

MMA has the following additional functionalities along with functionalities of MA:

삭제됨: To support MN,

삭제됨: should provide

a) MMA announcement;

b) Support session join of MN;

c) Support MN handover;

d) Data channel differentiation;

e) MN management

MN is the leaf of the relayed multicast delivery path and has the following functionality:

삭제됨: A

삭제됨: attaches to

삭제됨: a

삭제됨: provides

a) Session subscription;

b) Session tree join;

c) Session leave;

d) Session maintenance;

e) Seamless handover.

MCS is the root of the MMC-3 tree which provides contents to the MMC-3. MCS can be placed in the device which operates as an MA or in the independent device.

삭제됨: Wireless access network movement and fast

서식 있음: 표준

6.3 Protocol blocks

MMC-3 uses two different types of protocol blocks. The first block is used for controlling MMC-3 session, and the other block is used for delivering group data. Since SM is only used to control the MMC-3 session, it has only a control module. On the other hand, since both MA and MMA are used for control and data delivery, they consist of two modules, which are control module and data module. MN is used to receive group data. To receive group data, MN consists of both control module and data module.

삭제됨: second

삭제됨: SM only

Figure 2 shows the three types of path and interfaces that are used in MMC-3.

- MMC-3 control path between SM and other entity; MA, MMA, or MN and between MAs/MMAs or between MN and MMA;
- Data path between MA/MMA data modules or between MN data module and MMA data module;
- Local interfaces inside the MA, MMA, or MN; that is, between the control module and the data module.

삭제됨: /

삭제됨: /

삭제됨: /

삭제됨: /

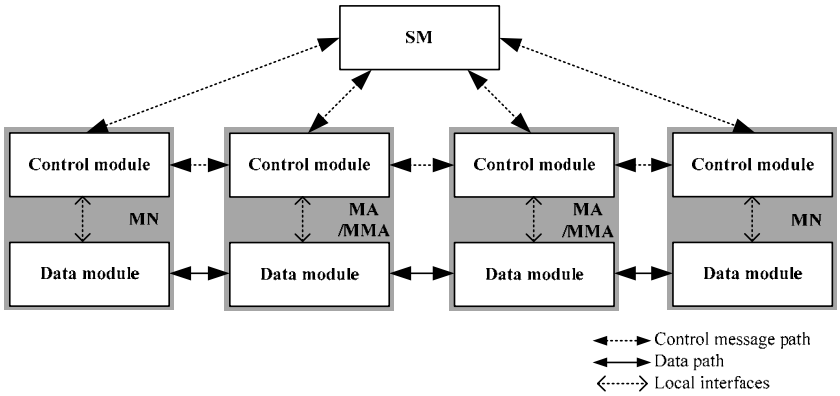


Figure 2 – Three types of path and interfaces in MMC-3

MMC-3 needs to use reliable protocol transport in exchanging messages to construct a robust and reliable multicast session. Thus, MMC-3 uses TCP in transmitting control messages.

SM should exchange control messages with other entities, MAs, MMAs, and MNs, to control and manage a group communication session. The control messages used by SM should be delivered reliably otherwise the session becomes unstable. For reliable delivery of control messages, SM uses TCP for transport protocol. The following Figure 3 shows a protocol stack of SM.

삭제됨: /

삭제됨: /

삭제됨: unrecoverable

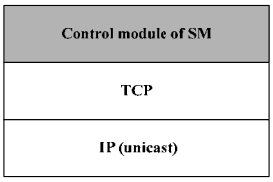


Figure 3 – Protocol stack of SM

MA, which refers to both MA and MMA, constructs a relayed multicast delivery path from one sender to many receivers and then forwards data along the constructed path. MA consists of a control module and a data module. The control module establishes the data delivery path. The data module sets up a data channel along the path constructed by the control module and then relays data through the data channel.

삭제됨: the

삭제됨: the

삭제됨: relayed

The MA's control module configures the control tree from the MCS to every leaf nodes by exchanging control messages with other MAs or with MMAs; also the control module is used for session control and management by SM.

The MA's data module relays application data along the tree configured by the control module. Data module of MMA is used to provide MMC-3 service to MN. Figure 4 shows the protocol stack of MA's data module; any kind of transport protocol can be used if needed.

To ensure that any kind of data transport mechanism can be adopted, two MAs (namely, the upstream MA and downstream MA) establish a data channel on the control tree by exchanging the data profiles described later.

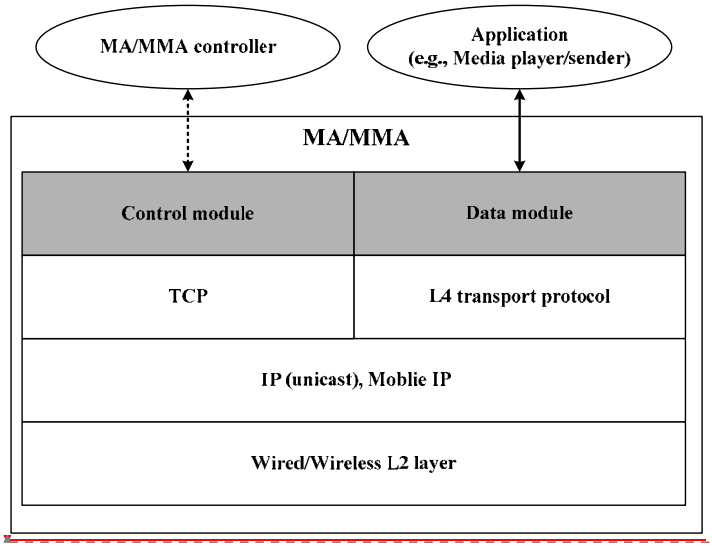


Figure 4 – Protocol stack of MA/MMA

The topologies of the two paths for control and data delivery are usually same because a data delivery path is constructed along the control tree. After the data delivery path is constructed, the application data sent by the MCS can be delivered to each MMA. At last, each MMA which receives the application data can provide the received data to MN.

MN regists to a MMA and then it can receive session data. An MN consists of a control module and a data module. The control module enables registration to MMA and handover. The data module enables data receiving from MMA.

Figure 5 shows the protocol stack of an MN.

삭제됨: MMAs
삭제됨: multicast
삭제됨: construct
삭제됨: delivery path
삭제됨: the
삭제됨: ,
삭제됨: And then
삭제됨: from
삭제됨: leaf
삭제됨: from MCS
삭제됨: application
삭제됨: s
삭제됨: multicast
삭제됨: movement
삭제됨: And t
변경된 필드 코드

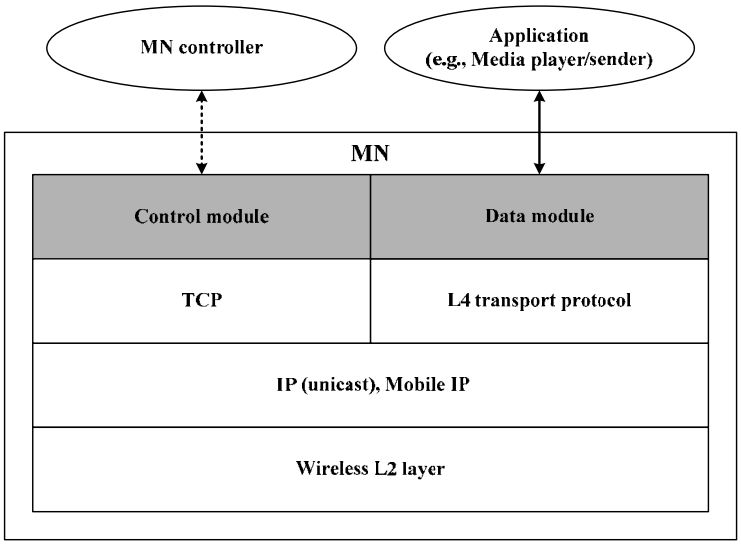


Figure 5 – Protocol stack of MN

The MN's data module receives application data from the MMA. Any kind of transport mechanism can be used if needed.

Note that MN does not join the control tree; registration to MMA does not mean joinning the control tree. Since MN can move frequently, MN must not join the control tree for preventing control overhead.

삭제됨: inserted

삭제됨: It is important

삭제됨: .

삭제됨: R

6.4 MMC-3 control model

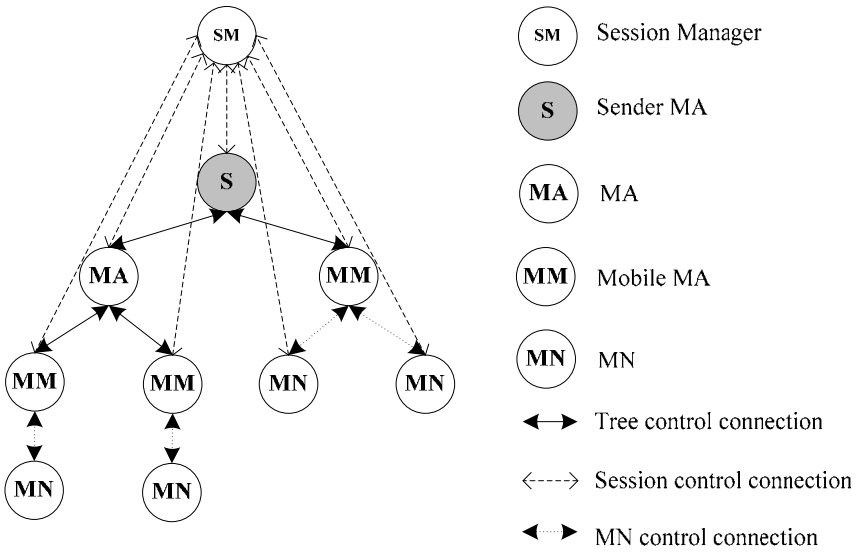


Figure 6 – Control connection between MMC-3 entities

MMC-3 control tree consists of one Sender MA and zero or more MAs and MMAs. The following are the control connections that exist in the control tree:

- Connections in among MAs and MMAs forming tree;
- Direct connection among SM, MA and MMA.

삭제됨: /

삭제됨: between

삭제됨: and

삭제됨: s

삭제됨: /

삭제됨: s

As mentioned above, The MN does not join the control tree. However there exists direct connection beteween SM and MN for session subscription.

삭제됨: s

6.5 MMC-3 data delivery model

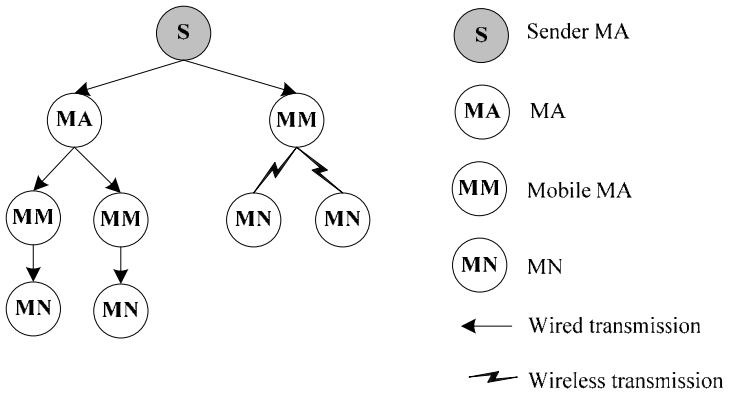


Figure 7 – MMC-3 data delivery model

Figure 7 shows the data delivering model. The MMA transmits the data to MNs through wireless transmission. It is possible that the MMA transmits the data to MNs through wired transmission if MN is in the non-MMA region; note that MMA does not transmit the data to MN directly. MMA sends the data to wireless access points in the non-MMA region and the wireless access points send the received data to MN through wireless transmission.

Editor's note: Description of data delivery model should be enhanced to cover real-time & reliable data delivery.

서식 있음: 수준 1

6.6 Types of control messages

Table 1 lists the MMC-3 messages with its meaning and the operation that is used.

Table 1– MMC-3 messages

Messages	Meaning	Operation
SUBSREQ	Subscription request	Session subscription
SUBSANS	Subscription answer	
PPROBREQ	Parent probe request	Neighbor discovery
PPROBANS	Parent probe answer	
RELREQ	Relay request	Data channel control
RELANS	Relay answer	
LEAVREQ	Leave request	Session leave
LEAVANS	Leave answer	
TERMREQ	Termination request	Session termination
TERMANS	Termination answer	
HSOLICIT	Head MA solicit	Management for multicast enabled network
HANNOUNCE	Head MA announce	
HLEAVE	Head MA leave	

삭제됨: Subscribe

삭제됨: initiation

삭제됨: Subscribe

HB	Heartbeat	Session tree maintenance
STREQ	Status report request	Session monitoring
STANS	Status report answer	
STCOLREQ	Status collect request	
STCOLANS	Status collect answer	
mADVERTISE	MMA advertisement	MMA announcement
mSOLICIT	MMA solicitation	MMA solicitation
mREGISTREQ	MN registration request	MN registration
mREGISTANS	MN registration answer	
mLEAVREQ	MN leave request	MN leave
mLEAVANS	MN leave answer	
mTERMNOTI	MN termination notification.	Session termination notificiation to MN
FAILCHECK	Failure check	Failure check

7 Protocol operation

7.1 Session manager's operation

7.1.1 Session initiation

To make the SM create a new session, a multicast contents server (MCS) should send SUBSREQ message including a session profile, which includes details to create a session such as the session name, media characteristics, and the group address. To distinguish the sessions from each other, the SM creates globally unique session identifier (SID) based on the group address and its own IP address. After a successful session creation, the SM answers using SUBSANS message including the SID and MAID. The MCSs may announce the session creation by using a web server or email. The MCSs may not need to announce the session creation if MCSs use dedicated group address which service users already know such as TV channel. However the way of session announcement is out of scope this Specification.

It can be happen that the group address in the subscription request is already used by another session. In that case, the SM denies the subscription request and notifies MCS that the group address is already used. Then the MCS chooses other group address and requests subscription again. If the modified session profile includes unique group address, then the SM allows MCS to create and to subscribe the session.

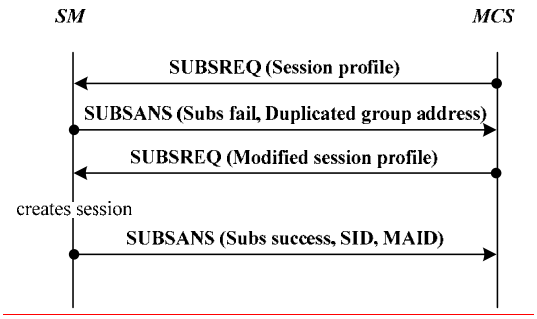


Figure 8 – Session initiation failure because of duplicated group address

After the successful session creation, the SM waits for a subscription request from MAs or MNs.

삭제됨: communication
삭제됨: provide
삭제됨: identification
삭제됨: returns
삭제됨: to the MCS
삭제됨: But
서식 있음: Figure
삭제됨: the
삭제됨: /
삭제됨: When the SM receives a subscription request from an MA/MN, the SM decides whether to accept the subscription request.

7.1.2 Membership control

On receiving subscription request from MA or MN, firstly the SM checks the SID in the request message, and then determines whether the request is acceptable according to the policies defined by the MMC-3 service administrator. The policies can be any criteria that can determine if the requesting MA or MN is a legitimate MMC-3 participant. MMC-3 session can be operated privately as well as publicly with some extra information such as system information and authentication information.

When the SID in the subscription request is valid, then the SM checks proposed MAID (MNID, in case of MN) and proposed data profile. If the MAID (MNID, in case of MN) proposed by MA has null or duplicated value, then the SM proposes a unique one; otherwise, the proposed MAID (MNID, in case of MN) will be used during the session. If the proposed data profile cannot be supported, the SM should reject the request with an appropriete reason. Otherwise, the SM can negotiate for the most effective data profile and sends back with the negotiated one.

When the subscription request from MA or MN is granted, then the SM responds with a confirmed MAID or MNID, neighbor list which includes the information of MAs or MMAs, and session dependent information. The SM should keep the information of MA, MMA, and MN. What exact information the SM should manatain is not defined in this document but identification (MAID or MNID), subscribed SID, uptime, and entity type (MA, MMA, or MN) can be included.

It is important that neighbor list may vary according to subscription requestor. In case of MA, including MMA, the SM gives the list of MAs that serves requesting session. However the SM gives the list of MMAs that serves requesting session when MN requests session subscription because MA does not have functionalities to support MN.

To kick out a specific MA, MMA, or MN, the SM starts the discard procedure by sending a LEAVREQ message with a reason code Kicked-Out (KO) and then updates session member list. Upon receiving SM's LEAVREQ message, MA, MMA, or MN leaves the session as soon as possible. Figure 9 illustrates the procedure, where the SM sends a LEAVREQ message with the reason code KO to MA B and then the MA B notifies its PMA and CMAs of its expulsion. When the MA B receives LEAVANS messages from PMA and all of its CMA, it leaves the session. If the MN which is in MMA region receives a leave request from the SM, it leaves the session promptly without any notification. If the MN which is in non-MMA region receives a leave request, it leaves the session with notifying its serving MMA of its expulsion. When MMA receives a LEAVREQ message from SM, MMA sends

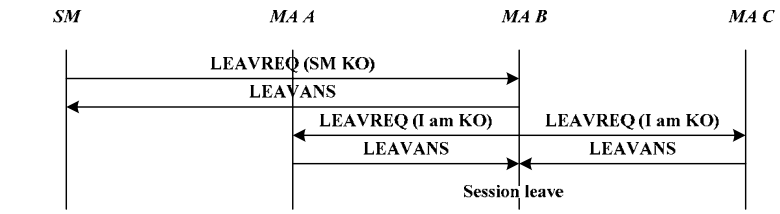


Figure 9 – When MA is kicked out by SM

7.1.3 Session monitoring

The SM can fetch status information of a specific MA by exchanging a status request and answer messages with any specific MA. Upon receiving the status request message, the MA responds with a status answer message that contains the requested information. Figure 10 shows how the SM monitors a specific MA.

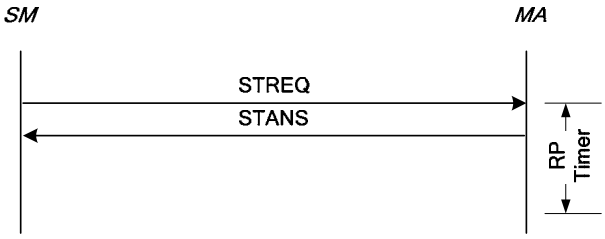


Figure 10 – Tree monitoring – Status report

삭제됨: Admission
삭제됨: /
삭제됨: policy
삭제됨: A
삭제됨: policy
삭제됨: /
삭제됨: s
삭제됨: /
삭제됨:
삭제됨: /MN
삭제됨: /
삭제됨: /
삭제됨: /
삭제됨: If the
삭제됨: requests session subscription,
삭제됨: If the MN requests session subscription,
삭제됨: /
삭제됨: leave request (
삭제됨:)
삭제됨: its
삭제됨: /
삭제됨: promptly
삭제됨: leaves the session with
삭제됨: ying
삭제됨: the
삭제됨: the
SM
LE
LE
삭제됨:

SM can also collect status information of an entire or a part of a session. In this case, the SM sends a status collect request message to the top MA of the part. Upon receiving the status collect request message, the MA should send a status answer back to the SM with appropriate information on the MA and its children. When the session size is large, the use of this mechanism for the entire session may cause overloading the network and system resources. To limit the scope of the monitoring, the status collect message should contain an option for the depth.

7.1.4 Session termination

The SM's ongoing session may terminate due to one of the following two reasons:

a) Administrative request;

b) SMA's leave.

Figure 11 shows the SM's session termination procedure.

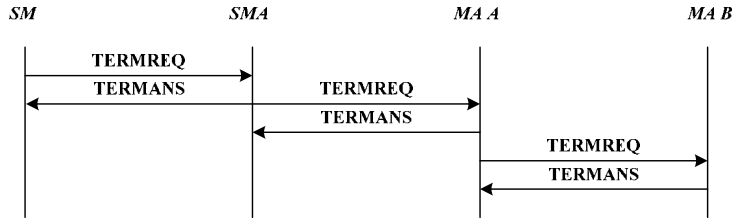


Figure 11 – Session termination issued by SM

Because a MMC-3 session can continue only when the SMA is alive, the SMA must notify the SM when it leaves. Having been notified SMA's leave, the SM should terminate the session promptly.

7.2 Multicast agent's operation

7.2.1 Session subscription

Subscription is the first stage for an MA to be enrolled in a MMC-3 session. Each MA must subscribe to the session by sending a subscription request (SUBSREQ) message to the SM.

7.2.1.1 Subscription of MCS

Note that the MCS must have finished its subscription before the other MAs and it should act as a root node in the tree hierarchy. At this stage, each MCS needs to know details of the session profile, such as the address of the session, the name of the session, and etc. In other words, subscription of MCS means not only subscription but also initiation of the session.

Figure 12 shows the procedure of session subscription procedure of MCS.

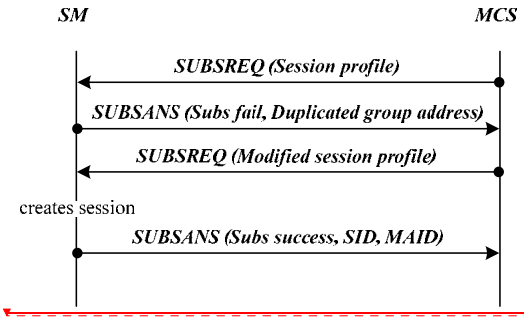


Figure 12 – MCS's subscription

In Figure 12, MCS sends a SUBSREQ message including session profile. Then SM decides whether to allow MCS to initiate the session based on received session profile. Since the group address which MCS requested to use for the

서식 있음: 제목 4

삭제됨: SMA

삭제됨: MA

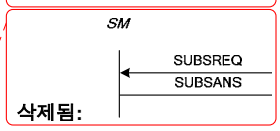
삭제됨: SM

삭제됨: and

삭제됨: policy

삭제됨: MMC-3

삭제됨: After SMA's successful subscription, MMC-3 session can be initiated.



삭제됨: SMA's

서식 있음: 표준

session is already in used, SM sends answer message with appropriate error code. Then MCS requests initiation and subscription again with different group address. Upon receiving modified session profile, SM decides to create the requested session and creates the session. After subscription of MCS to the created session, SM sends a SUBANS message including session related information e.g. SID of session and MAID of MCS.

7.2.1.2 Subscription of MA

Figure 13 shows the procedure of an MA subscription (for MA A and MA B). To subscribe a certain session, each MA sends a SUBSREQ message to the SM. Upon receiving the SUBSREQ message from MAs, the SM decides whether to accept the subscription request. If the request is accepted, the SM responds with a SUBSANS message including bootstrapping information such as a neighbor list, MAID. Otherwise, it responds with a SUBSANS message with appropriate error reason code.

After receiving a successful SUBSANS message from SM, both MAs can complete the subscription phase.

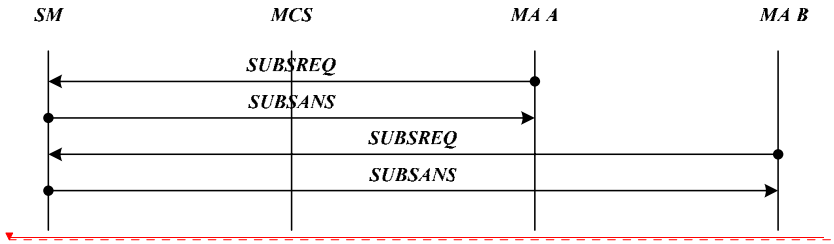


Figure 13 – MA's subscription

7.2.2 Neighbor discovery

Since all MAs are logically interconnected, it would be difficult for an MA to know the entire physical network condition. However, by using neighbor discovery procedures, each MA can explore the other MAs in the MMC-3 network and measure the distance between itself and the other MAs. The neighbor discovery mechanism consists of two steps. One is used in the multicast area, such as subnet LAN, and the other is used in the unicast area such as WAN.

7.2.2.1 Neighbor discovery in the local multicast area

This capability enables an MA to find the neighboring MAs inside the local multicast area. Multicast delivery is much efficient for the group service than unicast delivery. Therefore, the neighboring MA in the same local multicast area should be designed to be much closer than the neighboring MA in other network. The network distance in MMC-3 depends on the delay jitter, the hop count, the bandwidth, or etc.

Normally, an MA in same network is closer to a certain MA than other MAs. Each MA looks for a candidate PMA in its local network by multicasting a head multicast agent solicit (HSOLICIT) to a specific pre-assigned address (aka, multicast) at the beginning. If there is no answer, the MA becomes HMA, which is a representative of MAs in the multicast network.

7.2.2.1.1 HMA solicitation and announcement

At first, MAs must find the HMA of the multicast area. The HMA solicitation and announcement function enables MAs to find the HMA. One MA in the multicast area must be elected as the HMA taking charge of relaying the multicast data to the local multicast area. Although the HMA selection criteria are not defined in this document, factors such as session subscription time, distance from the MCS, or precedence of MAID can be used as parameters for choosing HMA. The HMA receives control and data packets from its PMA.

Figure 14 shows the procedure in which MA C finds other MAs in local multicast area. MA C needs to find HMA in local multicast area. MA C multicasts an HSOLICIT message using a specific group address to query the HMA. When the HMA receives an HSOLICIT message, it issues an HANNOUNCE message including system information of itself, such as local IP of HMA or HMA lifetime. Then MA C stops neighbor discovery and starts listening to prescribed multicast address to receive session service.

서식 있음: 제목 4
삭제됨: n
삭제됨: MMC-3
삭제됨: the
삭제됨: and
삭제됨: n
삭제됨: NL
삭제됨: the
삭제됨: (MA A and MA B)
SM
삭제됨:
삭제됨: M
삭제됨: It is desirable to assign the nearest node to its PMA.
삭제됨: and
삭제됨: the
삭제됨: broadcast
삭제됨: the
삭제됨: the
서식 있음: 제목 5
삭제됨: Once an
삭제됨: becomes a HMA, the HMA announces its existence to the multicast network by sending periodic HANNOUNCE messages
삭제됨: The HMA sends a HANNOUNCE promptly on receiving HSOLICIT from the multicast area.
삭제됨: Upon receiving the HANNOUNCE from the HMA, each MA considers that a HMA already exists in the same network and then assumes the HMA as its primary PMA candidate. Figure 13 shows the HMA selection procedure.

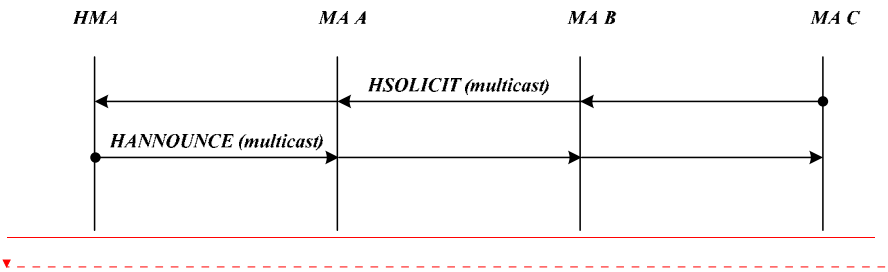
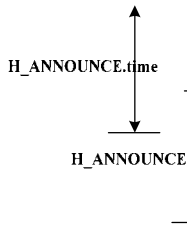


Figure 14 – HMA Solicitation and announcement

삭제됨:



삭제됨: its

Since evesdropping can occur within local network, authentication mechanism is needed. Thus both HSOLICIT and HANNOUNCE messages have to include authentication information. How to authenticate each other is not specified in this recommendation.

서식 있음: 제목 5

7.2.2.1.2 New HMA election

The new HMA election function elects an HMA in a local multicast area. The new HMA election can occur when there is no HMA in the local network. Initially, the local multicast area does not have any HMA. When MAs subscribe a certatin service session, MAs should go through the new HMA election procedure.

Each MA sends HSOLICIT message to its local network using multicast. If there is no HANNOUNCE message for a certain time which is T_HSOLICIT * N_HSOLICIT, the MA decides to be an HMA and multicasts an HANNOUNCE message. When there are multiple MAs in the local network, all MAs should have different HSOLICIT timer value (T_HSOLICIT) to prevent flooding of HSOLICIT message; the MA which has shortest T_HSOLICIT value becomes an HMA. The HMA can find out that it is the only node in the multicast area if it does not receive HSOLICIT message for HSOLICIT waiting time (W_HSOLICIT) and then it may cease to relay service data into its local network.

Another case in which the MAs must go through new HMA election is when an HMA leaves the subscribed session. The rest of MAs must compete in the new HMA election. Figure 15 shows how MA A becomes a new HMA in a local multicast area. As mensioned above, each MA has its own HSOLICIT timer value (T_HSOLICIT) and the timer value would be common T_HSOLICIT plus a criteria factor. The criteria factor can be derived from various factors such as the distance from MCS, IP address, etc.

서식 있음: Figure

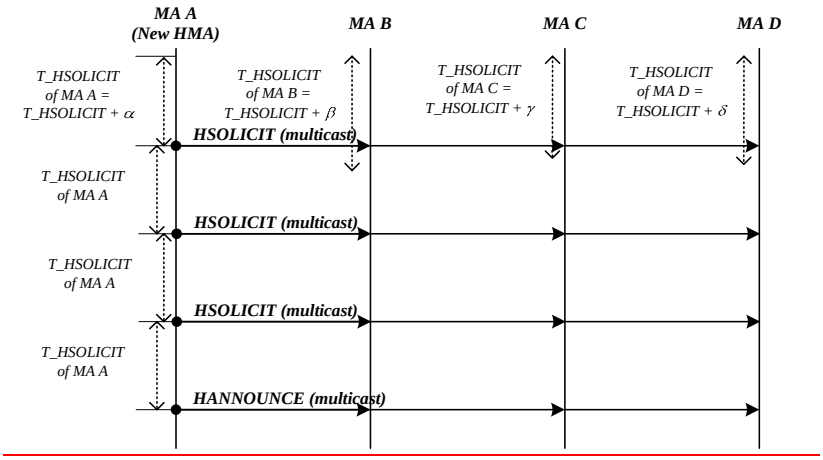


Figure 15 – Procedure of HMA election

Four MAs are competing to be the HMA in the local multicast area. Four MAs start its own HSOLICIT timer. The MA with the shortest HSOLICIT timer sends an HSOLICIT message first. MA A has the shortest timer and multicasts an HSOLICIT message to the local multicast area. Other MAs (i.e., MA B, MA C, and MA D) receive the HSOLICIT message and suppress sending the HSOLICIT message and then restart the HSOLICIT timer, again. If the MA which already sent an HSOLICIT message receives the HSOLICIT message sent from other MA, the MA does not suppress nor restart its HSOLICIT timer. MA A waits an HANNOUNCE message during its HSOLICIT time. Since there is no answer, MA A sends an HSOLICIT message several times. The number of HSOLICIT message sending is defined as a

N_HSOLICIT with a default value being 3. MA A will know that there is no HMA in the multicast area and decides to be the HMA by multicasting an HANNOUNCE message to the local multicast area.

It can occur that more than two MAs multicast HANNOUNCE message to the local network because each of them considers itself as a HMA. In such case, only one of them has to be selected as the HMA. Figure 17 shows procedure for HMA contention. It is assumed that both MA A and MA C have same HSOLICIT timer and also have same session subscription time, and also assumed that MAID of MA A is smaller than that of MA C. Since HSOLICIT timer value of both MAs is equal, both MAs issue HANNOUNCE message after sending HSOLICIT message N_HSOLICIT times. Upon receiving the HANNOUNCE message from each other, each MA decides which MA should be selected as an HMA according to HMA selection rule. As a result, MA A is selected as an HMA. Although the rule for selecting HMA can be varied, this recommendation uses following HMA selection rules:

a) If they have different session subscription time, earlier session subscriber will be selected as an HMA.

b) Otherwise, the MA which has lower MAID will be elected as an HMA.

서식 있음: enumlev1

서식 있음: 제목 5

7.2.2.1.3 HMA continuity

Once an MA is elected as an HMA, it must continue its role until it leaves the session or the session is terminated. Figure 16 shows how the HMA can continue its role as an HMA. As mentioned above, each MA has its own HSOLICIT timer (T_HSOLICIT). Each MA starts its HSOLICIT timer and the MA with the shortest HSOLICIT timer starts sends an HSOLICIT message at the expiration of its timer. Other MAs which have longer HSOLICIT timer suppress HSOLICIT message sending at the reception of HSOLICIT message from other MA. The HMA responds by multicasting HANNOUNCE message. This procedure continues periodically.

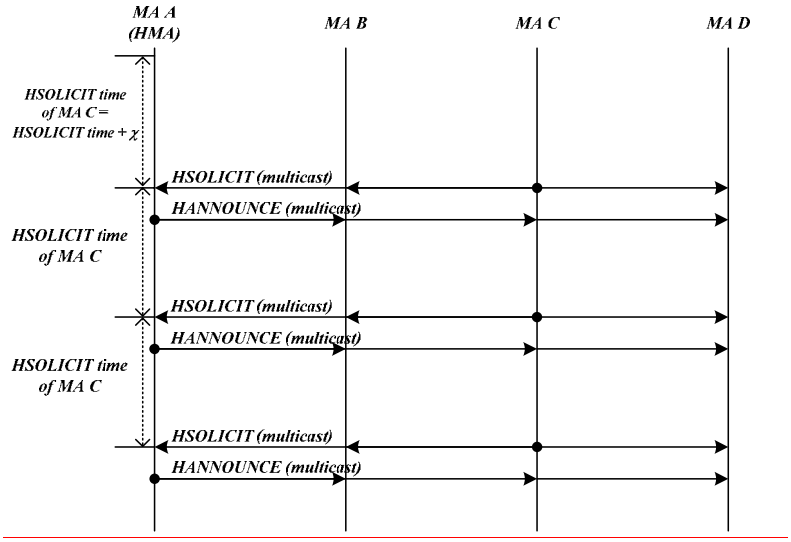


Figure 16 – Periodic HMA announcement caused by HMA solicitation

If the HMA does not receive HSOLICIT message within certain period of time (HSOLICIT waiting time), it recognizes that there is no other MA in the local multicast area. The HSOLICIT wait time is defined as W_HSOLICIT. Recommended HSOLICIT waiting time is T_HSOLICIT * N_SOLICIT. The HMA does not multicast received data anymore.

NOTE – The HMA starts again multicasting HANNOUNCE message and received data when a new MA in the same local network subscribes the session and performs HMA solicitation.

서식 있음: Note 1

It can be happen that a new MA may have smaller HSOLICT timer than the current HMA's timer. In that case, the role of HMA is not changed. The new MA may become the next HMA if the current HMA leaves the session.

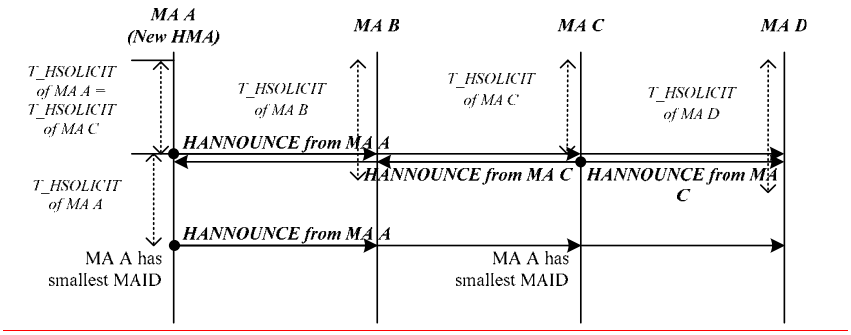


Figure 17 – Procedure for HMA contention

7.2.2.2 Neighbor discovery in the unicast area

Neighbor discovery in the unicast area enables an MA to find other MAs outside the local multicast network.

NOTE – The HMA must perform same function to the area as the EdgeMA in the unicast area along with the HMA function for the multicast area.

In MMC-3, an MA is required to have information about other MAs participating in the same session because the MA needs to make logical connection with other participating nodes. This also pertains to the HMA in the multicast area which also needs to have logical unicast connection with other participating nodes outside of the local multicast area.

Figure 18 shows the procedure of neighbor discovery in the unicast area. MA D sends a PPROBREQ message to all MAs listed in the neighbor list given by the SM in subscription phase. MA D will get a PPROBANS message from each MA. MA D needs to make decision based on the received PPROBANS messages and it decides to select MA B based on its parent decision rules.

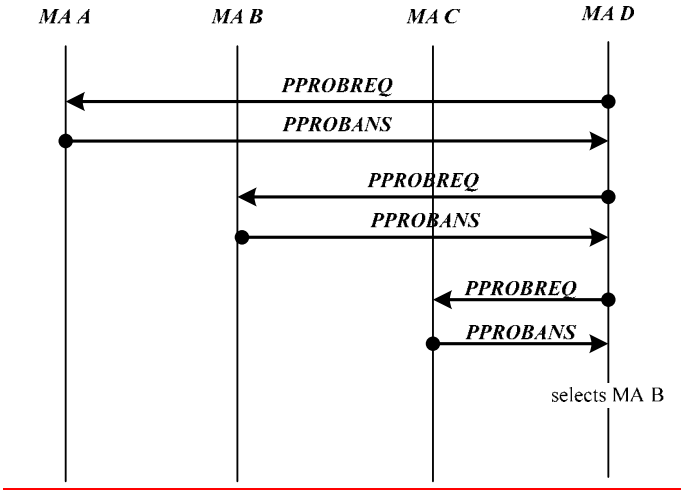


Figure 18 – Sequence of neighbor discovery in the unicast area

MAs can gradually learn the MMC-3 tree topology by exchanging the neighbor list of each MA by conducting neighbor discovery. Because of the finite system resource of each MA, the maximum number of neighbor list to be exchanged should be bounded.

To prevent each MA suffered from PPROBREQ message implosion, the maximum number of PPROBREQ messages for a certain period should be limited as N_MAX_PROBE.

Moreover system information should be included within a PPROBANS message to prevent performance degradation. By using system information within the PPROBANS message, placing low-capability node in high position on the tree hierarchy that may cause entire performance degradation can be prevented.

서식 있음: 가운데

삭제됨: U

서식 있음: Note 1

서식 있음: Figure

삭제됨: Each MA should start neighbour discovery procedure based on the initial bootstrapping information given by the SM. As shown in Figure 17, each

삭제됨: tree information

삭제됨:
The basic neighbor discovery mechanism is as follows: first, by using the PPROBREQ and PPROBANS, each MA can exchange a certain number of NLs at every interval (PPROBE.time).

삭제됨: NLs

서식 있음: 표준

삭제됨:

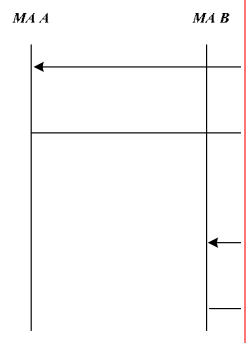


Figure 17 – Protocol sequence of neighbor discovery

7.2.2.3 Tree join

Tree join procedure enables each MA (both MA and MMA) to choose PMA inside the subscribed MMC-3 session. Figure 19 shows how an MA selects its PMA based on the neighbor list given by the SM. The joining MA (MA E) conducts neighbour discovery first. MA E sends a PPROBREQ message to one or more MAs listed in the neighbor list (MA A, C, and D) and it awaits a PPROBANS message. Upon receiving a PPROBANS message, the MA E can select the nearest MA. In Figure 19, the joining MA E considers that the MA D is the best and then chooses the MA D as its PMA. After a PMA is selected, the joining MA E sends to the MA D a RELREQ message which contains a proposed data profile.

If the RELREQ message is acceptable, MA D responds with a successful RELANS message which includes the negotiated data profile to be used. Otherwise, MA D returns a reason code of the rejection.

Upon receiving a successful RELANS message, data channel between MA D and MA E is established according to the negotiated data profile. Otherwise, MA E should try the second optimal PMA candidate for join.

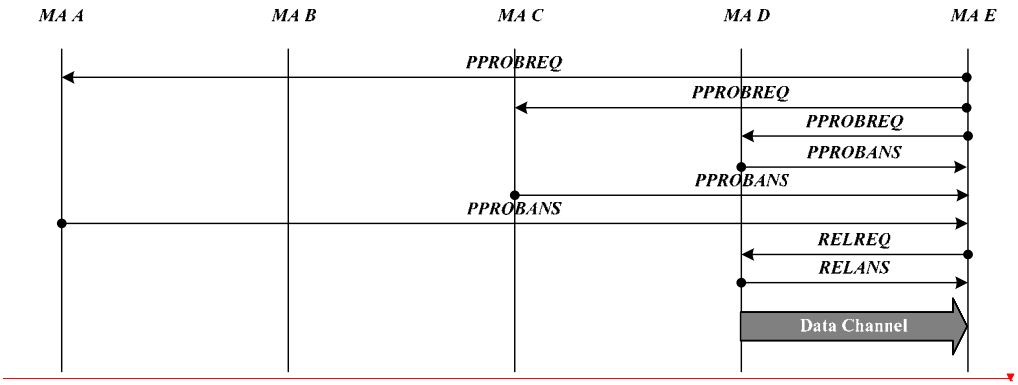
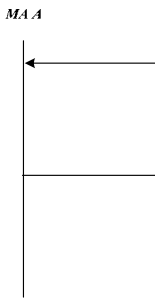


Figure 19 – Protocol sequence of successful tree join

If no MA wants to relay data to the joining MA, the joining MA can retry tree join procedure after a certain period. The retrial time is beyond the scope of this Specification. Figure 20 shows when all the MAs listed in the neighbor list given by the SM rejected MA D's relay request. However MA D already learned about the existence of MA A during previous exchanges of PPROBREQ and PPROBANS messages, it can restart the joining procedure from MA A.

삭제됨: a
삭제됨: Figure 39
삭제됨: NL
삭제됨: nodes
삭제됨: NL
삭제됨: successful
삭제됨:
삭제됨:
삭제됨: Figure 39
삭제됨: (node E)
삭제됨: (node E)
삭제됨: will
삭제됨: ,
삭제됨: the
삭제됨: ,
삭제됨: the
삭제됨: the
삭제됨: the


삭제됨:
삭제됨: can be set by the user, though this issue
삭제됨: NL
삭제됨: node
삭제됨: E's
삭제됨: E
삭제됨: B
삭제됨: B

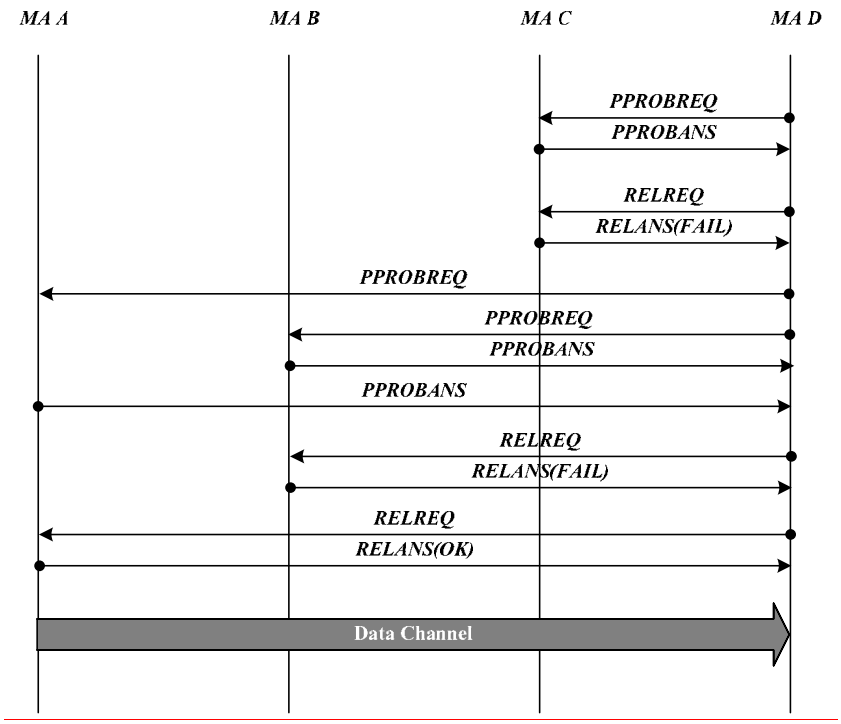


Figure 20 – Sequence of unsuccessful tree join and retrial

7.2.3 Leave

An MA may leave a session or a PMA during the session lifetime. To make a MMC-3 tree robust, each MA should notify its departure to the PMA and CMA(s). Upon receiving this notification, the PMA and each CMA should follow the appropriate procedure. In addition, departing MA should notify its departure to the SM. Then SM conducts appropriate operation.

The MMC-3 considers four types of departure:

a) An MA leaves the session at the request of a service user.

b) An MA leaves its PMA to switch parent.

c) The expulsion of an MA from its PMA or SM.

d) The departure of an MCS from its session.

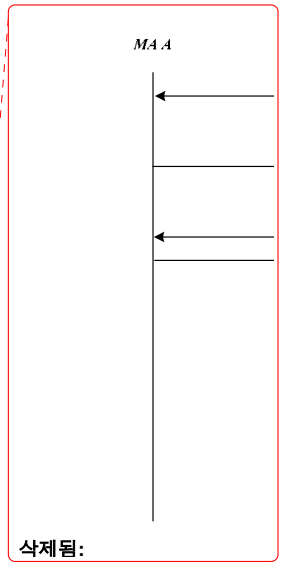
The detailed operations for the cases are described in the following subclauses.

7.2.3.1 When MA leaves a session

MAs may leave a session at any time during the session's lifetime. Before leaving, each MA must notify the PMA and CMA(s) of its departure. The PMA deletes the node from its CMA list for the session and reserves a space for a new CMA. In addition to notification to both PMA and CMA(s), leaving MA should also notify the SM of its departure. Then the SM checks whether requesting MA is subscribed to the session which the MA requests to leave. If the MA is subscribed for the session, SM modifies the information related to the departing MA. Otherwise SM ignores the request.

7.2.3.1.1 MA leave in the unicast area

To leave a session, an MA sends a LEAVREQ message to its CMAs. Each CMA who receives the LEAVREQ message should promptly start to connect to an alternative PMA by sending a RELREQ message to the PMA candidate. If successful, each CMA sends its old PMA a LEAVANS message. When the departing MA receives LEAVANS messages from all of its CMAs, it sends a LEAVREQ message to its PMA. Then PMA of the departing MA responds with LEAVANS message. Before departing the session, the MA should send a LEAVREQ message to the SM to notify its departure. This notification lets the SM manage each MA properly.



삭제됨: .
삭제됨: The first one refers to
서식 있음: enumlev1
삭제됨: an
삭제됨: that
삭제됨: The second one refers to a
삭제됨: that
삭제됨: s
삭제됨: The third one refers to t
삭제됨: The final one refers to t
삭제됨: SMA
삭제됨: a
삭제됨: a
삭제됨: n
삭제됨: a)
서식 있음: 제목 5, 다음 단락과의 사이에 페이지 나누기, 현재 단락을 나눔

NOTE – Data delivery from leaving MA to each CMA should be continued until each CMA connects with new PMA. It means that data delivery from PMA to leaving MA also should be continued until connection between two MAs is disconnected.

Figure 21 shows how an MA leaves a session in unicast area. In this scenario, the procedures of leaving for a non-HMA and the HMA are the same, except the HMA follows the HLEAVE message exchanging sequence.

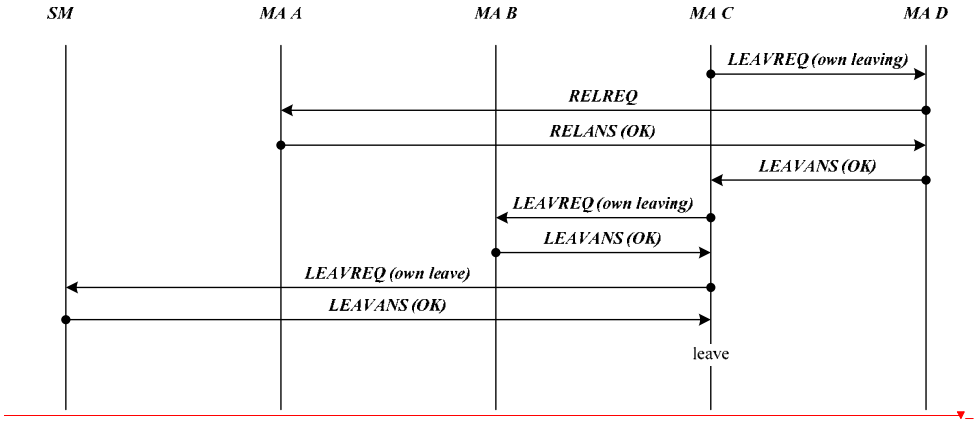


Figure 21 – MA leave in the unicast area

7.2.3.1.2 MA leave in the multicast area

There are two cases of MA's leaving within a multicast area. The first case is of HMA's leaving and the other is of MA's leaving. Whenever the HMA of a multicast area wants to leave a session, it should notify its departure to the CMAs inside the local network as well as to the CMAs and the PMA in the other network.

Figure 22 shows how MA B, which acts as HMA, leaves a session. The HMA (MA B) sends a LEAVREQ message to its direct CMA (MA E) in the other network. Upon receiving the LEAVREQ message, MA E starts to switch parent and responds to MA B with a LEAVANS message. MA B multicasts a HLEAVE message into the local network. The HLEAVE message is used to announce the departure of the HMA.

Upon receiving the HLEAVE message from HMA, both MA C and MA D from Figure 22 wait for a certain back-off time before multicasting the HANNOUNCE message. MA C sends the HANNOUNCE message for the first time and becomes a new HMA. This step occurs because the MA C has a shorter back-off time than that of MA D as described in clause 7.2.2.1. Because MA B is a point which is connected to outside network, MA C should undertake the role of the MA B by connecting to the PMA in the outside network. Figure 22 shows how MA C selects for its parent MA A, which is the PMA of MA B. Before MA B leaves the session, it should send a LEAVREQ message to the SM. Then the SM modifies the information related to MA B.

NOTE – departing HMA should stop multicasting the session data and send a LEAVREQ message when it receives a HANNOUNCE message from the new HMA because the data sent by old HMA can collide with the data sent by new HMA.

서식 있음: Note 1
MA A
삭제됨:
삭제됨: b)
서식 있음 … [2]
삭제됨: unicast
삭제됨: the
삭제됨: C
삭제됨: where the m … [3]
삭제됨: C
삭제됨: F
삭제됨: unicast
삭제됨: F
삭제됨: s
삭제됨: C
삭제됨: as well as
삭제됨: with an empt … [4]
삭제됨: D
삭제됨: E
삭제됨: The
삭제됨: D
삭제됨: D
삭제됨: any other
삭제됨: the leaving
삭제됨: C
삭제됨: unicast
삭제됨: the
삭제됨: D
삭제됨: C
삭제됨: unicast
삭제됨: the
삭제됨: D
삭제됨: the
삭제됨: B
삭제됨: the
삭제됨: C
서식 있음: Note 1

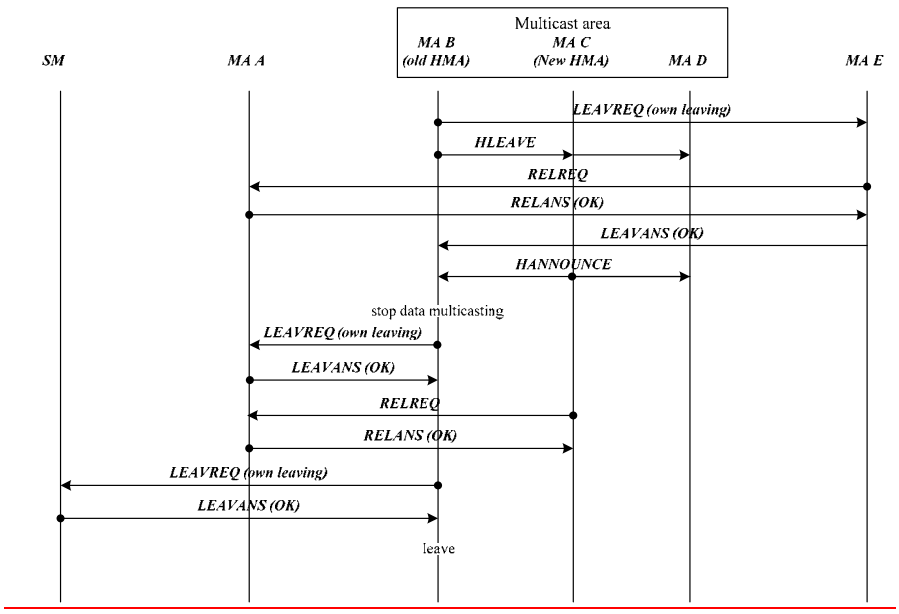


Figure 22 – HMA leave in the multicast area

Whenever any non-HMA of a multicast area wants to leave a session, it silently leaves the session. MA C or MA D from Figure 22 does not need to notify other MAs of its departure. Before, however, it leaves the session, each MA should send a LEAVREQ message to the SM. Then the SM modifies the information related to the departing MA.

7.2.3.2 When MA leaves its PMA – for parent switching

An MA that wants to switch its PMA can leave its current PMA. As described in clause 7.2.4.4, MAs can change its PMA only when MAs receives HB message from its PMA. When an MA changes its PMA, the MA does not need to send a LEAVREQ message to its CMAs. The CMAs do not need to know about the departure of their PMA as long as they successfully receive data. To switch PMA, the MA sends a pseudo HB message to its CMAs to prevent partition recovery operation described in clause 7.2.4.3.2 and then it sends a RELREQ message to the other PMA candidate. In Figure 23, since MA C changes its PMA from MA A to MA B, ROOTPATH of MA C should be also changed. Thus MA B gives its ROOTPATH to MA C when it sends a RELANS message. An old PMA (MA A) that receives a LEAVREQ message deletes the leaving MA from its CMA list but keeps the information of the departing MA in its neighbor list because the leaving MA is still alive in the session.

NOTE 1 – an MA can switch parent only when it receives a HB message to keep tree stable. The HB mechanism is described in clause 7.2.4.1.and parent switching is described in clause 7.2.4.4.

NOTE 2 – HB* in Figure 23 means pseudo-HB message.

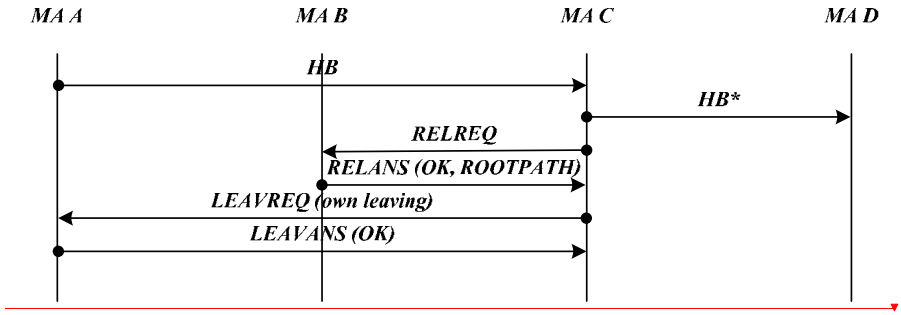
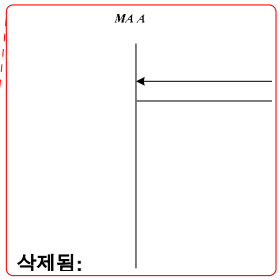
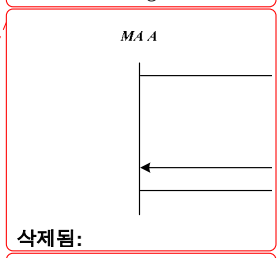


Figure 23 – MA's departure for parent switching

삭제됨:


삭제됨: The
삭제됨: D
삭제됨: E
삭제됨: **from**
삭제됨: may
삭제됨: In this case,
삭제됨: with the reason code set to PS (parent switching)
삭제됨: NL
삭제됨: Figure 22 shows how an MA switches its parents.
서식 있음: Note 1
삭제됨: Note that
삭제됨: s
삭제됨: unchanged

삭제됨:


삭제됨: **leaving**

7.2.3.3 When MA is kicked out

MMC-3 has a mechanism for discarding a certain MA. For example, when the SM wants to discard a specific MA; and when an MA expels its CMA after it was aware that it cannot support more CMAs.

7.2.3.3.1 Expulsion of an MA by its PMA

A PMA can expel one of its CMAs when the PMA suffers from depleted system resources and can no longer feed its CMA, or when the PMA finds that one of its CMAs has depleted its system resources. Then the expelled MA should find another PMA candidate, which would allow for a new CMA.

Figure 24 shows an example of a message flow. First, a PMA, namely the MA B, sends a LEAVREQ message with a reason PMA KO to expel MA C. The MA C searches other PMA and sends a RELREQ message. After switching parent to MA A, MA C transmits a LEAVANS message to its old PMA, MA B.

NOTE – although if MA C in Figure 24 has CMAs, CMAs of MA C does not need to know about expulsion of MA C as long as they successfully receive data like parent switching.

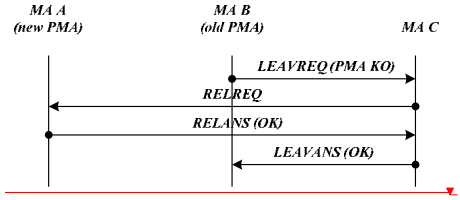


Figure 24 – When MA is kicked out by its PMA

7.2.3.3.2 Expulsion of an MA by the SM

The SM can discard any MA by sending a LEAVREQ message with a reason SM kicked-out (SM KO). Upon receiving a LEAVREQ message from the SM, an MA must leave the session promptly. After the expulsion, the SM should update the information related to the expelled MA e.g. session member list, MA list, or etc.

In the message flow shown in Figure 25, the SM tells MA B to leave the session by sending a LEAVREQ message with a reason SM KO. MA B must leave the session but, before leaving, MA B must notify its PMA and CMAs of its expulsion by sending a LEAVREQ message with a reason I am KO. Then the PMA, MA A, removes the information related to MA B and the CMA, MA C, should find new PMA to receive the session data continuously.

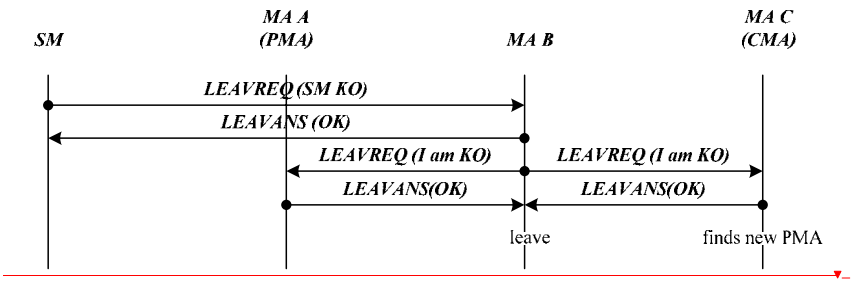


Figure 25 – When MA is kicked out by SM

7.2.3.4 When MCS leaves the session

Because a MMC-3 session cannot exist without a MCS, a MCS never leaves a session before the session is terminated. However the MCS may leave the session because of administrative purpose, the session should be terminated. The MCS may also leave the session abruptly because of failure, and then SM should terminate the session. The termination caused by MCS failure is described in clause 7.2.4.3.

Figure 26 shows the departure procedure of an MCS from a session. The MCS sends a LEAVREQ message to the SM. Upon receiving the LEAVREQ message from the MCS, the SM checks whether mapping between MCS and the session requested to terminate is correct and removes the session information and then replies with LEAVANS message. Upon receiving the LEAVANS message from the SM, the MCS sends a LEAVREQ message with reason indicating that MCS leaves to its direct CMAs. The LEAVREQ message should be relayed downward promptly to make MMC-3 session

삭제됨: s
삭제됨: a network manager wants
삭제됨: a
삭제됨: a)
서식 있음: 제목 5
삭제됨: the
삭제됨: An
삭제됨: C
삭제됨: D
삭제됨: D
삭제됨: s
삭제됨: relay request
삭제됨: s
삭제됨: D
서식 있음: Note 1
MA A
삭제됨:
삭제됨: b)
서식 있음: 제목 5
삭제됨: its
SM
LEAVI
LEAV
삭제됨:
삭제됨: SMA
삭제됨: n
삭제됨: n
삭제됨: SMA
삭제됨: n
삭제됨: SMA
삭제됨: In this case, when
삭제됨: SMA
삭제됨: s
삭제됨: SMA
삭제됨: SMA
삭제됨: SMA
삭제됨: SMA
삭제됨: SMA
삭제됨: with reason SMA leave

terminated. If the MCS or the MA which sends LEAVREQ message to its CMAs does not receive the LEAVANS message from its all CMAs, it retransmits LEAVREQ message to unanswered CMAs.

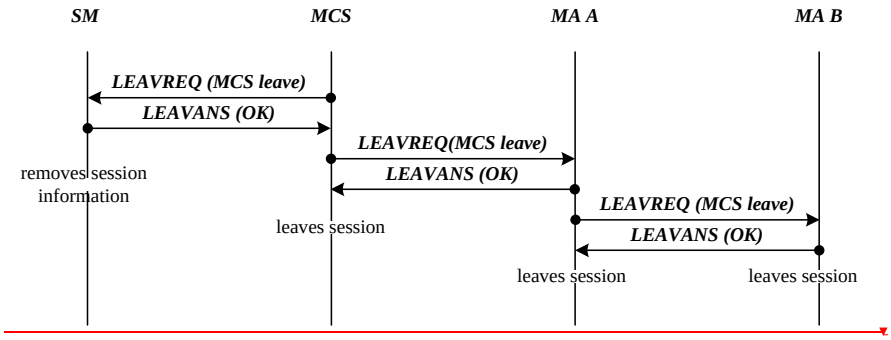


Figure 26 – Procedure for MCS's departure

7.2.4 Maintenance

7.2.4.1 Heartbeat

Heatbeat message is used to check the MMC-3 tree status. The heartbeat, which gives unified synchronizing information, helps each MA detect whether the current MMC-3 tree is robust. Theheartbeat (HB) message contains information on the data delivery path, namly, ROOTPATH. The ROOTPATH includes a relayed data path of the tree hierarchy.

Figure 27 shows the MMC-3 heartbeat procedure. SM initiates the heartbeat procedure by sending HB message to the MCS. MCS propogates the HB message to its child MAs along the MMC-3 tree. Each MA appends its local informations to the HB message and forwards the message to its child MA. The appended informations include MAID, per-hop network distance and system information such as in-and-out bandwidth, number of CMA capacity, etc. The leaf MA which can be MA or MMA will receive HB message with full ROOTHPATH, which contains informations of all the MAs visited along the MMC-3 tree.

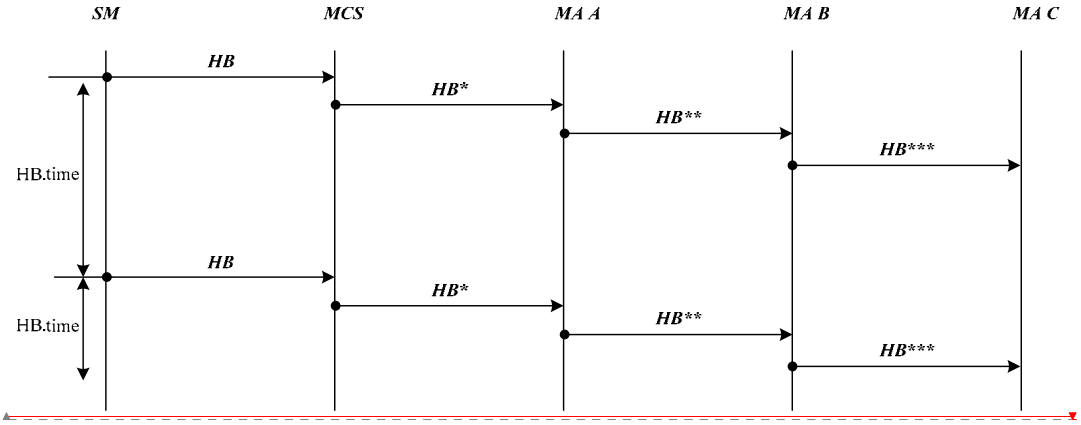
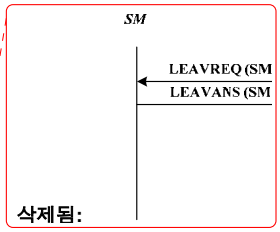
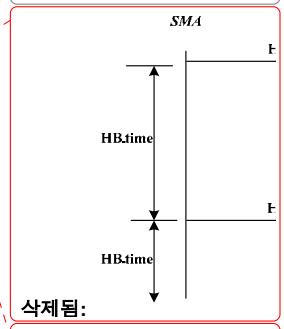


Figure 27 – Heartbeat message flow

The HB message contains system informations of the precedence MAs. MA can decide to change PMA with the receiption and analysis of the HB message. In that case, MA needs to send a pseudo-HB message to its CMAs. This will prevent the whole MAs in the MMC-3 tree from performing simultaneous parent switching. After parent switching, MA modifies HB message based on the new ROOTPATH in the RELANS message and sends modified HB message to its CMAs.



삭제됨: SMA's
삭제됨: leaving
삭제됨: The purpose of the heartbeat is to keep the constructed MMC-3 tree robust.
삭제됨: to the session
삭제됨: session is currently alive
삭제됨: It also contains useful
삭제됨: ed
삭제됨: which follows
삭제됨: In this procedure, the SMA
삭제됨: sends
삭제됨: , along the tree,
삭제됨: descendants;
삭제됨: e
삭제됨: descendant then
삭제됨: the hop
삭제됨: , which may
삭제됨: affordable
삭제됨: , to the HB and forwards the modified HB to its descendants. Finally the ROOTPATH
변경된 필드 코드



서식 있음: 표준

7.2.4.2 Status Report

There are two methods for SM to retrieve status report from MMC-3 tree. The first method, which is shown in Figure 28, retrieves status information of specific MA through STREQ and STANS message. The second method, as shown in Figure 29, retrieves status informations of all node in a part of the MMC-3 tree.

Figure 28 shows method SM retrieving information of specific MA, namely MA B. In this procedure, the SM sends an STREQ message to MA B and requests for specific information. In response, MA B sends SM a STANS message with the requested information.

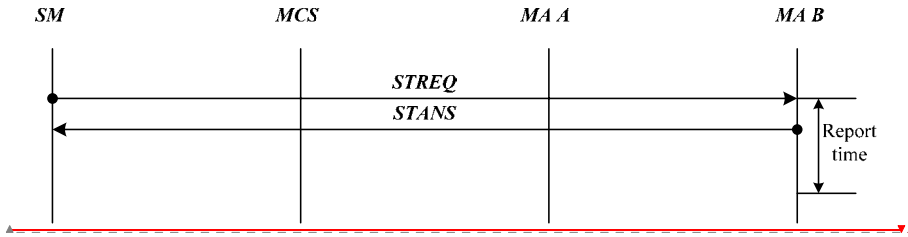


Figure 28 – MA monitoring by status report

Figure 29 shows how the SM requests for status informations from a MMC-3 sub-tree with more than two-depth to a specific MA (MCS and MA A each) to collect status information for the sub-tree. The MA sends STREQ message to the root of the sub-tree with tree depth. The expected response time from the sub-tree, which is, Session Tree Time, may be varied according to depth of the MMC-3 sub-tree.

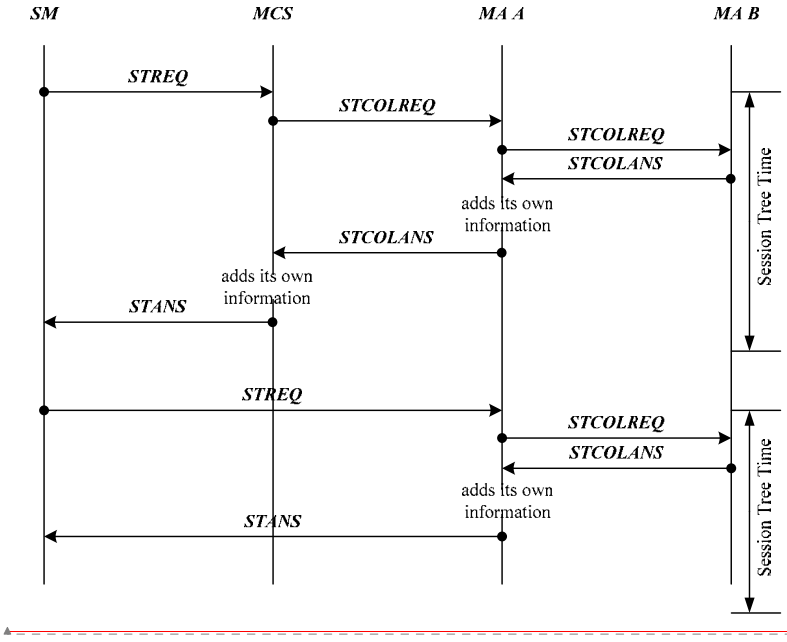
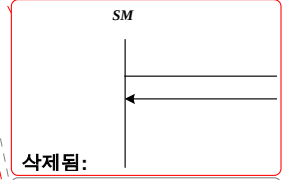


Figure 29 – Tree monitoring by collecting status report

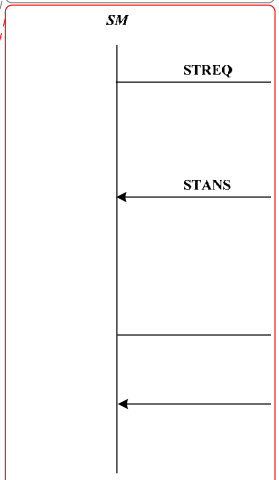
7.2.4.3 Fault detection & recovery

This procedure is performed by each MA when each MA detects network faults and recovers from the problems to make the MMC-3 tree robust. Network faults such as looping or partitioning are often caused by an MA's frequent and careless movements. To detect and recover such network faults, MMC-3 provides the following fault detection and recovery mechanisms.

삭제됨: **Monitoring**
삭제됨: MMC-3 has
삭제됨: types of monitoring mechanisms
삭제됨: one
삭제됨: monitors a
삭제됨: other one
삭제됨: which is
삭제됨: monitors
삭제됨: through a specific MA
삭제됨: how an SM monitors a
삭제됨: one or more
삭제됨: types of status
삭제됨: from MA B
삭제됨: the



삭제됨:
변경된 필드 코드
삭제됨: **Tree**
삭제됨: queries the scoped area of a tree. That is, the SM asks for merged information on the scoped area of a tree by sending an STREQ
삭제됨: SMA
삭제됨: scoped area
변경된 필드 코드



삭제됨:

7.2.4.3.1 Loop detection and recovery

Loop occurs when MA inadvertenly becomes CMA of its descendant. This can occur during parent switching. One such scenario can exists as follows:

a) MA A performs parent probe procedure and chooses certain MA (which is, MA B) as a candidate PMA.

b) Choosen MA B happens to be a discendent of MA A.

c) MA A receives HB message and changes PMA from current PMA to MA B.

d) Thus, Loop has occurred.

Figure 30 shows the procedure of loop detection and recovery. To prevent loop from occurring, MA A receives RELANS message from MA B. MA A checks whether it is already in the ROOTPATH list which is in the RELANS message. Since MA A is ancestor of MA B, MA A is already in the ROOTPATH list. Thus, MA A will detect loop. MA A disconnects connection between MA A and MA B by exchanging LEAVREQ/LEAVANS messages. After connection disconnection, MA A sends HB message to its CMA.

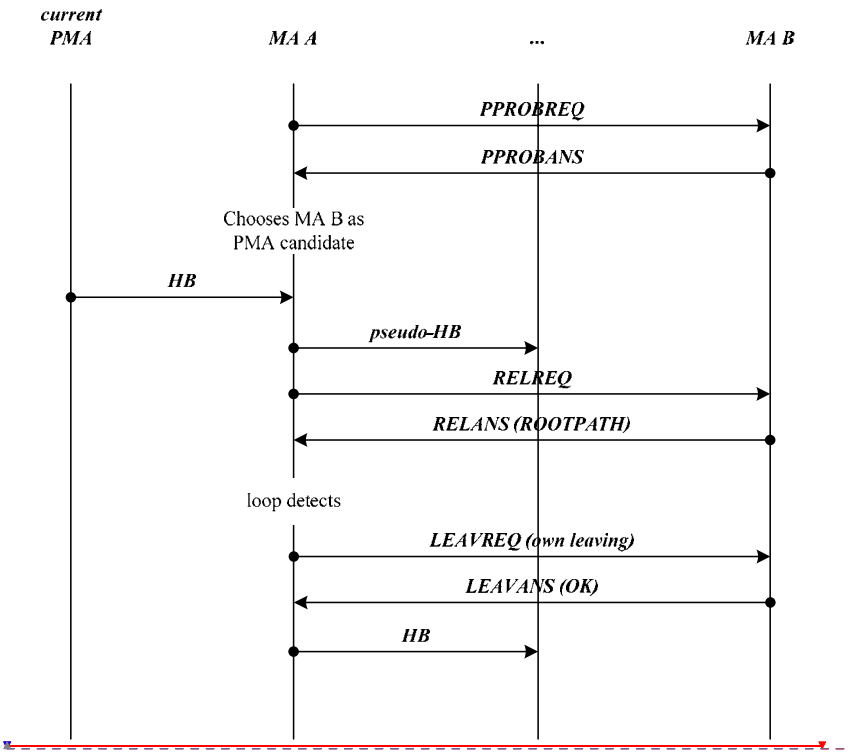


Figure 30 – Loop detection and recovery

7.2.4.3.2 Network partitioning detection and recovery

Whenever an MA fails to receive the HB message for a certain time, the MA assumes that it is partitioned from the tree. The time should be set for sufficient time to allow for a network delay. MMC-3 defines the time as HB_time × MAX_PARTITION_CNT.

A partition can occur whenever one of the partition's associates fails. The MA detects the source of the partitioning by contacting its associates; the MA then solves the problem. Figure 31 shows how MA C detects tree partitioning: that is, a tree partition is detected whenever MA C fails to receive the HB message for a certain period (HB_time × MAX_PARTITION_CNT). The failure to receive the HB message triggers the transmission of a number of PPROBREQ messages towards its associates except for MA B. In addition to the transmission of a number of PPROBREQ messages, the MA C sends CHECKREQ message to the SM to check whether MA B is alive.

NOTE – MA C may send PPROBREQ message toward MAs in the neighbor list or MAs in the ROOTPATH except for MA B.

삭제됨: a)
서식 있음: 제목 5, 없음
삭제됨: A loop can be detected by checking the ROOTPATH contained in HB. Because the ROOTPATH gives the path track from the SMA to itself, the duplicated hop in the ROOTPATH means that a loop has formed. Whenever a loop occurs, each MA performs the following loop recovery mechanism: for the scenario described in Figure 29, MA Y examines the HB; MA Y then confirms the existence of a loop whenever it receives HB_{n+3} because MA Z, which is a CMA of MA Y, is already listed in the ROOTPATH twice. To recover from the loop, MA Y sends MA Z a LEAVREQ message to disconnect.
서식 있음: enumlev1
삭제됨:
삭제됨:
삭제됨:
변경된 필드 코드
서식 있음: 제목 5
삭제됨: b)
삭제됨: _TIME
삭제됨: Z
삭제됨: Z
삭제됨: _TIME
서식 있음: Note 1, 수준 1

In Figure 31, if SM does not receive an STANS message from MA B for certain time thus SM considers that MA B has failed. In this case, SM modifies the information of MA B indicating failure.

If MA B responds to SM with STANS message, SM does nothing. In this case, network connection between MA B and MA C could be the problem.

During an MA's repairing the partition, the MA's descendants may also consider that the network has partitioned and they may start to repair the partition. As a result, an MA's fails in just one point can cause an entire tree to collapse. To prevent from this problem, an MA, which is repairing a network fault, generates a pseudo HB message to its descendants to notify that the session is temporarily partitioned and being recovered.

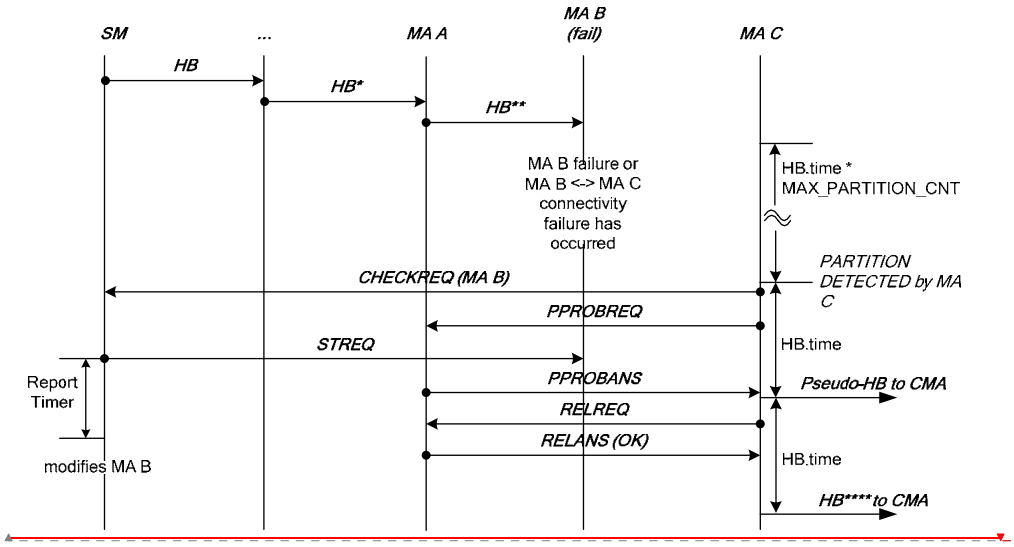


Figure 31 – Network partitioning detection and recovery

7.2.4.3.3 Failure of SM and MA

Failure of SM, MCS, or MA can occur. For robust service, MMC-3 should handle failure for each case.

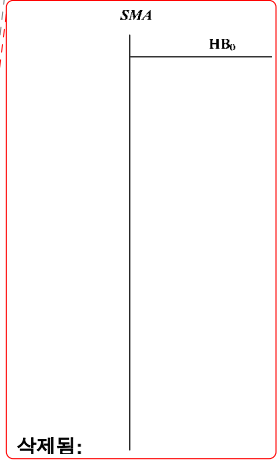
For SM failure, MCS will know the failure of SM, since MCS does not receive periodic HB message from the SM, MCS can recognize the SM failure. In that case, on going session may be continued but both session creation and subscription are impossible. If the service should be managed by the SM, MCS should send TERMREQ message to its CMA(s) to terminate the session.

In case of MA failure, the MA may be a PMA of a certain MA or be a CMA of a certain MA. An MA can recognize failure of its PMA by missing either periodic HB message or RELANS message and can also recognize failure of its CMA by missing periodic RELREQ message. Figure 32 shows an example of PMA detection and recovery procedure. Since the MA B does not receive either HB message or RELANS message during its timer, the MA B sends CHECKREQ message to SM to check whether the MA A works or not. The MA B also performs neighbor discovery simultaneously. SM sends STREQ message to the MA A but the MA A does not answer because of failure. Thus SM modifies the information related to the MA A. After neighbor discovery and tree join, the MA B receives data from MCS. If the MA A answers STREQ message with STANS message, the SM does not remove the information related to the MA A and does nothing. In that case, although MA A is not failed, MA B changes it parent.

In case of MCS failure, direct CMA(s) of MCS can recognize failure of MCS. Then the MA sends a CHECKREQ message to the SM to check whether the MCS is alive. Then SM sends STREQ message to the MCS. If MCS answers with the STANS message, SM does nothing. If, however, MCS does not answer, SM considers that the MCS is failed and sends TERMREQ message to all of MCS's direct CMAs to terminate the session. The TERMREQ message is relayed along the control tree and the session is terminated.

삭제됨: MA Z receives a PPROBANS message from MA A and MA B but no response from MA C, the current PMA of MA Z. MA Z detects that the partitioning occurs as a result of the failure of the direct PMA of MA Z; MA Z then tries to switch parents in order to recover from the partitioning.

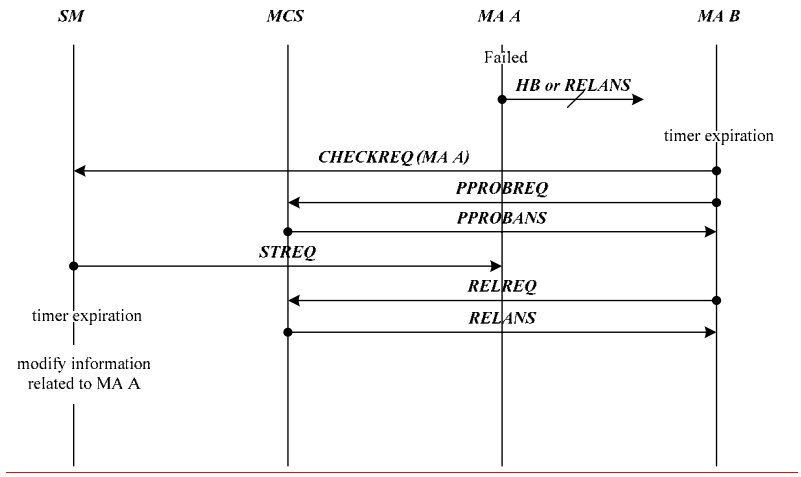
변경된 필드 코드



삭제됨:

서식 있음: 제목 5

서식 있음: 표준



서식 있음: Figure
변경된 필드 코드

Figure 32 – PMA failure detection and recovery

7.2.4.4 Tree improvement

삭제됨:

Tree improvement procedure occurs when an MA finds one or more efficient PMA candidates and tries switching to the found one. By continuing the tree improvement procedure during the session, session tree can be improved gradually.

삭제됨: MMC-3

The procedure for finding better nodes follows the neighbor discovery mechanism described in clause 7.2.2. At every turns of the neighbor discovery, each MA compares the QoS parameters of its current PMA with those of the newly discovered node. When an MA found a better MA than its current PMA, then the MA can switch its current PMA to a newly discovered MA according the parent switching procedure described in clause 7.2.3.2.

While the tree is being improved, network faults such as a loop or partition can occur from parent switching. In particular, network faults may occur in the following cases: when multiple MAs in the same branch may try to switch their PMAs at the same time and when multiple MAs along the branch may try to successively switch their PMAs.

삭제됨: easily

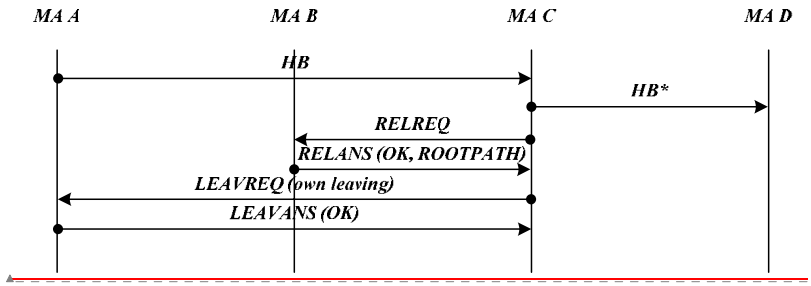
삭제됨: by

To keep a tree from these hazards, MMC-3 guarantees the atomic condition, in which each MA can switch a parent only after receiving a HB message with an unchanged ROOTPATH. In addition, a threshold value is used to prevent frequent parent switching. It means that an MA can try to change its PMA only when the cost of connection between itself and its PMA is better than the prescribed threshold value. The cost may be hop count, delay, bandwidth, or etc but a detailed description of the cost as well as the threshold value is not address in this document because it can be varied according to service policy.

Editor's note: description for the threshold value for parent switching will be added in clause 9.5.

서식 있음: 글꼴: 기울임꼴, 강조

서식 있음: 글꼴: 기울임꼴, 강조



서식 있음: Figure
변경된 필드 코드

Figure 33 – Procedure to switch parent for tree enhancement

서식 있음: Figure_No & title, 수준 1

When an MA starts parent switching, it sends pseudo-HB message to its CMA(s) to prevent its CMA(s) from conducting partition recovery. Although MAs can change it PMA only when it receives HB message, MAs do not change its PMA when it receives pseudo-HB message. Since simultaneous parent switching can cause partition or loop, an MA should not conduct parent switching when its PMA is already conducting parent switching.

7.2.5 Termination

To terminate session, the SM sends a TERMREQ message to MCS as shown in Figure 34. A MCS (or MA) that receives a TERMREQ message from the SM (or PMA) sends the TERMANS message back to the SM (or PMA) and then forwards the TERMREQ message to its CMAs until it reaches the end nodes of the tree. Finally, the session is closed gradually. Upon receiving a TERMANS message, SM and MAs remove the information related to the closed session.

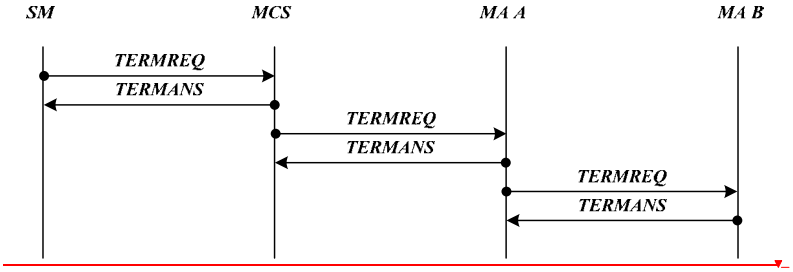


Figure 34 – Session Termination issued by SM

7.3 Mobile multicast agent's operation

Mobile multicast agent (MMA) is an MA with mobility support function for MNs. MMA has the same function with MA along with additional function to support MNs with mobility.

7.3.1 MMA announcement

MNs' operation is based on MMA, so MMAs must inform MNs of its existence. Thus MMAs send periodically mADVERTISE message into their networks using multicast to announce MMA's existence. When MMAs receive mSOLICIT message sent by MNs, MMAs also transmit immediately the mADVERTISE message to MNs using prescribed multicast address. The mADVERTISE message contains information that is needed by MNs such as, MMA information, control channel information (IP multicast address).

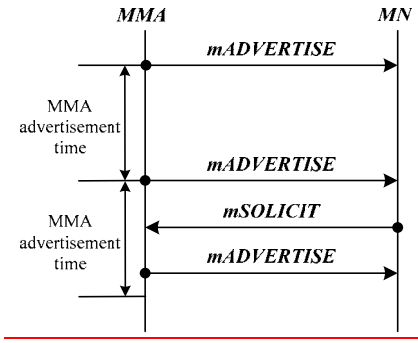


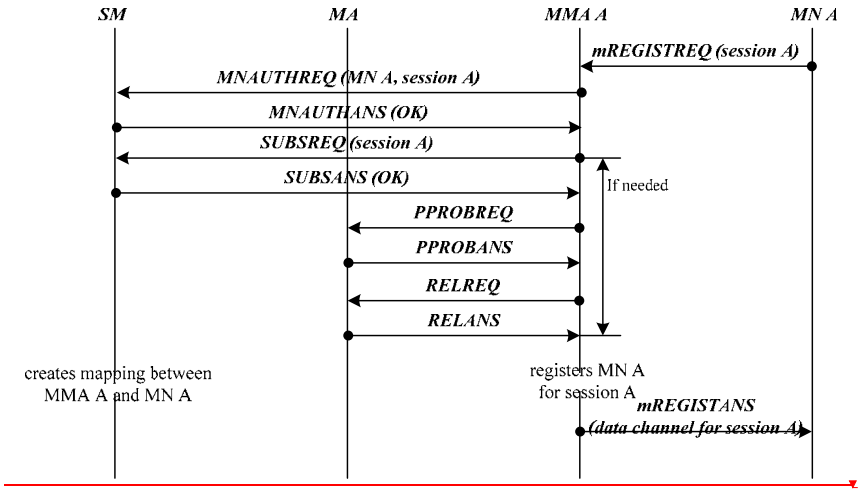
Figure 35 – MMA announcement

7.3.2 Support session join of MN

In case of MNs in the MMA region, MNs can receive session service by registering to the MMA through mREGISTREQ message after session subscription. Upon receiving the mREGISTREQ message from MN, MMAs must register the MN in its local registry and, if needed, request SM for joining the multicast session through SUBSREQ message and join the session tree. When the MMAs ready to provide the request session service to the MN, it informs the MN of the data channel (IP multicast address) for the session. Then the MN listens to the data channel to receive multicast service. For efficient service, MMA maintains the number of MNs for each session. If the number of MNs for a specific session becomes to zero which means that there is no MNs belonging to the MMA for the session, then the MMA should not forward the session data.

삭제됨: SMA
삭제됨: n
삭제됨: SMA
삭제됨:
삭제됨: This clause only describes mobility su… [5]
삭제됨: The
삭제됨: can exist any… [6]
삭제됨: the
삭제됨: the
삭제됨: The
삭제됨: broadcasts
삭제됨: the wireless interface
삭제됨: the
삭제됨: , servicing session
삭제됨: The mADVE… [7]
삭제됨:
삭제됨: The
삭제됨: multicast
삭제됨: After
삭제됨: the MMA
삭제됨: es
삭제됨: the
삭제됨: the
삭제됨: ,
삭제됨: If
삭제됨: can
삭제됨: multicast
삭제됨: MA
삭제됨: T
삭제됨: uses
삭제됨: may
삭제됨: which belong… [8]
삭제됨: specific
삭제됨: equals
삭제됨: broadcast

Since MMAs can not identify whether requesting MN is proper to register, MMAs send an MNAUTHREQ message including the information about requesting MN to SM. In Figure 36, MMA A request to authenticate MN A by sending the MNAUTHREQ message. Then SM checks whether the MN A was subscribed or not for session A. If MN A was subscribed before sending mREGISTREQ message to MMA A, the SM creates mapping between MMA A and MN A in its managing database, e.g. table, and replies with an MNAUTHANS message meaning authenticate successful. Otherwise, the SM sends an MNAUTHANS message indicating authentication fail. MMA A conducts remaing procedure according to received MNAUTHANS message. If MN A can receive the session service, MMA A registers MN A for session A and sends mREGISTANS message which includes data channel for session A. Otherwise MMA A sends mREGISTANS indicating fail to register.

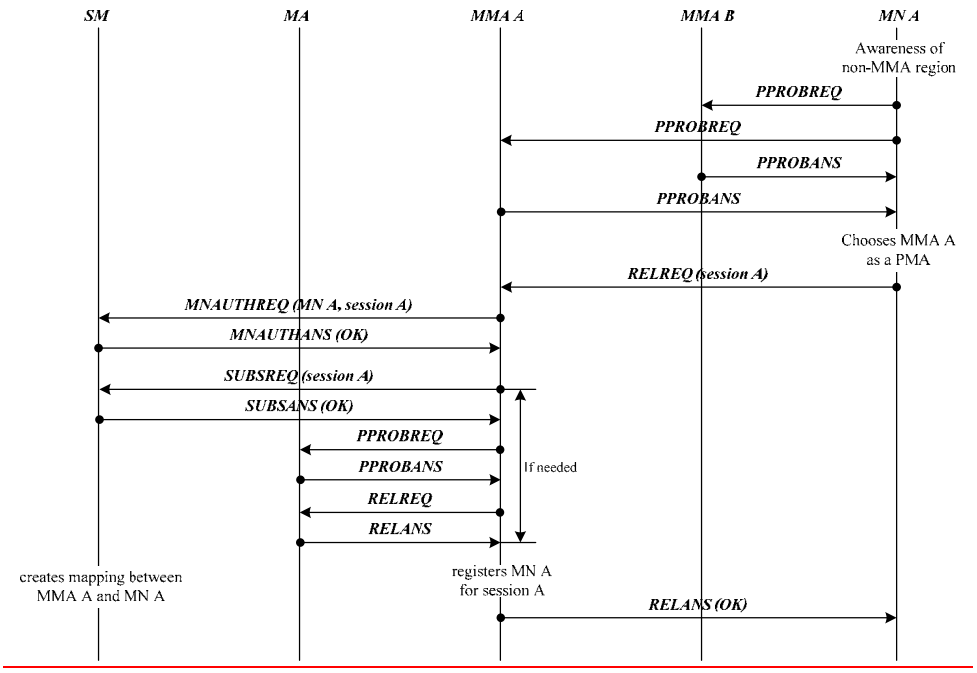


삭제됨:

Figure 36 – Session join of MN (MMA region)

서식 있음: 표준

In case of MNs in the non-MMA region, each MN can notify that it is in the non-MMA region after MMA discovery procedure described in clause 7.4.2. Then it conducts parent probing procedure based on neighbor list which includes only MMAs. MMAs operate same as MA's parent probing procedure. When each MN chooses one its PMA candidate, it sends a RELREQ message to a specific MMA. Upon receiving the RELREQ message, MMA checks whether to allow the relay request. If it allows join, MMA puts the MN information into its managing database, e.g. MN table or session table and sends a RELANS message indicating successful join.



서식 있음: Figure

Figure 37 – Session join of MN (non-MMA region)

MMAs should keep the type of MNs because related message is different according to the region wherein the MN operates.

서식 있음: 표준

Editor's note: Need to define MN authenticate method.

서식 있음: 글꼴: 기울임꼴

7.3.3 Support MN handover

Editor's note: Conceptual figure for handover needs to be added.

서식 있음: 글꼴: 기울임꼴, 강조

서식 있음: 글꼴: 기울임꼴, 강조

서식 있음: 글꼴: 기울임꼴, 강조

서식 있음: 글꼴: 기울임꼴

When MMA received registration request (mREGISTREQ) from MN which has come from other network, the MMA should provide continous service to the MN. To achieve it, the MMA should subscribe the session requested by the MN. If the MMA is paricipating in the session, then it can broadcast the multicast data to the MN. Otherwise, the MMA requests session subscription with subscription request message (SUBSREQ) to SM. And then the MMA should inform the previous MMA of handover of the MN and join the session tree. By sending relay request message (RELREQ) to previous MMA, both purposes can be accomplished. To prevent rejection of relay request, urgent join control data could be used. The previous MMA acts as a PMA to the MMA in forwarding session data and recognizes the handover of the MN. After successful join to session tree, the MMA provides session data to the MN. The MMA would find better PMA performing parent probing procedure.

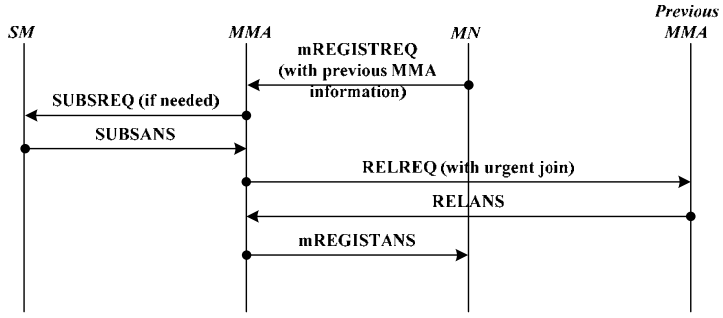


Figure 38 – Supporting MN handover

7.3.4 Data channel setting to prevent duplicated packet

If more than two MMA area overlaps, then MN might receive duplicated multicast packet from all the MMAs in the overlapped area. In order to prevent duplicated packet, MMA should use different data channel in broadcasting multicast packet to the wireless interface. That is, the two MMAs may use different multicast address for same session. The mechanism for differentiating multicast address between adjacent MMAs is not specified in this Recommandation.

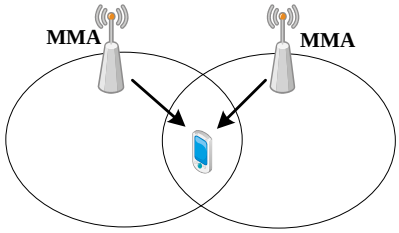


Figure 39 – Receiving duplicated packets

7.3.5 MN management

Each MMA maintain MN table to manage serving session. The MN table may consist of MNID, IP address of MN, connection status, registration request timer. MN can be used within both non-MMA region and MMA region. If MN is used within non-MMA region, MMA provides session data to MN through unicast; otherwise MMA provides session data to MN through multicast. Thus MMA should manage each MN differently. The connection status field is used for that purpose. Each record of the table is refleshed when MMA sends mREGISTANS message as response of mREGISTREQ message. If MMA does not receive the mREGISTREQ message from specific MN within registration request timer, MMA changes the MN's record status to "waiting for deletion". After waiting more N_WAIT_REG times, MMA removes the record from MN table.

7.3.5.1 MN expulsion

When a MMA serves numbers of MNs which operate in remote non-MMA region, it may need to expel some MNs for administrative purpose, e.g. resource limitation. Figure 40 shows procedure for MN expulsion by MMA. MMA sends a LEAVREQ message to a certain MN to be expelled. Then MN fins other MMA and it replies with a LEAVANS message.

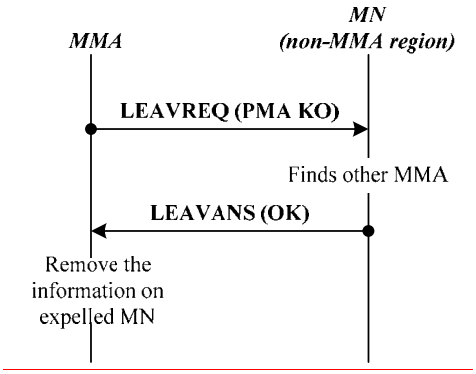


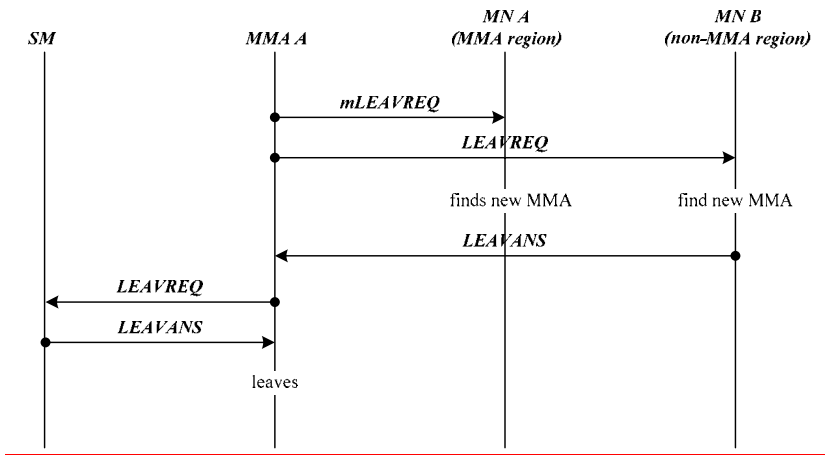
Figure 40 – MN expulsion by MMA

7.3.6 Leave

7.3.6.1 When MMA leaves session(s)

MMAs may leave the subscribed session for administrative purpose or may be failed. In case of departure, MMAs should let both PMA and CMA(s) as well as MNs which it manages know its departure. In case of notifying to other MAs, i.e. PMA and CMA(s), MMAs act as like MA. In case of notifying to MNs, however, MMAs act differently. MMAs send mLEAVREQ message to MNs in its managing MMA region using multicast but send LEAVREQ message to MNs in remote non-MMA region using unicast. Figure 41 shows procedure for MMA leave.

서식 있음: 제목 4
서식 있음: Figure
서식 있음: Figure_No & title
서식 있음: 제목 3
서식 있음: 제목 4



서식 있음: Figure

서식 있음: Figure_No & title

Figure 41 – MMA leave procedure

Since there is a member in its region, MMA A sends mLEAVREQ message into its region. Then MN A should find new MMA as described in clause 7.4.3.2. MMA A sends a LEAVREQ message to MN B and then MN B finds new MMA.

NOTE 1 – MN A does not conduct MMA solicitation because it knows that MMA which manages its belonging region leaves.

서식 있음: Note 1

NOTE 2 – MN B does not conduct MMA solicitation because it knows that there is no MMA in its belonging region.

Because MMAs send an MNAUTHREQ message to SM when it receives mREGISTREQ message from MN, SM can modify MN information, especially registered MMA information.

When MMA wants to depart from all subscribed session, e.g. shut down, MMA sends mLEAVREQ message or LEAVREQ message to all its MNs and sends LEAVREQ message including all of subscribed session information to SM.

Editor's note: control data for LEAVREQ message to indicate departure from multiple sessions should be defined.

서식 있음: 글꼴: 기울임꼴

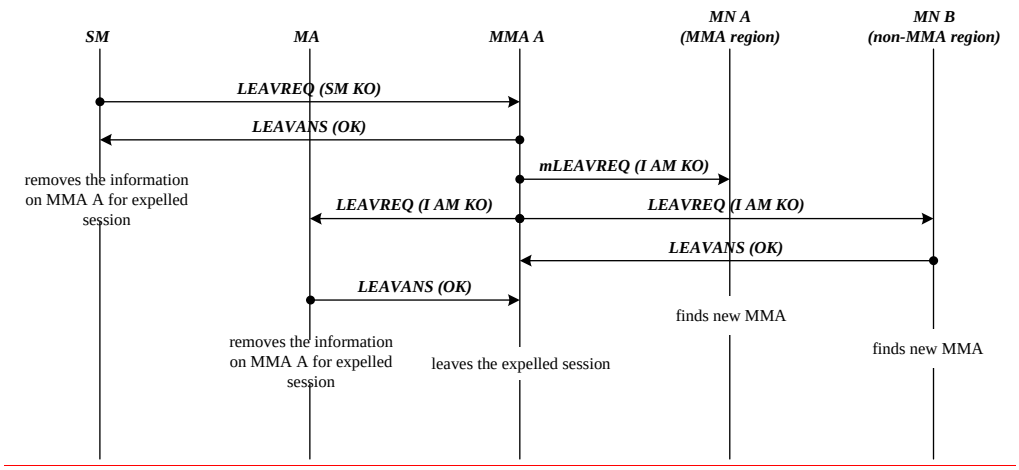
서식 있음: 글꼴: 기울임꼴

7.3.6.2 When MMA is kicked out

서식 있음: 제목 4

Since MA can expel its CMA, MMA can be expelled from its PMA during session. When MMA is expelled from its PMA, MMA operates as described in clause 7.2.3.3, i.e., MMA finds new PMA.

MMA can also be expelled from SM. Since expulsion from SM means expulsion from session, MMA should announce MNs which it manages of its expulsion to let MNs find other MMA. Figure 42 depicts procedure of MMA expulsion by SM. Upon receiving a LEAVREQ message from SM, MMA A relies with a LEAVANS message. Then SM removes the information about MMA A from related session information. After replying to SM, MMA A sends mLEAVREQ message into its region using multicast and also sends LEAVANS message to its PMA and MN B using unicast. Then both MN A and MN B find new MMA to serve. PMA of MMA A removes the information about MMA A from its CMA information for the session which MMA A is expelled from. After receiving LEAVANS message from all MNs in remote non-MMA region, the MMA leaves the expelled session. Note that MNs in MMA region do not have to conduct MMA solicitation because MNs already know that there is no MMA in its belonging region.



서식 있음: Figure

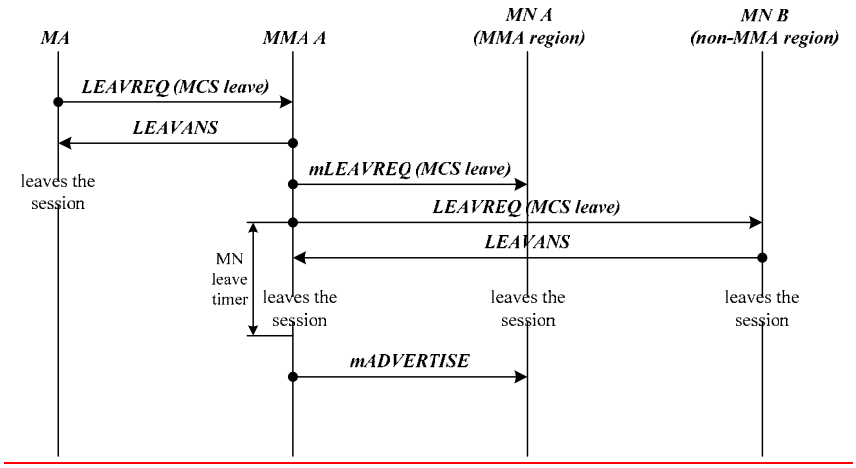
서식 있음: Figure_No & title

Figure 42 – MMA expulsion by SM

7.3.6.3 When MCS leaves the session

When MCS leaves its session, the session should be terminated. Procedure from MCS to MAs is described in clause 7.2.3.4. In this clause, Procedure from MMA to MNs is described. Figure 43 shows message flow for MCS's departure at MMA point of view. MMA sends a mLEAVREQ message into its region using multicast and sends a LEAVREQ message to MN B using unicast. Note that both mLEAVREQ message and LEAVREQ message include reason code indicating that MCS leaves session. After that time, MMA A sends an mADVERTISE message including the closed session information with is set session lifetime to zero.

When MMAs send LEAVREQ message to MNs in non-MMA region, MMAs start MN leave timer to retransmit LEAVREQ message if MNs which does not receive the LEAVREQ message exist. During timer, MMA removes the information on replied MNs from its database. If unanswered MNs exist upon timer expiration, MMA retransmits a LEAVREQ message to unanswered MNs and resets the timer again. Sending a LEAVREQ message to MN in remote non-MMA region is done prescribed times. After all MNs in non-MMA region answer, MMA can leave the session.



서식 있음: Figure

서식 있음: Figure_No & title

Figure 43 – Message flow for MCS's departure

서식 있음: 제목 3

7.3.7 Reporting

The SM can request certain information by sending an STREQ message to MMAs. The SM can also request the information about MNs which each MMA manages. Then the SM can use the reported information for management.

For example, SM can request system information, the information about address or uptime of managing MNs. Figure 44 shows the reporting procedure when a MMA receives an STREQ message from the SM.

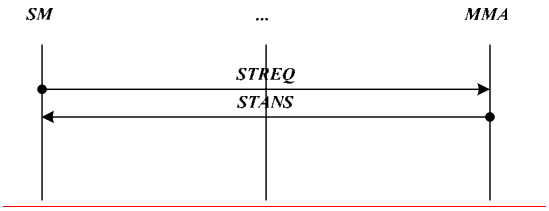


Figure 44 – MMA monitoring by status report

7.3.8 MN failure detection

MN may be failed during the session. Since uptime information of MN may be needed to maintain billing information or etc, MMAs should have capability to detect MN failure and to let SM know it. MMAs can recognize failure of MN if it does not receive a mREGISTREQ message sent from registered MNs in the MMA region or not receive a RELREQ message sent from registered MNs in the non-MMA region.

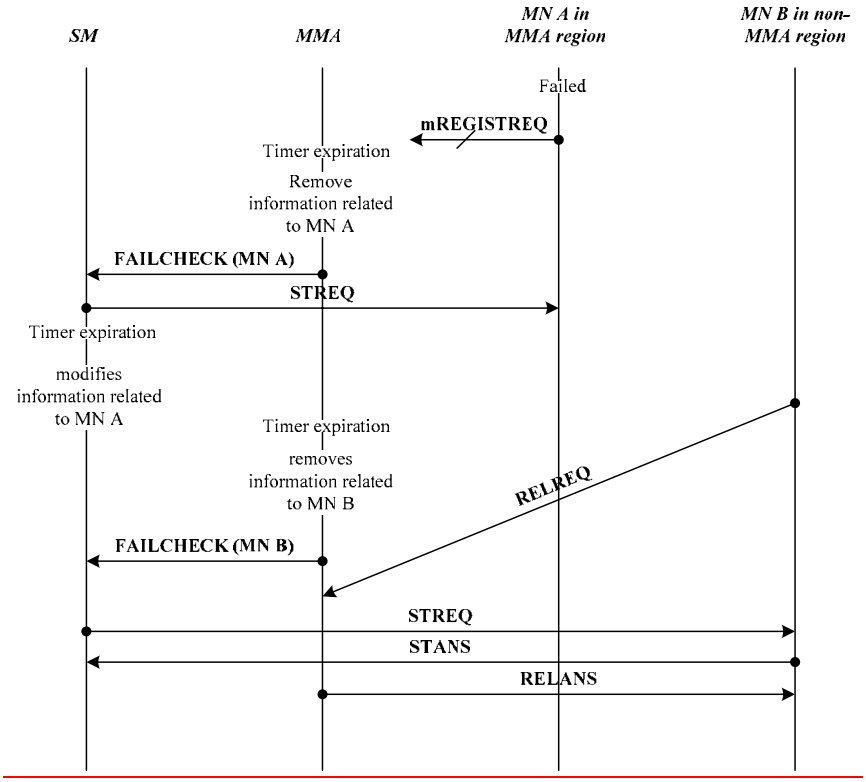


Figure 45 – MN failure detection and handle

Figure 45 shows how a MMA detects MN failure and how to handles it. Since MNs in the MMA region send mREGISTREQ message to MMA priodically, MMA can detect failure of MN if the mREGISTREQ message is not arrived from MN A until timer for MN A expires. Then the MMA sends FAILCHECK message to the SM and also removes information related to MN A. The SM sends STREQ message to MN A but MN A does not answer. Thus the SM modifies the information related to MN A for example, uptime of MN A. Although the MMA considers that MN B is failed, MN B is still alive; the RELREQ message sent by MN B is delayed because of network problem. In this case,

서식 있음: Figure
서식 있음: Figure_No & title
서식 있음: 제목 3
서식 있음: Figure
서식 있음: Figure_No & title

MN B sends STANS message to the SM in response to STREQ message. Thus the SM does nothing. MN B sends RELREQ message to the MMA again because it does not received RELANS message from the MMA.

MMAs may not receive either mREGISTREQ message or RELREQ message from a certain MN when the MN moves from one network to another network. In such case, MMAs remove the information related to the moved MN but detailed description is explained in clause 7.4.5.

7.3.9 Termination

When subscribed session should be terminated, MMAs should leave the session and should also announce termination of the session to MNs which it manages.

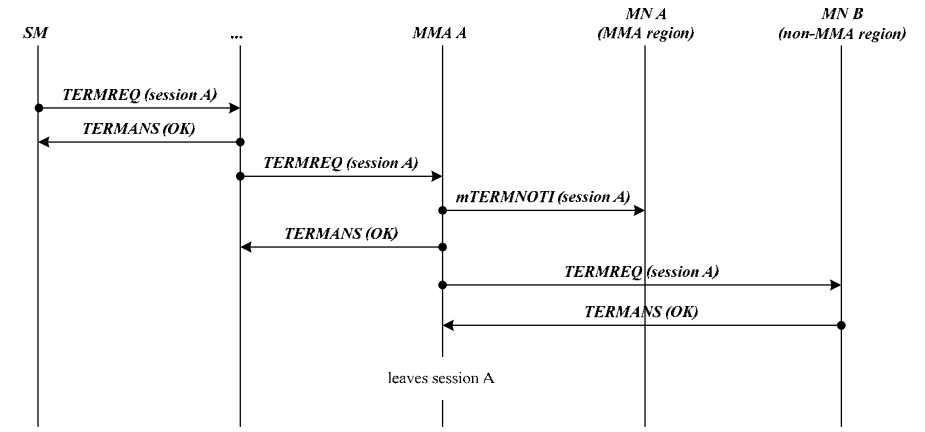


Figure 46 – Termination procedure of MMA

In Figure 46, MMA A sends an mTERMNOTI message into its managing region for MN A using multicast and sends a TERMREQ message to MN B using unicast. MNs in MMA region does not have to answer. The reason is as follows:

a) Multiple answers may cause collision in local wireless network.

b) MNs can recognize about termination of subscribed session when it can not receive mREGISTANS message.

When MMAs leave the terminated session, MMAs should remove MN information for the session.

7.4 Mobile Node's operation

7.4.1 Initiation

The Mobile node (MN) must go through the initiation process after a power on. The MN need to go through authentication and authorization process to get connection access to the network. After the network attachment, MN can subscribe to the MMC-3 session through SM. Before the initiation process, the MN needs the MMC-3 session information. The MMC-3 session information is announced through various methods, such as web-page, e-mail, etc. The user of the MN selects the multicast session and starts the MN initiation process.

MNs send a SUBSREQ message to the SM to subscribe to the multicast session as shown in Figure 47. The SUBSREQ message contains information for the SM to decide whether to accept or reject the multicast session subscription request by MNs. In order to implement a manageable multicast service, the SM needs an authentication method to verify MNs and authorization method to approve of the MNs' request. The decision rule for authentication and authorization is dependent on the service provider. Therefore, this Recommendation does not specify the precise SM's decision rule in accepting the new MN's subscription request.

Once the SM decides to accept the MN's session subscription, bootstrap information is given to the MN in the SUBSANS message which contains the result of the subscription and the active list of MMAs that can provide the subscribed MMC-3 service to the MN. The list of MMA is needed in case the MN cannot find a local MMA that provide service to the wanted session. If the SM decides to reject the MN's session subscription, then it would also return the SUBSANS message containing the reason for rejection.

서식 있음: 제목 3

서식 있음: Figure

서식 있음: Figure_No & title

서식 있음: enumlev1, 번호 매기기 + 수준:1 + 번호 스타일: a, b, c, … + 시작 번호: 1 + 맞춤: 왼쪽 + 맞춤 위치: 39.7 pt + 탭 간격: 0 pt + 들여쓰기 위치: 57.7 pt

삭제됨: by the MMC-3 service administrator

삭제됨: The

삭제됨: s

삭제됨: the

삭제됨: the

삭제됨: s

삭제됨: the MN can receive

삭제됨: the MMC-3 service

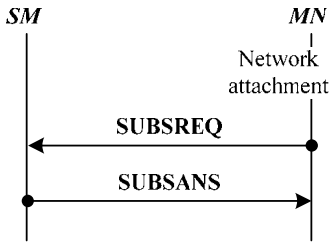


Figure 47 – Mobile Node subscription

After the successful subscription process, MNs need to find a MMA through MMA discovery procedure which is described in the next clause.

삭제됨: s

7.4.2 MMA discovery

After the initiation, MNs need to find a MMA which provides the subscribed service. Since MMA may exist in the local network, MNs send an mSOLICIT message to the predefined multicast address given by the SM. MMA replies with an mADVERTISE message, if it receives the mSOLICIT message as shown in Figure 48.

삭제됨: s

삭제됨: .

삭제됨: s

삭제됨: the MMA exists in the network

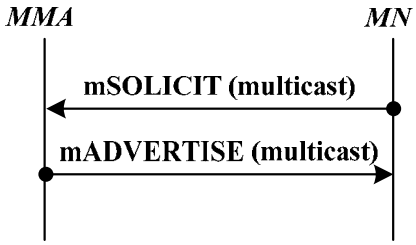


Figure 48 – MMA discovery procedure

If the MMA is not in the network, MN will not get an mADVERTISE message. In this case, MN skips the MMA discovery procedure and performs the session join function in the following clause.

삭제됨: reply

7.4.3 Session join

Session join procedure is for MN to join the MMC-3 session and to receive MMC-3 multicast service. Figure 49 and Figure 50 shows procedure in MN joining the MMA.

7.4.3.1 Network with MMA

This procedure is for the session join of MN in the area with MMA. MN sends the mREGISTREQ message to MMA in multicast. If the MMA is not currently servicing the requested session, then it needs to join the session. The session join procedure of MMA is equivalent to procedure defined in clause 7.2.

If the MMA is ready providing the session, it sends a reply with mREGISTANS message. The flow for the session join in network with MMA is illustrated in Figure 49.

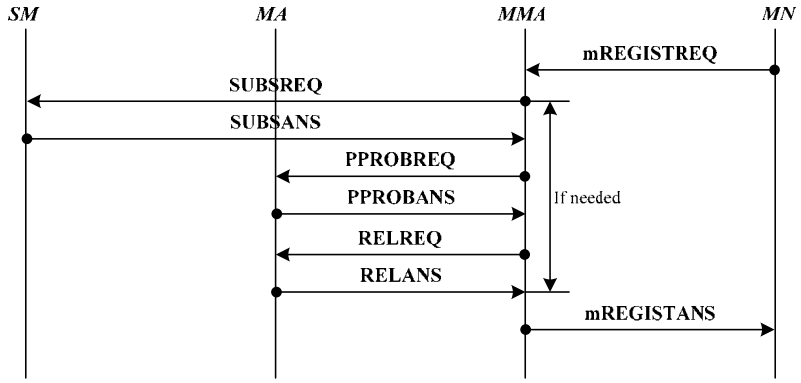


Figure 49 – Session join with MMA

After a successful join, MN and MMA exchange mREGISTREQ/mREGISTANS message periodically to maintain service session.

삭제됨: multicast

7.4.3.2 Network without MMA

This procedure is for the session join of MNs in the area without MMA. Since there is no MMA in the local network, the MN must try to find an appropriate MMA that can provide MMC-3 service. The SM has given a list of MMAs in the SUBSANS message when MNs subscribe the session. Thus, MN can join by requesting connection to each MMA in the list.

삭제됨: with the MMC

MN sends a RELREQ message to remote MMA in the list through unicast. If the remote MMA is unable to service the MN, then it sends a RELANS message with a reason indicating rejection. If the remote MMA can support the MN, it will reply with a RELANS message indicating successful join. If the MMA did not subscribe the requested session, it should subscribe and join the session. The join procedure for the remote MMA is equivalent to procedure defined in clause 7.2.

삭제됨: the
삭제됨: will
삭제됨: rejection in the
삭제됨:
삭제됨: successful the
삭제됨: by
삭제됨: providing
삭제됨: the

If the MMA is ready to provide the session service, it sends a reply with a RELANS message. The flow for the session join in network without MMA is illustrated in Figure 50.

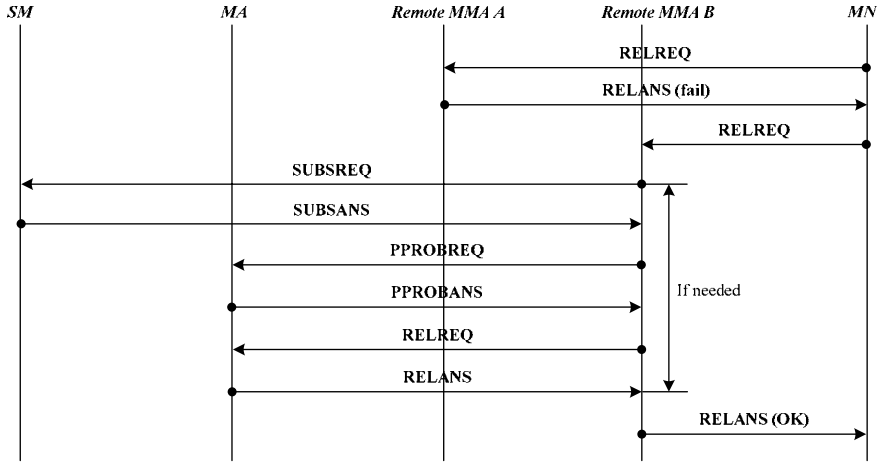


Figure 50 – MN Session join without MMA

After a successful join, MN and MMA exchange periodically the RELREQ/RELANS message to maintain the service session.

삭제됨: multicast

7.4.4 Session leave

Figure 51 shows procedure for session leave of MN. To leave the subscribed session, MN A just stops sending mREGISTREQ message to MMA and sends a LEAVREQ message to SM. Then SM modifies the information related to the MN A and replies with LEAVANS message. When MMA does not hear mREGISTREQ message from MN A for prescribed time period, then the MMA removes MN A's information from its register for the session which the unanswered MN had subscribed. In addition, MMA sends a FAILCHECK message to SM to check whether MN A is failed or not because the MMA consider that MN A is failed. Upon receving the FAILCHECK message, SM checks whether the MN A is subscribed for the session to be checked. Since SM already modifies the information related to MN A, SM ignores the FAILCHECK message.

Since the number of members, MNs, for the session equals to zero, MMA stops sending periodic RELREQ message for the session to its PMA; as a result, the session data forwarding to the MMA is terminated and the MMA does not provide the session service until it receives request from MN.

MN B sends a LEAVREQ message to MMA and also sends a LEAVREQ message to SM. Upon receiving the LEAVREQ message, both SM and MMA modify the MN information and responds with a LEAVANS message. Upon receiving LEAVANS messages, the MN leaves the session promptly.

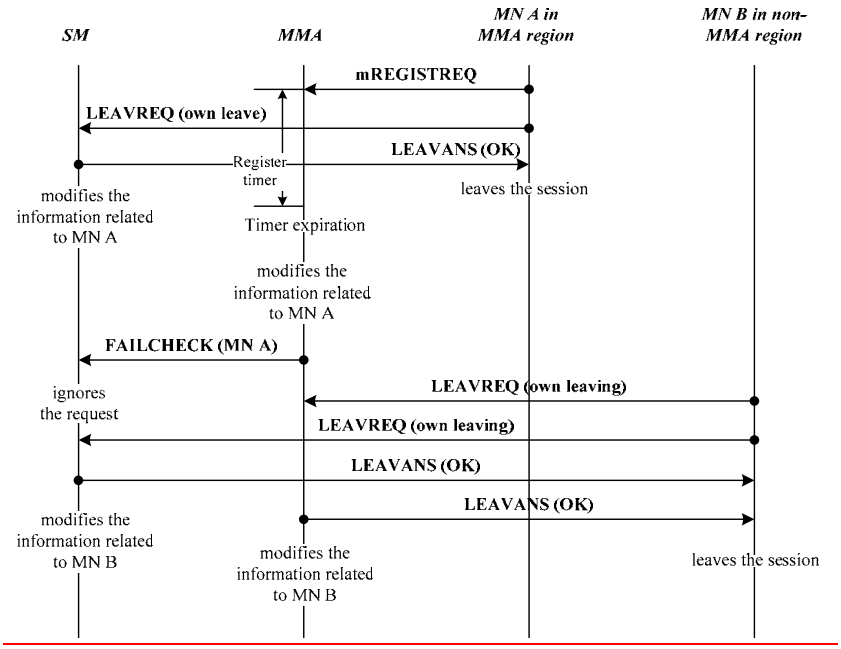


Figure 51 – Procedure for session departure of MN

7.4.5 Handover

Editor's note: Conceptual figure for handover need to be added.

Handover procedure is for MN to switch its MMA when the MN moves from one network to another mobile network. The handover procedure can be defined in two approaches. The first approach is to define handover for movement to MMA region, the second approach is movement to non-MMA region.

7.4.5.1 MN movement to MMA region

When MN, which is served by old MMA, moves into a new mobile network which is managed by new MMA, the MN perceives that it is time to change its MMA. MN can perceive movement by receiving mADVERTISE message from new MMA or by disconnection with the old MMA.

When MN founds a new MMA, MN tries to join with the new MMA by sending an mREGISTREQ message. New MMA which receives MN's registration request subscribes to requested service session by exchanging subscription messages with SM.

삭제됨: MMC-3
삭제됨: a specific
삭제됨: specific
삭제됨: the
삭제됨: status
삭제됨: If
삭제됨: specific
삭제됨: then the
삭제됨: parent
삭제됨: .
삭제됨: If the leaving MN is in non-MMA region, the
삭제됨: the serving
삭제됨: the
삭제됨: removes
삭제됨: from its register
삭제됨: R
서식 있음: Figure
서식 있음: Figure_No & title
삭제됨: area
삭제됨: area
삭제됨: area
삭제됨: MMC-3

After the new MMA's successful attachment, MN receives an mREGISTANS message of successful registration; then the MN receives data from new MMA but refuses the data from old MMA.

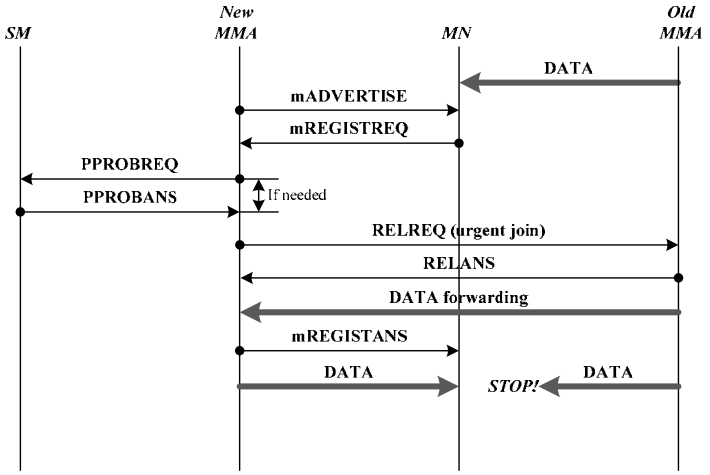


Figure 52 – Handover procedure in MMA area

7.4.5.2 MN movement to non-MMA region

삭제됨: area

MN can move to non-MMA region. The MN will notice that the data it should be receiving is lost and find no reply to the mSOLICIT message. The MN will know that it has moved to the region without MMA.

삭제됨: area

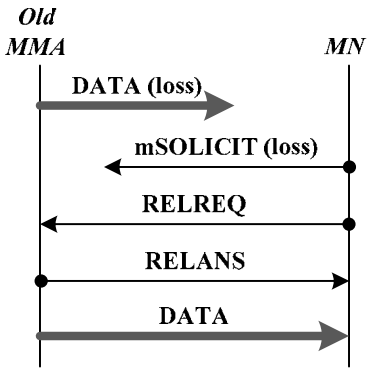


Figure 53 – Handover procedure in non-MMA region

삭제됨: area

MN sends a RELREQ message to the old MMA. Then the old MMA answers with a RELANS message. If the old MMA lacks resource in forwarding data to the MN, then it will reply with a failure message. In that case, the MN has to start the session join with the new MMAs in the list of MMAs given by the SM. If the old MMA can provide MMC-3 service for the MN, it sends a RELANS message indicating successful join and then it forwards the session data to the MN.

삭제됨: T

삭제됨: the

삭제됨: will

삭제됨: multicast

NOTE – Although MN can not perform the seamless handover when it moves to the non-MMA region, but the MN can still receive the session service continuously if the MMA supports buffering capability.

서식 있음: Note 1

7.4.6 Session termination

The MMC-3 session can terminate. The session termination is initiated by the SM. Figure 54 shows the procedure of session termination. The SM sends TERMREQ message to the MA in the terminating session tree. If the MA receives a TERMREQ message, it sends TERMANS message back to SM and then forwards TERMREQ message to child MA which can be MA or MMA. All nodes in the MMC-3 session must terminate the MMC-3 session upon reception of TERMREQ message. The MNs in MMA region do not answer because lots of answer messages from MNs can make

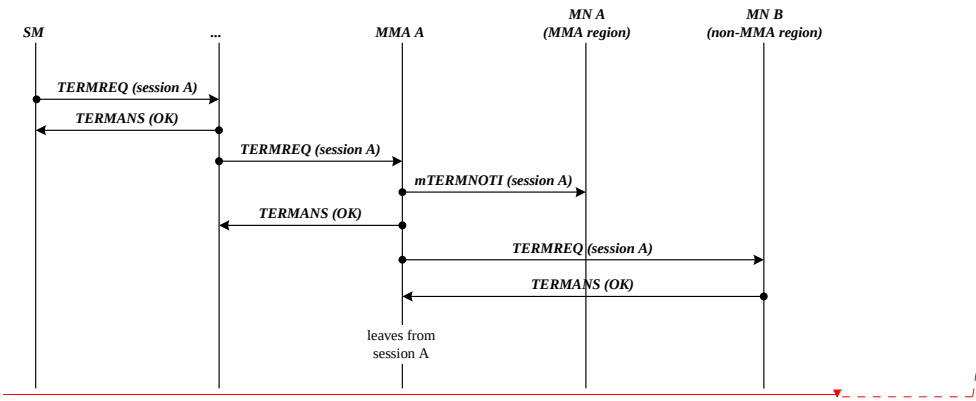
삭제됨: MMC-3

삭제됨: es

삭제됨: send TERMANS message as response of TERMREQ message

삭제됨: TERMANS

congestion in wireless network. However each MN in non-MMA region sends TERMANS message to disconnection connection between MMA and itself.



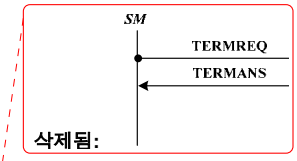
삭제됨:


Figure 54 – Session termination procedure

서식 있음: 제목 3

7.4.7 Reporting

서식 있음: 표준

Although MN does not participate in the control tree, the SM can request the system information of a specific MN to maintain the information about each MN. The STREQ message is also used to check whether a specific MN is alive or not. Figure 56 shows how an MN reports its information when it receives STREQ message.

서식 있음: Figure

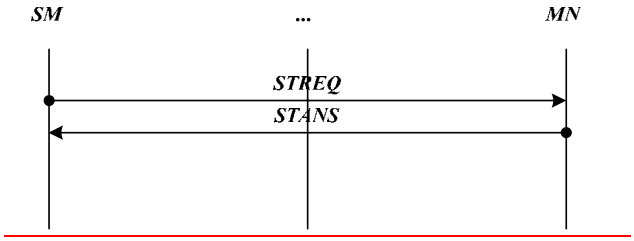


Figure 55 – MN monitoring by status report

서식 있음: 표준

서식 있음: 제목 3

7.4.8 MMA failure detection

MMA may be failed during the session. To provide stable service, failure of MMA should be detected and reported to SM. In addition, MNs served by the failed MMA should recognize failure of MMA and find another MMA to receive service continuously. If an MN does not receive mREGISTANS message in MMA reigon or RELANS message in non-MMA region, it can recognize failure of MMA.

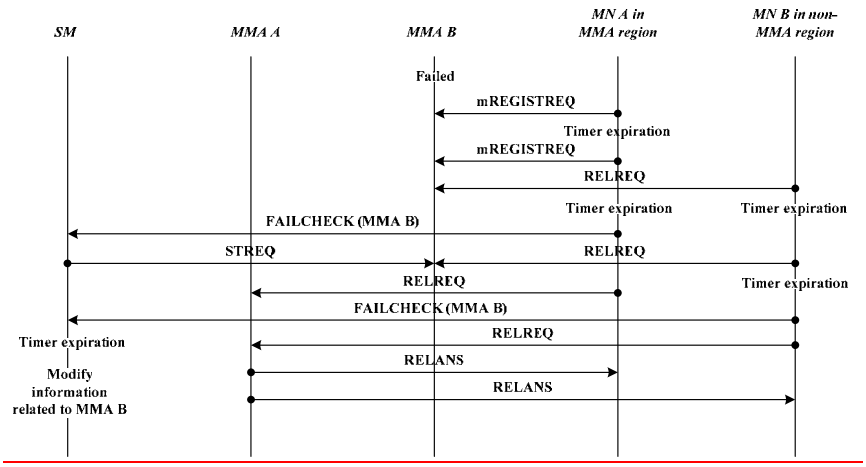


Figure 56 – MMA failure detection

Figure 56 shows how each MN detects MMA failure and handles it. MN A in MMA region sends mREGISTREQ message to MMA B and MN B in non-MMA region sends RELREQ message to MMA B. However MMA B does not answer because of failure. Upon expiration of timer, both MN A and MN B retry message sending. After numbers of trial, they can recognize failure of MMA B and then send a FAILCHECK message to SM. Since both MNs should keep its service continuous, they find another MMA (MMA A) and request data relay. SM sends an STREQ message to MMA B but it does not respond. Thus SM modifies the information related to MMA B. The FAILCHECK message from MN B in Figure 56 arrives later than the FAILCHECK message from MN A. Since SM already sent the STREQ message to MMA, check request from MN B should be ignored to prevent unnecessary duplicated sending. If timer of STREQ is not expired when FAILCHECK message from MN B arrives, SM just ignores the request. If the timer of STREQ is expired and the information related to MMA is modifed, SM also ignores the request from MN B because MMA is not a member of the session which MN B subscribes.

8 Messages

This clause describes the formats and required information of the MMC-3 messages. This Recommendation | International Standard defines MMC-3 message format based on IPv4 network.

8.1 Common format of MMC-3 message

Figure 57 shows the message format containing common format for all MMC-3 messages. The value in the parenthesis represents the length of each field in bits. Each field has the following meaning and value:

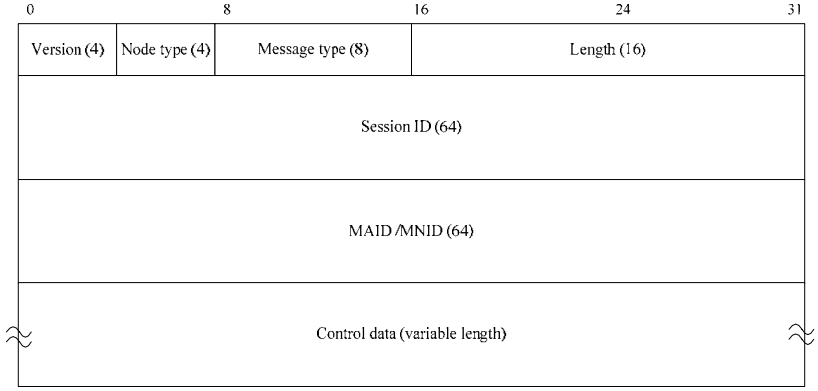


Figure 57 – Common MMC-3 message format

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of SM, MA, MMA, or MN in Table 3;

c) *Message type* – denotes the type of message (see Table 2);

d) *Length* – denotes the totol length of the message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in clause 9.1.1;

f) *MAID / MNID* – denotes the MAID or MNID of the originator or sender of the message. Its value shall contain the local IP address and port number as defined in clause 9.1.2;

g) *Control data* – denotes the control data used by each type of message as necessary.

Session ID and MAID (MNID) must be a unique value to identify the session and MA (MN), respectively. MMC-3 provides the rule for generating the ID value used for session ID and MAID (MNID) in clause 9.1.

8.2 Control data format

Figure 58 shows the MMC-3 control data format. The Control type field describes the type of control data used, and the Length field is the total size of the control data excepting the size of sub-control data. Since the Control type field is 1-byte long, the maximum number of unique control types is limited to 256 cases.

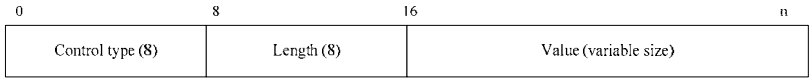


Figure 58 – MMC-3 control data format

a) *Control type* – denotes the type of control data. Its value shall be set to one of coded value in Table 4;

b) *Length* – denotes the total length of the control data (in bytes, except the length of sub-control data field);

c) *Value* – denotes the value for each control data.

Whenever the message needs to specify detailed control information, MMC-3 sub-control data is used. The format of the sub-control data is shown in Figure 59.

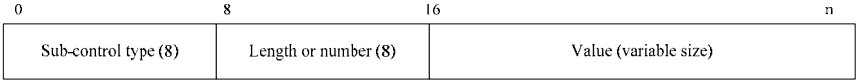


Figure 59 – MMC-3 sub-control data format

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to one of coded value in Table 5 through Table 8;

b) *Length or number* – denotes the length in byte or the number of sub-control data values (depending on the sub-control data value);

c) *Value* – denotes the value for each sub-control data.

Control data can be used with only the control type as shown in Figure 60.

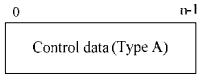


Figure 60 – Use of control type only

Whenever sub-control data is used, an appropriate control data must precede. Figure 61 shows that an example of control data with a sub-control data.

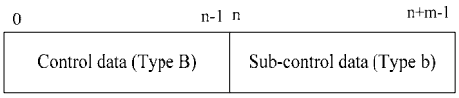


Figure 61 – Use of control data with sub-control data

One or more control data can be used in the MMC-3 Control data field. A MMC-3 message which needs to include multiple control data should align multiple control data as shown in Figure 62.

삭제됨: n

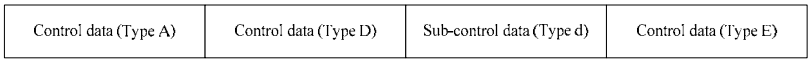


Figure 62 – Use of multiple control data

8.3 MMC-3 control message

This clause defines control messages used in MMC-3. MMC-3 defines ten sets of *request and answer* manner (sometimes called as *request and confirm* manner) of messages and one heartbeat message. MA means both MA and MMA unless otherwise noted.

8.3.1 SUBSREQ

삭제됨: n

8.3.1.1 The SUBSREQ control message is used to subscribe to a MMC-3 session. By issuing the SUBSREQ control message, each MA or MN can obtain bootstrap information from SM when accepted. The message format is shown in Figure 63.

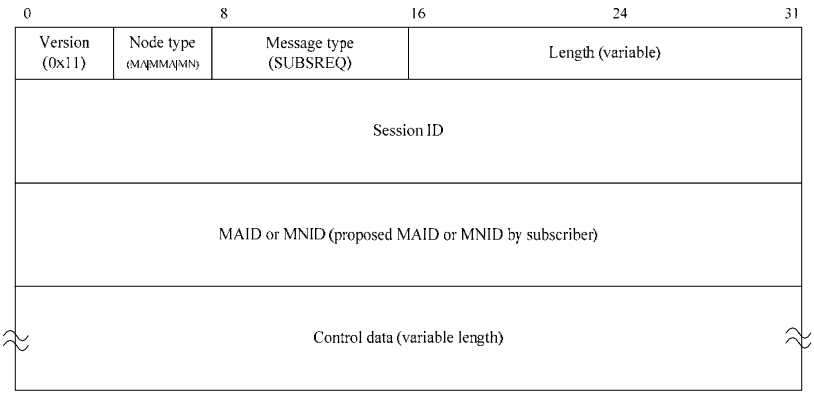


Figure 63 – SUBSREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the type of the node sending the message. Its value shall be set to the coded value for one of MA, or MN in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x02 (see Table 2);

d) *Length* – denotes the length of the SUBSREQ control message including control data (in bytes);

e) *Session ID* – shall be set to a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID or MNID* – shall be set to the proposed MAID or MNID of the subscriber. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.1.2 Following table shows the control data types which can be used within the SUBSREQ message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
SYSINFO	A description of the system information of MA or MN.	O
DATAPROFILE	A description of the requirements for forwarding data.	O
AUTH	Authentication information for verifying the sender.	O

8.3.2 SUBSANS

8.3.2.1 The SUBSANS control message is used by SM to give the results of the session subscription request and bootstrap information for the session. The message format is shown in Figure 64.

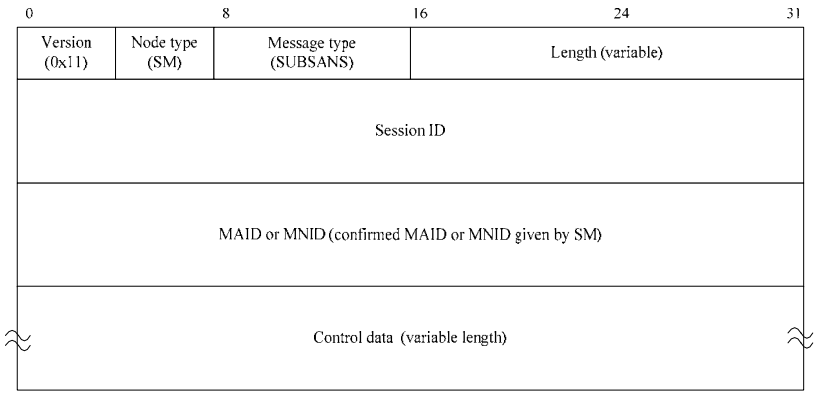


Figure 64 – SUBSANS control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for SM in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x03 (see Table 2);

d) *Length* – denotes the length of the SUBSANS control message including control data (in bytes);

e) *Session ID* – shall be set to a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the confirmed MAID or MNID of the subscriber (confirmed by SM). Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.2.2 Following table shows the control data types which can be used within the SUBSANS message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
RESULT	The result of the subscription request.	M
NEIGHBORLIST	A list of MAIDs for performing the neighbor discovery.	M
DATAPROFILE	A description of the requirements for forwarding data.	O
AUTH	Authentication information for verifying the sender.	O

8.3.3 PPROBREQ

8.3.3.1 This is used to perform the neighbor discovery procedure for determining the actual network condition and for exploring neighbors as well. PPROBREQ control message is also used to check whether its counterpart is still alive. Figure 65 illustrates the format of the PPROBREQ control message.

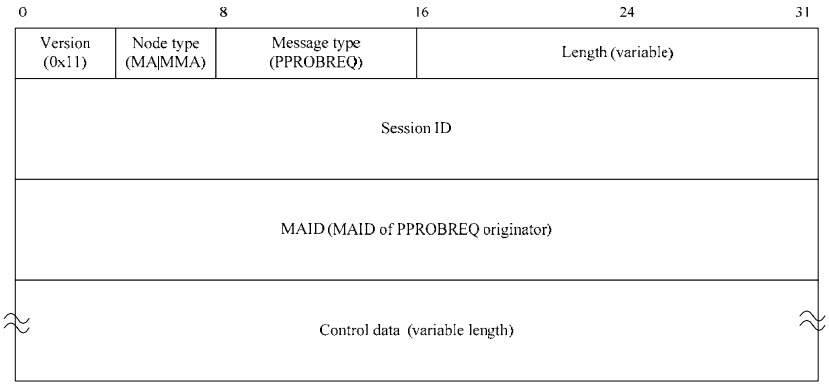


Figure 65 – PPROBREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message originator's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x04 (see Table 2);

d) *Length* – denotes the length of the PPROBREQ control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.3.2 Following table shows the control data types which can be used within the PPROBREQ message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
TIMESTAMP	To measure of distance between sending and receiving MAs / MMAs.	M
DATAPROFILE	A description of the requirements for forwarding data.	O
ROOTPATH	A description of the path from MCS.	O
NEIGHBORLIST	A list of MAs for performing the neighbor discovery.	O
SYSINFO	A description of the system information of MA.	O

삭제됨: SMA

8.3.4 PPROBANS

8.3.4.1 PPROBANS control message is a response to the PPROBREQ control message used in neighbor discovery procedure to confirm available MA in the network. PPROBANS control message may contain the actual network condition values and a series of its neighbor information. Figure 66 illustrates the format of the PPROBANS control message.

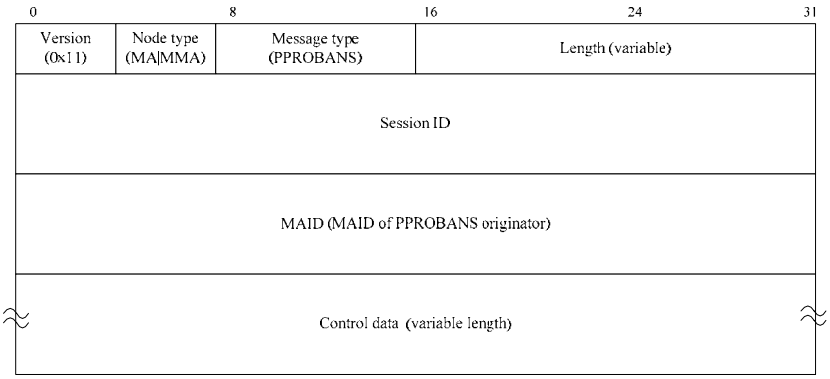


Figure 66 – PPROBANS control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x05 (see Table 2);

d) *Length* – denotes the length of the PPROBANS control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the originator or sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.4.2 Following table shows the control data types which can be used within the PPROBANS message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
TIMESTAMP	To measure of distance between sending and receiving MAs.	M
NEIGHBORLIST	A list of MAs for performing the neighbor discovery.	M
ROOTPATH	A description of the path from MCS.	M
SYSINFO	A description of the system information of MA.	M
DATAPROFILE	A description of the requirements for forwarding data.	O

삭제됨: SMA

8.3.5 HSOLICIT

8.3.5.1 HSOLICIT control message is used to process self-organization in a local network. The purpose of this message is to find the HMA in the local multicast network. Figure 67 illustrates the format of HSOLICIT control message.

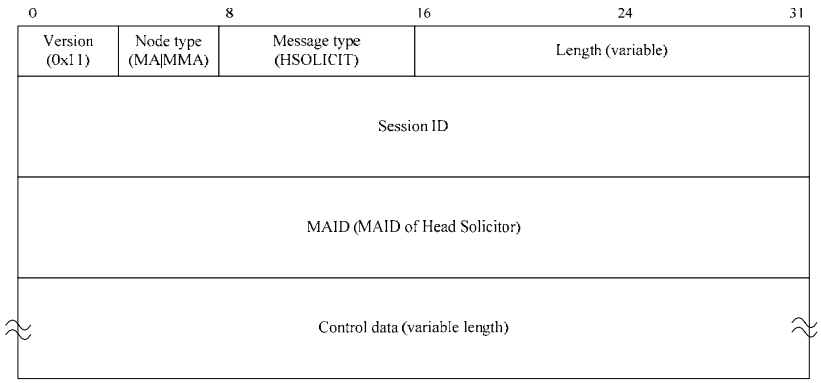


Figure 67 – HSOLICIT control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x06 (see Table 2);

d) *Length* – denotes the length of the HSOLICIT control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.5.2 Following table shows the control data types which can be used within the HSOLICIT message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
AUTH	Authentication information for verifying the sender.	M

8.3.6 HANNOUNCE

8.3.6.1 In response to HSOLICIT control message, HANNOUNCE control message is used to announce HMA's existence in the local multicast network. Figure 68 shows the format of this message.

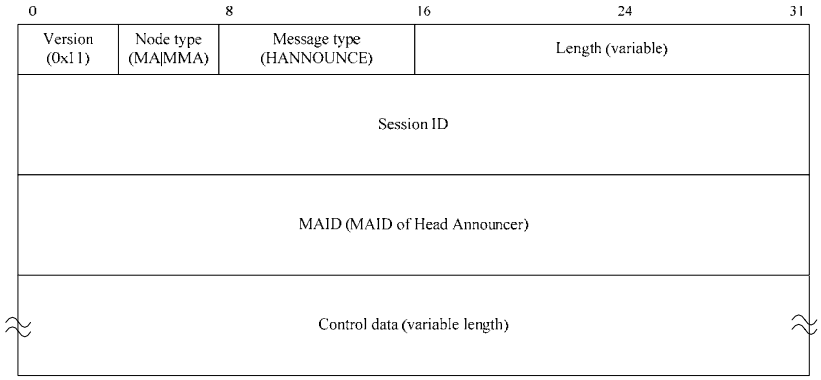


Figure 68 – HANNOUNCE control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x07 (see Table 2);

d) *Length* – denotes the length of the HANNOUNCE control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the HMA. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.6.2 Following table shows the control data types which can be used within the HANNOUNCE message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
AUTH	Authentication information for verifying the sender.	M
SYSINFO	A description of the system information of MA.	M
NEIGHBORLIST	A list of MAs for performing the neighbor discovery.	O

8.3.7 HLEAVE

8.3.7.1 This is used to announce to its local network that HMA is leaving the MMC-3 session. Figure 69 illustrates the format of this message.

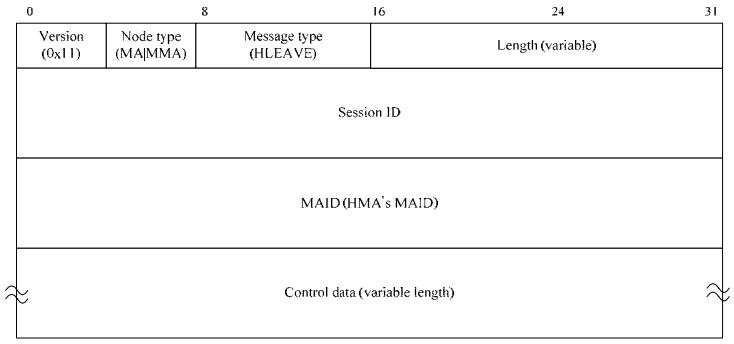


Figure 69 – HLEAVE control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *N*ode type – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x08 (see Table 2);

d) *Length* – denotes the length of the HLEAVE control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the HMA. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.7.2 Following table shows the control data types which can be used within the HLEAVE message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
NEIGHBORLIST	A list of MAs for performing the neighbor discovery.	M
ROOTPATH	A description of the path from MCS.	M
AUTH	Authentication information for verifying the sender.	M
REASON	A reason for leaving of MA.	M

삭제됨: SMA

8.3.8 RELREQ

8.3.8.1 This control message is used by CMA to request data forwarding from PMA. It may include a data profile to negotiate data channel. After relationship is established, MA (or MMA) can keep the relationship by periodical exchanging RELREQ and RELANS control messages. Figure 70 shows the format of this message.

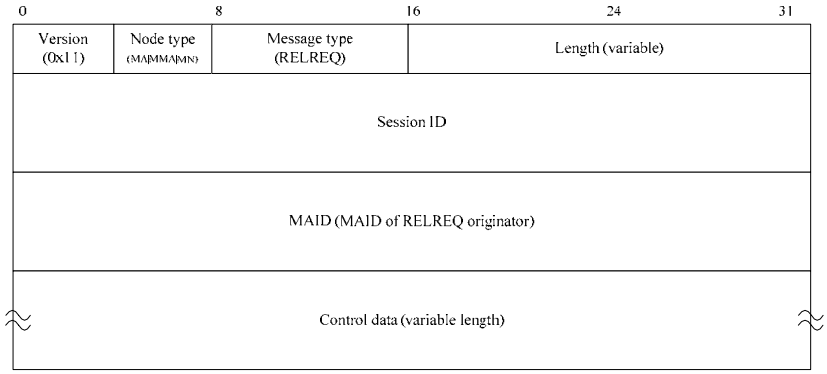


Figure 70 – RELREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to
- the coded value for MA or MMA in Table 3;
- the coded value for MN in Table 3 for relay request when it is in non-MMA region.

c) *Message type* – denotes the type of the message. The value shall be set to 0x09 (see Table 2);

d) *Length* – denotes the length of the RELREQ control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.8.2 Following table shows the control data types which can be used within the RELREQ message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Purpose	M/O
NEIGHBORLIST	A list of MAs for performing the neighbor discovery.	M
DATAPROFILE	A description of the requirements for forwarding data.	O
URGENTJOIN	Indication for urgent join related to supporting handover of MN.	O
MNLEAVE	Indication for leave of MN related to supporting handover of MN.	O
RP_COMMAND	A description for requesting the specific rootpath information of MA.	O
TIMESTAMP	To measure of distance between MMA and MN.	O

8.3.9 RELANS

8.3.9.1 In response to RELREQ control message, RELANS control message is issued by PMA to CMA. The purpose of this message is to specify whether the relay request is allowed. It may also contain negotiated data profile. The message format of RELANS control message is shown in Figure 71.

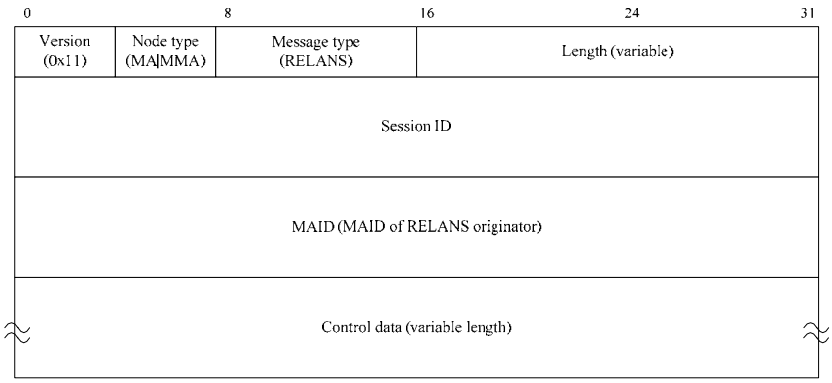


Figure 71 – RELANS control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x0C (see Table 2);

d) *Length* – denotes the length of the RELANS control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.9.2 Following table shows the control data types which can be used within the RELANS message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meanin	M/O
RESULT	A result of the relay request.	M
ROOTPATH	A description of the path from MCS.	O
DATAPROFILE	A description of the requirements for forwarding data.	O
SYSINFO	A description of the system information of MA.	O
NEIGHBORLIST	A list of MAs for performing the neighbor discovery.	O
TIMESTAMP	To measure of distance between MMA and MN.	O

삭제됨: SMA

8.3.10 STREQ

8.3.10.1 STREQ control message is used for monitoring the status of MAs in the session. Figure 72 shows the format of this message.

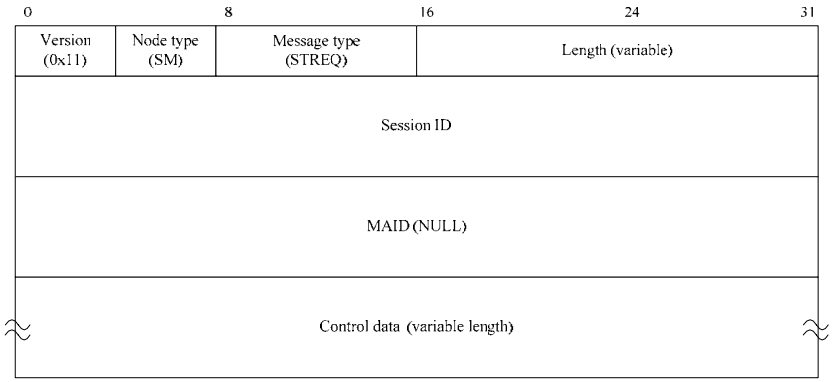


Figure 72 – STREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for SM in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x12 (see Table 2);

d) *Length* – denotes the length of the STREQ control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – Since SM does not have an MAID, this field should be set to zero;

g) *Control data* – may include the following meaning and value:

8.3.10.2 Following table shows the control data types which can be used within the STREQ message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
SI_COMMAND	A description for requesting the specific information of MA.	M
TREEEXPLOR	Limitation of tree scoping for prevention of hazards because of report implosion.	O

8.3.11 STANS

This message is used for reporting the system information of MA. Figure 73 shows the format of the STANS control message.

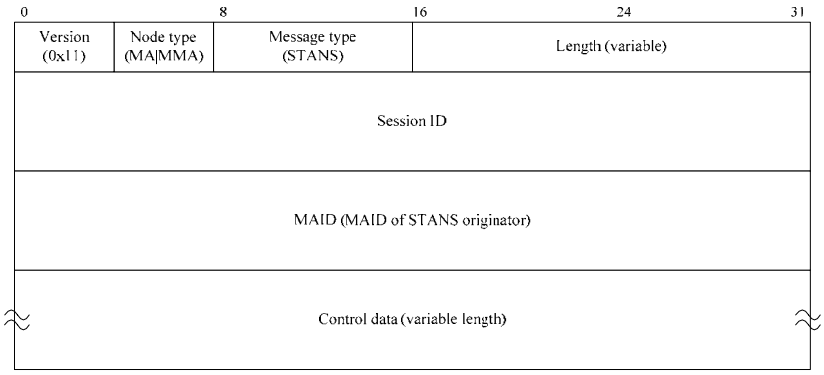


Figure 73 – STANS control message format

The description of each field is as follows:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x13 (see Table 2);

d) *Length* – denotes the length of the STANS control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the originator or sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

Control type	Meaning	M/O
SYSINFO	A description of the system information of MA.	M

8.3.12 LEAVREQ

8.3.12.1 This control message is used for three different purposes, one of which is for leaving. When leaving the MMC-3 session or its PMA for parent switching, MA sends LEAVREQ control message to the corresponding MAs based on the leaving procedure.

SM and PMA may use this control message to expel MA but their targets are different. The target of SM is an MA in the session; that of PMA is only its CMA.

Finally, this control message is used for terminating a session. When MCS leaves the session, this message should be forwarded to the endmost MA in the tree hierarchy. Figure 74 illustrates the format of the LEAVREQ control message.

삭제됨: SMA

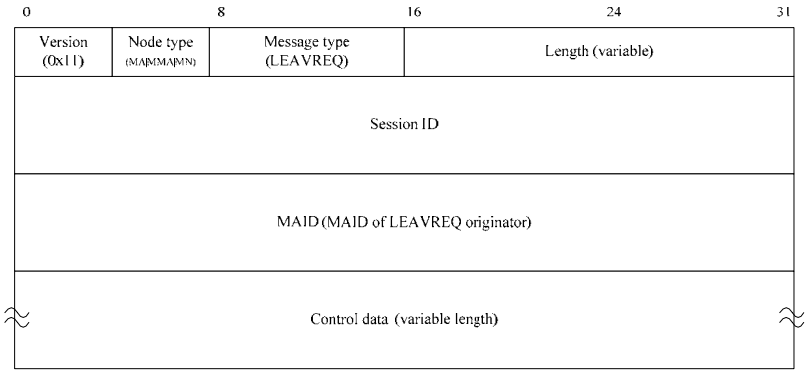


Figure 74 – LEAVREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to
- the coded value for one of MA, MMA in Table 3;
- the coded value for MN in Table 3 for notifying its leaving when it is served by remote MMA.

c) *Message type* – denotes the type of the message. The value shall be set to 0x16 (see Table 2);

d) *Length* – denotes the length of the LEAVREQ control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.12.2 Following table shows the control data types which can be used within the LEAVREQ message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
REASON	A reason for leaving of MA.	M

8.3.13 LEAVANS

8.3.13.1 As a confirmation of the LEAVREQ control message, LEAVANS control message is sent back by the MA receiving LEAVREQ control message. Figure 75 illustrates the format of the LEAVANS control message.

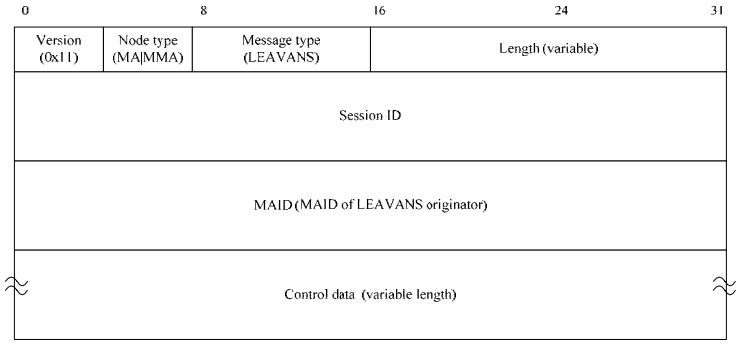


Figure 75 – LEAVANS control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x17 (see Table 2);

d) *Length* – denotes the length of the LEAVANS control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.13.2 Following table shows the control data types which can be used within the LEAVANS message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
RESULT	A result of leave request.	M

8.3.14 TERMREQ

8.3.14.1 TERMREQ control message is used to terminate an existing MMC-3 session. It is issued by the SM and subsequently forwarded by MA/MMA to the endmost MNs along the tree hierarchy. Figure 76 shows the format of the TERMREQ control message.

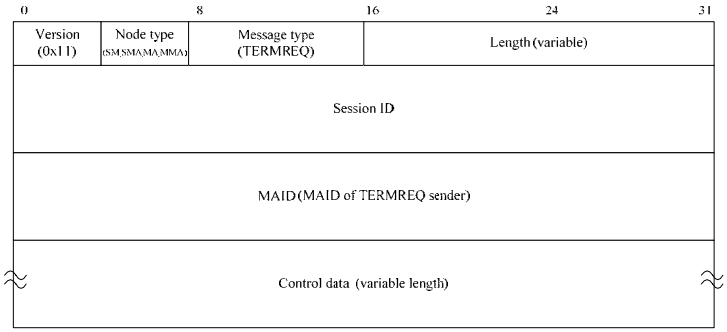


Figure 76 – TERMREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to

- the coded value for SM in Table 3 for request to terminate an specific session;
- the coded value for MCS in Table 3 for notifying of its serving session termination;
- the coded value for one of MA, MMA in Table 3 for relaying termination request;

삭제됨: SMA

c) *Message type* – denotes the type of the message. The value shall be set to 0x19 (see Table 2);

d) *Length* – denotes the length of the TERMREQ control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2 (set to zero for SM);

g) *Control data* – may include the following information:

8.3.14.2 Following table shows the control data types which can be used within the TERMREQ message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
REASON	A reason for terminating the session.	M

8.3.15 TERMANS

8.3.15.1 Figure 77 illustrates the format of the TERMANS control message.

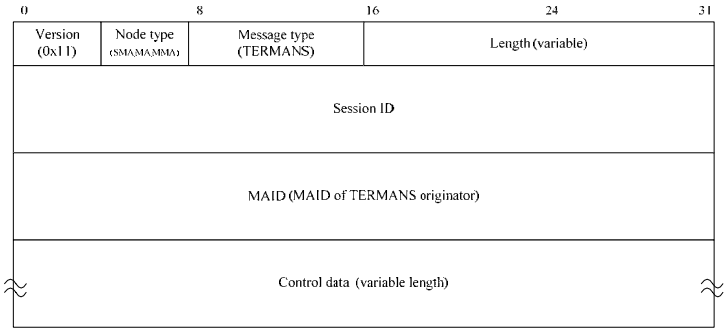


Figure 77 – TERMANS control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to
- the coded value for MCS in Table 3 in case of SM-initiated termincation procedure;
- the coded value for one of MA, MMA in Table 3 for relaying termination request;

삭제됨: SMA

c) *Message type* – denotes the type of the message. The value shall be set to 0x1A (see Table 2);

d) *Length* – denotes the length of the TERMANS control message including control data (in bytes);

e) *Session* ID – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

Control type	Meaning	M/O
RESULT	A result of the termination request.	M

8.3.16 FAILCHECK

8.3.16.1 FAILCHECK control message is used to terminate an existing MMC-3 session. It is issued by SM to check whether a certain MA/MMA/MN is alive by requesting system information. Figure 78 shows the format of FAILCHECK control message.

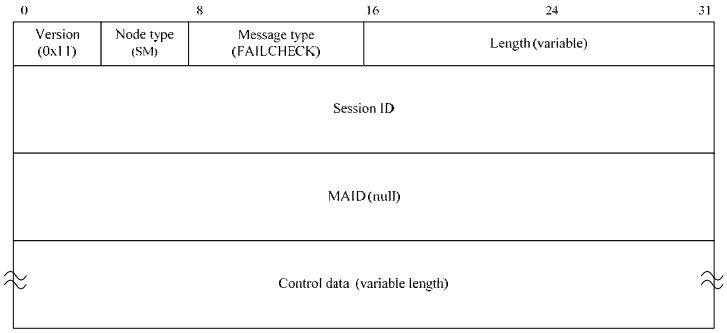


Figure 78 – FAILCHECK control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for SM in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x1C (see Table 2);

d) *Length* – denotes the length of the FAILCHECK control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – Since SM does not have an MAID, this field should be set to zero;

g) *Control data* – may include the following information:

8.3.16.2 Following table shows the control data types which can be used within the FAILCHECK message. Details of following control data are described in clause 8.4.

Control type	**Meaning**	**M/O**
SI_COMMAND	A description for requesting the specific information of MA.	M

8.3.17 mADVERTISE

8.3.17.1 mADVERTISE control message is used by MMA to advertise its existence to MNs; the MNs can be existed or newly arrived one. Figure 79 illustrates the format of the mADVERTISE control message.

서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음

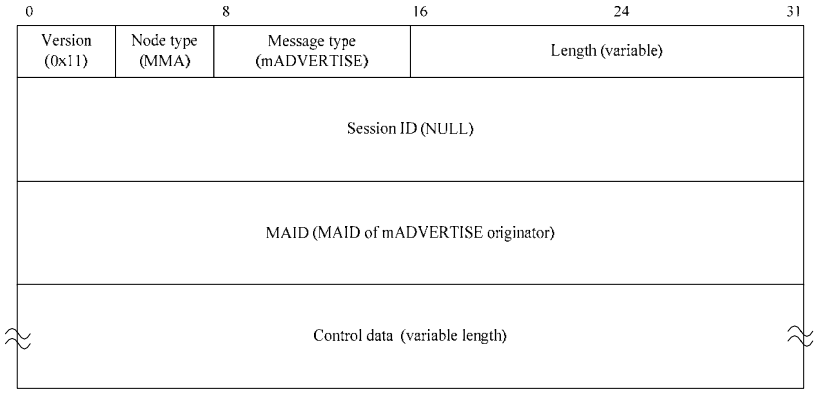


Figure 79 – mADVERTISE control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x21 (see Table 2);

d) *Length* – denotes the length of the mADVERTISE control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the MMA. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.17.2 Following table shows the control data types which can be used within the mADVERTISE message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
SYSINFO	A description of the system information of MMA.	M
TIMESTAMP	To measure of distance between MMA and MN.	O
GROUPINFO	A description of serving session information.	M
LIFETIME	The maximum registration time that MN can register to MMA. Before this time expires, MN has to re-register to MMA.	M
CTRLCH	A description of control channel.	M

8.3.18 mSOLICIT

8.3.18.1 mSOLICIT control message is used by MN to solicit any possible MMA in newly visited network; the MN already existed can use this message when an announcement from MMA has not arrived for a specific time. Figure 80 illustrates the format of the mSOLICIT control message.

서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음

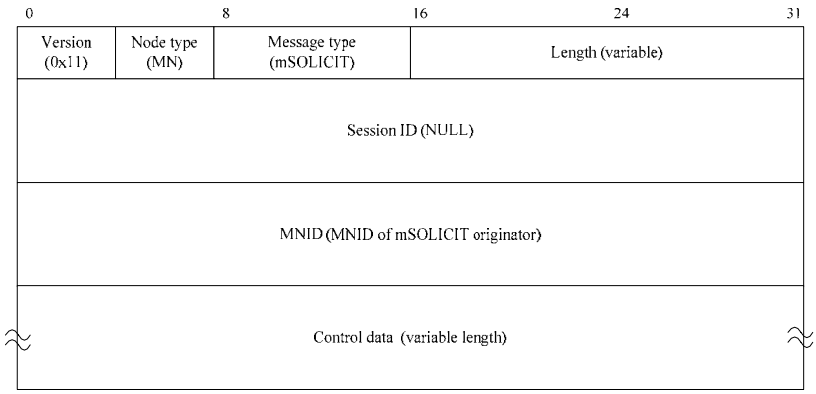


Figure 80 – mSOLICIT control message format

Each field has the following meaning and value:

a) Version – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) Node type – denotes the message issuer's node type. Its value shall be set to the coded value for MN in Table 3;

c) Message type – denotes the type of the message. The value shall be set to 0x22 (see Table 2);

d) Length – denotes the length of the mSOLICIT control message including control data (in bytes);

e) Session ID – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) MNID – denotes the MNID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) Control data – may include the following information:

8.3.18.2 Following table shows the control data types which can be used within the mSOLICIT message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
SYSINFO	A description of the system information of MMA.	O

TIMESTAMP	To measure of distance between MMA and MN.	O

8.3.19 mREGISTREQ

8.3.19.1 This message is used by MN to register itself to MMA. Figure 81 illustrates the format of the mREGISTREQ control message.

서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음

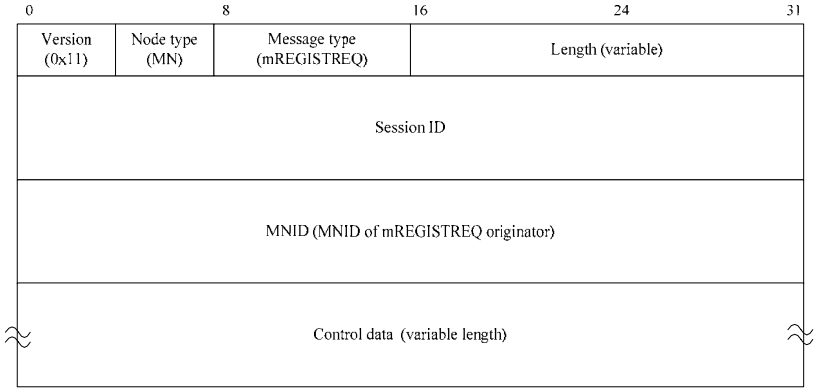


Figure 81 – mREGISTREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for MN in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x23 (see Table 2);

d) *Length* – denotes the length of the mREGISTREQ control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MNID* – denotes the MNID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information

8.3.19.2 Following table shows the control data types which can be used within the mREGISTREQ message. Details of following control data are described in clause 8.4.

삭제됨: **Error! Reference source not found.**

Control type	Meaning	M/O
LIFETIME	The time when MN requests registration to MMA.	M
IDENTIFICATION	An identification code to identify the message.	M
OMAINFO	A description of MMA which served to the MN previously.	O

8.3.20 mREGISTANS

8.3.20.1 mREGISTANS control message is used by MMA to confirm that MMA has registered the MN. Figure 82 illustrates the format of the mREGISTANS control message.

서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음
서식 있음: 글꼴: 굵게 없음

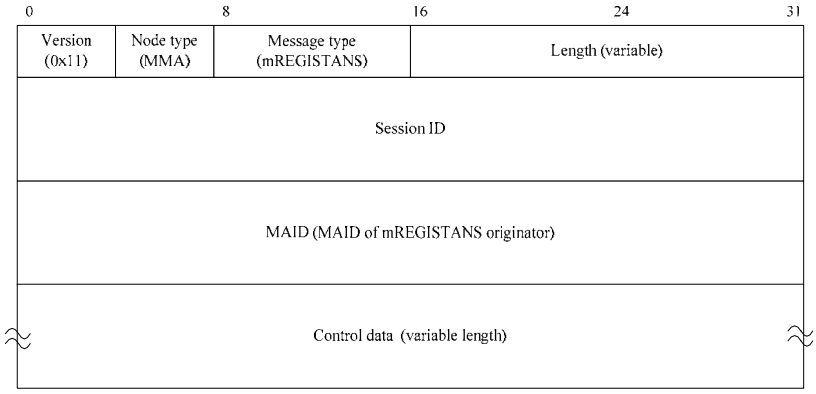


Figure 82 – mREGISTANS control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x24 (see Table 2);

d) *Length* – denotes the length of the mREGISTANS control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information

8.3.20.2 Following table shows the control data types which can be used within the mREGISTANS message. Details of following control data are described in clause 8.4.

삭제됨: Error! Reference source not found.

Control type	Meaning	M/O
RESULT	The result of the registration request.	M
IDENTIFICATION	An identification code to identify the message.	M
DATACH	A description of data channel which delivers session data.	M

서식 있음: 제목 3

서식 있음: 스타일 제목 4 + 10 pt 굵게 없음1

8.3.21 mLEAVREQ

8.3.21.1 mLEAVREQ control message is used by MMA to make MNs in the MMA region leave the session. Figure 83 illustrates the format of the mLEAVREQ control message.

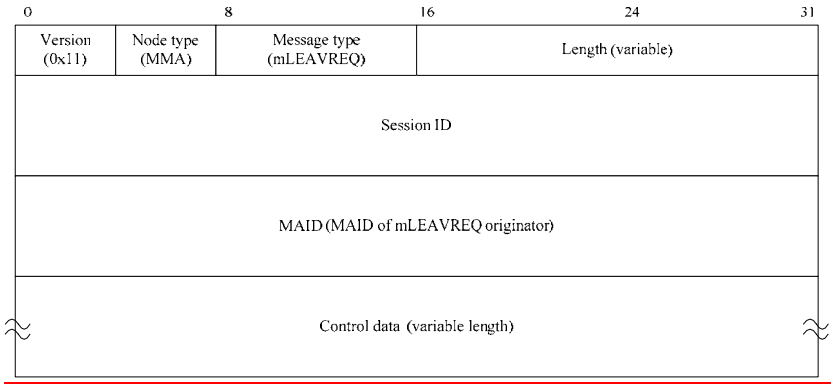


Figure 83 – mLEAVREQ control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x25 (see Table 2);

d) *Length* – denotes the length of the mREGISTANS control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the originator of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information

8.3.21.2 Following table shows the control data types which can be used within the mLEAVREQ message. Details of following control data are described in clause 8.4.

Control type	Meaning	M/O
REASON	A reason for leaving.	M

서식 있음: 제목 3

8.3.22 mLEAVANS

8.3.22.1 As a confirmation of the mLEAVREQ control message, mLEAVANS control message is sent back by the MN receiving mLEAVREQ control message. Figure 84 illustrates the format of the mLEAVANS control message.

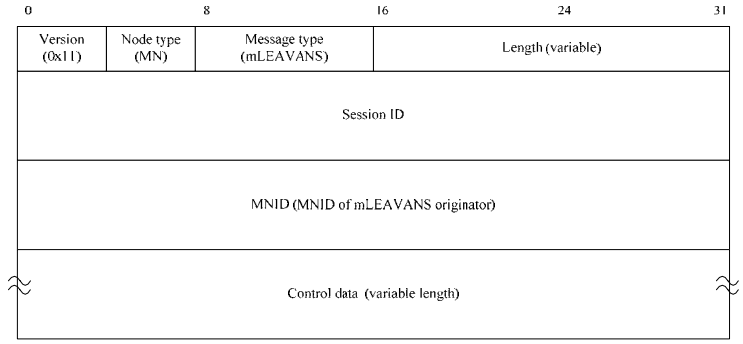


Figure 84 – mLEAVANS control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for one of MA, MMA in Table 3;

c) *Message type* – denotes the type of the message. The value shall be set to 0x26 (see Table 2);

d) *Length* – denotes the length of the mLEAVANS control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MNID* – denotes the MNID of the originator of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.22.2 Following table shows the control data types which can be used within the mLEAVANS message. Details of following control data are described in clause 8.4.

Control type	Meaning	M/O
RESULT	A result of leave request.	M

8.3.23 mTERMNOTI

서식 있음: 제목 3

8.3.23.1 mTERMNOTI control message is used to terminate an existing MMC-3 session. It is issued by MMAs to notify session termination to MNs in MMA region. Figure 85 shows the format of mTERMNOTI control message.

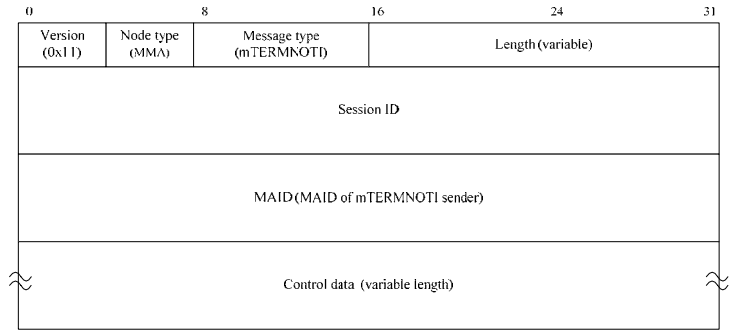


Figure 85 – mTERMNOTI control message format

Each field has the following meaning and value:

a) *Version* – denotes the current version of MMC-3. Its value shall be set to 0x11;

b) *Node type* – denotes the message issuer's node type. Its value shall be set to the coded value for MMA in Table 3;

서식 있음: enumlev1, 왼쪽, 들여쓰기: 왼쪽: 0 pt, 첫 줄: 0 pt, 간격 앞: 0 pt, 단락 뒤: 0 pt

c) *Message type* – denotes the type of the message. The value shall be set to 0x27 (see Table 2);

d) *Length* – denotes the length of the mTERMNOTI control message including control data (in bytes);

e) *Session ID* – denotes a 64-bit value of Session ID as defined in 9.1.1;

f) *MAID* – denotes the MAID of the sender of the message. Its value shall contain the local IP address and port number as defined in 9.1.2;

g) *Control data* – may include the following information:

8.3.23.2 Following table shows the control data types which can be used within the mTERMNOTI message. Details of following control data are described in clause 8.4.

Control type	Meaning	M/O
REASON	A reason for terminating the session.	M

8.4 MMC-3 Control data

8.4.1 SYSINFO control data

8.4.1.1 This control data specifies the system information of MA, e.g. in/out bandwidth, controllable number of CMAs.

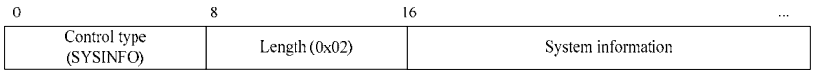


Figure 86 – Control data – SYSINFO

a) *Control type* – denotes the type of the control data. Its value shall be set to 0x08 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x02 which means 2-byte;

c) *System information* – may include the following sub-control data.

서식 있음: enumlev1

The sub-control data that may follow the SYSINFO control data are shown in Figure 87 through Figure 93.

8.4.1.2 Figure 87 shows the SI_POS_BW sub-control data format.

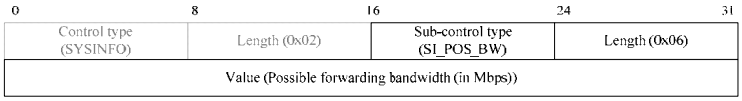


Figure 87 – Sub-control data –SI_POS_BW

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x13 (see Table 5);

b) *Length* – denotes the length of sub-control data. The value shall be set to 0x06 which means 6-byte;

c) *Value* – shall be set to the possible forwarding bandwidth that MA can offer.

8.4.1.3 When notifying non-HMAs in the same multicast area of its system information, HMA may include system information of MA, such as in-and-out bandwidth, and number of controllable CMAs. HMA may also include additional information such as Local IP and HMA lifetime to recover from the HANNOUNCE collision.

Figure 88 shows the SI_IP sub-control data. Each field has the following meaning and value:

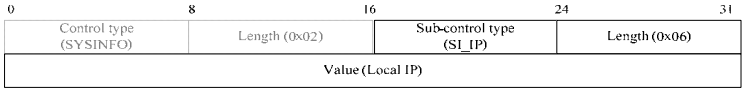


Figure 88 - Sub-control data – SI_IP

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x11 (see Table 5);

b) *Length* – denotes the length of sub-control data. The value shall be set to 0x06 which means 6-byte;

c) *Value* – shall be set to the IP address of local host.

8.4.1.4 The SI_UPTIME sub-control data is shown in Figure 89. It can be used as the lifetime of HMA or the report on the system uptime since MA joined the session. Each field has the following meaning and value:

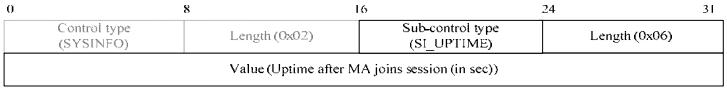


Figure 89 – Sub-control data – SI_UPTIME

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x12 (see Table 5);

b) *Length* – denotes the length of sub-control data. The value shall be set to 0x06 which means 6-byte;

c) *Value* – shall be set to the time after the node joins the MMC-3 session (in second).

8.4.1.5 Figure 90 shows the SI_ROOM_CMA sub-control data. It can be used to report the room for CMAs. Each field has the following meaning and value:

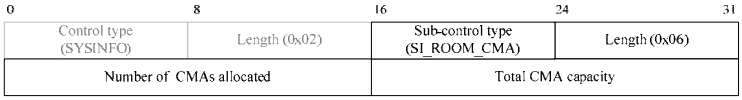


Figure 90 – Sub-control data – SI_ROOM_CMA

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x41 (see Table 5);

b) *Length* – denotes the length of sub-control data. The value shall be set to 0x06 which means 6-byes;

c) *Number of CMAs allocated* – shall be set to the number of allocated rooms for the CMAs;

d) *Total CMA capacity* – shall be set to the total CMA capacity. So the available number of rooms for CMA will be the difference between the number of CMAs allocated and total CMA capacity.

8.4.1.6 Figure 91 shows the report on the bandwidth that can be provided by a system. Each field has the following meaning and value:

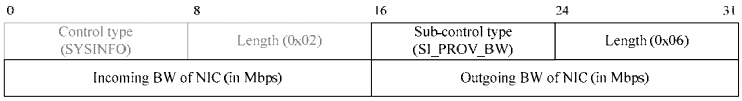


Figure 91 – Sub-control data – SI_PROV_BW

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x14 (see Table 5);

b) *Length* – denotes the length of sub-control data. The value shall be set to 0x06 which means 6-byte;

c) *Incoming BW of NIC* – shall be set to the maximum incoming bandwidth of network interface card;

d) *Outgoing BW of NIC* – shall be set to the maximum outgoing bandwidth of network interface card.

8.4.1.7 Figure 92 shows the report on the status of tree PMA and CMAs of MA. Each field has the following meaning:

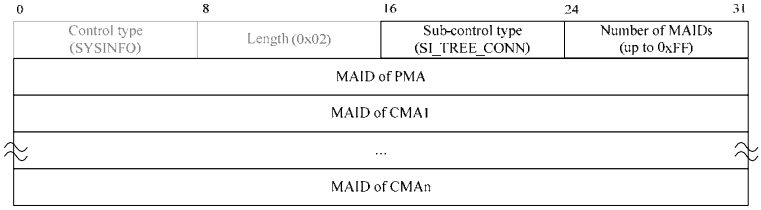


Figure 92 – Sub-control data – SI_TREE_CONN

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x51 (see Table 5);

b) *Number of MAIDs* – denotes the the number of MAIDs in the list. The value shall be set to n+1 in hexadecimal. Since the length of this field is 8-bit, maximum value of this field is 0xFF which means that 255 MAIDs, one for PMA and 254 MAIDs for CMAs, are included in the SI_TREE_CONN sub-control data;

c) *MAID of PMA* – shall be set to the MAID of directly attached PMA;

d) *MAID of CMA n* – shall be set to the MAID of n-th directly attached CMA.

8.4.1.8 Figure 93 shows the report on the member of tree. Each field has the following meaning:

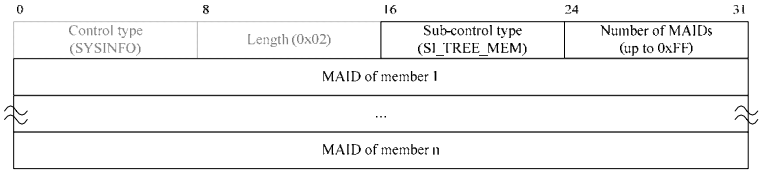


Figure 93 – Sub-control data – SI_TREE_MEM

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x69 (see Table 5);

b) *Number of MAIDs* – denotes the number of MAIDs in the list. The value shall be set to n in hexadecimal. Since the length of this field is 8-bit, maximum value of this field is 0xFF which means that there are 255 members in the tree;

c) *MAID of member n* – shall be set to the MAID of n-th tree member.

8.4.1.9 Figure 94 shows the report on the managing MNs of MMA. This sub-control data is used when a MMA reports list of its managing MNs. The managing MNs include both MNs in the MMA region and MNs in the non-MMA region. Each field has the following meaning:

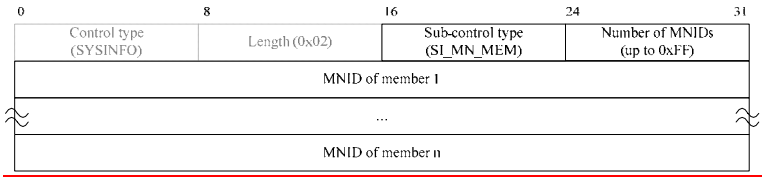


Figure 94 – Sub-control data – SI_MN_MEM

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x79 (see Table 5);

b) *Number of MNIDs* – denotes the number of MNIDs in the list. The value shall be set to n in hexadecimal. Since the length of this field is 8-bit, maximum value of this field is 0xFF which means that the MMA manages 255 MNs for each session;

c) *MNID of member n* – shall be set to the MNID of n-th member managed by a specific MMA.

서식 있음: 스타일 제목 4 + 10 pt 굵게 없음1

서식 있음: Figure

서식 있음: Figure_No & title

8.4.2 DATAPROFILE control data

8.4.2.1 DATAPROFILE control data delivers the controllable data profile of each MA. DATAPROFILE control data may be used within control messages for negotiating data channel.

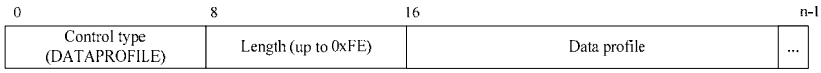


Figure 95 – Control data – DATAPROFILE

Each field has the following meaning and information:

a) *Control type* – denotes the type of control data. Its value shall be set to 0x03 (see Table 4);

b) *Length* – denotes the length of the control data. The value shall be set to n/8 in hexadecimal which means the total length of the DATAPROFILE control data in byte. Since the length of this field is 8-bit, maximum value of this field is 0xFF which means the length of the DATAPROFILE control data is 255-byte including 253-byte of the "Data profile" field. But, since the sum of the length of the Data profile field and the length of padding field is aligned to multiple of 4-byte, maximum value of the Length field can be 0xFE;

c) *Data profile* – denotes the data profile that MA wants to use. Data profile is the description of the characteristics of the data channel. It follows the SDL-like encoding scheme;

d) *Zero or more padding* – Since Data profile consists of a text-based variable message, the size may vary. To align a length of 4 bytes, each data profile pads zero or more 1-byte zero padding as shown in Figure 95.

8.4.3 AUTH control data

8.4.3.1 Authentication information is delivered using AUTH control data. The authentication algorithm used is defined in AUTH_ALG field. AUTH_ALG code is defined in Table 9.

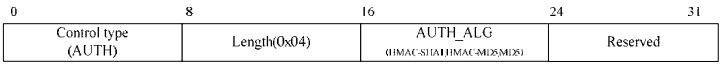


Figure 96 – Control data – AUTH

Each field has the following meaning and information:

a) *Control type* – denotes the type of control data. Its value shall be set to 0x01 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x04 which means 4-byte;

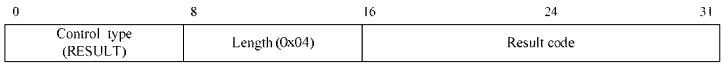
c) *AUTH_ALG* – denotes the type of authentication algorithm. The value shall be set to

- the coded value for HMAC-SHA1 in Table 9;
- the coded value for HMAC-MD5 in Table 9;
- the coded value for MD5 in Table 9;

d) *Reserved* – reserved for the futher use.

8.4.4 RESULT control data

8.4.4.1 This control message specifies whether MA's request is successful or not. If MA's request is successful, the OK code is includes within the Result code field. Otherwise, an appropriate error code is given. Figure 97 shows the format of RESULT control data.



삭제됨: control data

Figure 97 – Control data – RESULT

Each field has the following meaning and value:

a) *Control type* – denotes the type of control data. Its value shall be set to 0x05 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x04 which means 4-byte;

c) *Result code* – denotes the result of the request. The codes and their meaning are listed in Table 13.

8.4.5 NEIGHBORLIST control data

8.4.5.1 If a subscription is successful, SM gives neighbor lists which include MAID of active MAs back to the subscriber. The NEIGHBORLIST control data can be used as bootstrap information by each subscriber. Figure 98 shows the format of NEIGHBORLIST, note that it only delivers MAID.

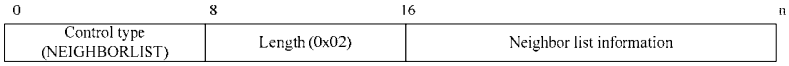


Figure 98 – Control data – NEIGHBORLIST

a) *Control type* – denotes the type of control data. Its value shall be set to 0x04 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x02 which means 2-byte;

c) *Neighbor list information* – denotes the series of information on MAIDs. The following are the usage and format:

8.4.5.2 Figure 99 shows the sub-control data that follows NEIGHBORLIST control data. Each field has the following meaning and value:

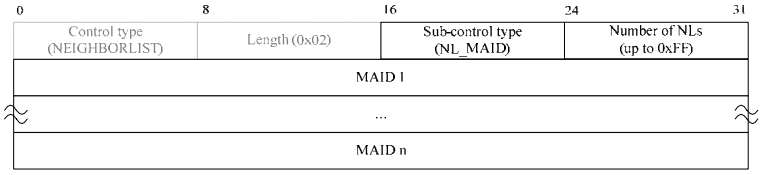


Figure 99 – Sub-control data – NL_MAID

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x01 (see Table 6);

b) *Number of NLs* – denotes the number of MAIDs. Since the length of the Number of NLs field is 8-bits, maximum value of this field is 0xFF which means that 255 MAID of neighbors are listed in the NEIGHBOTLIST control data;

c) *MAID n* – shell be set to the MAID of n-th neighbor.

8.4.6 ROOTPATH control data

8.4.6.1 To prevent loop and solve the triangular problem, the probed MA must include its rootpath using ROOTPATH control data shown in Figure 100. Each field has the following meaning and value:

삭제됨:

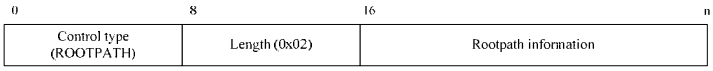


Figure 100 – Control data – ROOTPATH

a) *Control type* – denotes the type of control data. Its value shall be set to 0x07 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x02 which means 2-byte;

c) *Rootpath information* – includes rootpath information. The following are the format and usage:

8.4.6.2 Figure 101 shows the sub-control data of ROOTPATH control data.

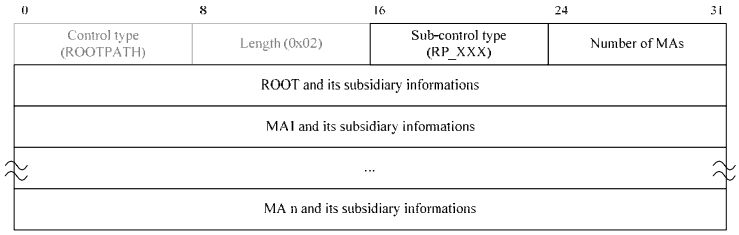


Figure 101 – Sub-control data – RP_XXX

Each field has the following meaning and value:

서식 있음: enumlev1

a) *Sub-control type* – denotes the type of sub-control data. The codes and their meaning are listed in Table 7;

b) *Number of MAs* – denotes the number of MAs on the rootpath;

c) *One or more information* – The information about hop according to sub-control type. The size of each field is fixed and can be calculated by combination of each type length.

8.4.7 TIMESTAMP control data

8.4.7.1 Figure 102 shows the TIMESTAMP control data used to examine the distance between two MAs.

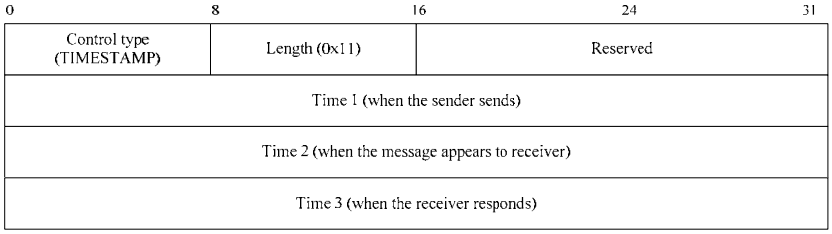


Figure 102 – Control data – TIMESTAMP

Each field has following meaning and value:

a) *Control type* – denotes the type of control data. Its value shall be set to 0x09 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x10 which means 16-byte.;

c) *Reserved* – Reserved for the further use;

d) *Time1* – shall be set to the time when the message is sent to its counterpart;

e) *Time2* – shall be set to the time when the message appears to the counterpart;

f) *Time3* – shall be set to the time when the receiver of the message sends the TIMESTAMP control data in response.

8.4.8 REASON control data

8.4.8.1 To specify the reason for leaving of MA, the LEAVREQ/HLEAVE control message must include REASON control data. TERMREQ control message must include REASON control data to specify the reason for terminating the session. Figure 103 shows the REASON control data format.

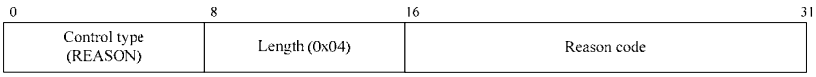


Figure 103 – Control data – REASON

Each field has following meaning and value:

a) *Control type* – denotes the type of control data. Its value shall be set to 0x05 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x04 which means 4-byte;

c) *Reason code* – shall be set to an integer value to indicate the specific reason for leaving. The encoded value and its meaning follow the codes specified in Table 10 and Table 11.

8.4.9 RP COMMAND control data

8.4.9.1 RP COMMAND control data is used to request the rootpath information of MA. RELREQ control message can include RP COMMAND control data to specify what type of rootpath it needs.

Figure 104 shows the RP COMMAND control data format.

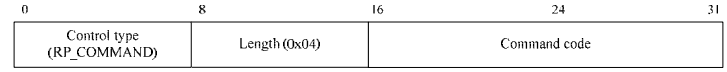


Figure 104 – Control data – RP COMMAND

a) *Control type* – denotes the type of control data. Its value shall be set to 0x01 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x04 which means 4-byte;

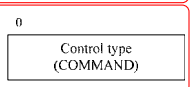
삭제됨: specific

삭제됨: The STREQ

삭제됨: should

삭제됨: status report

삭제됨:



삭제됨: 2

c) *Command code* – shall be set to the value to indicate the specific rootpath information. The encoded value and its meaning are same as specified in Table 7.

삭제됨: command

삭제됨: Table 5

삭제됨: If SM wants to know IP address of MA, SM sends STREQ message with COMMAND control data which includes SI_IP as a command code.

8.4.10 SI_COMMAND control data

8.4.10.1 SI_COMMAND control data is used to request the specific information of MA. STREQ control message should include SI_COMMAND control data to specify what status information it needs.

Figure 105 shows the SI_COMMAND control data format.

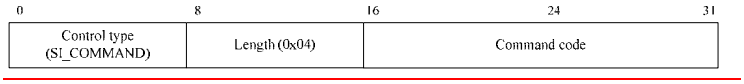


Figure 105 – Control data – SI_COMMAND

a) *Control type* – denotes the type of control data. Its value shall be set to 0x02 (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x04 which means 4-byte;

c) *Command code* – shall be set to the value to indicate the specific system information. The encoded value and its meaning are same as specified in Table 5. If SM wants to know IP address of MA, SM sends STREQ message with SI_COMMAND control data which includes SI_IP as a command code.

8.4.11 TREEEXPLOR control data

8.4.11.1 Inspecting whole tree status can cause hazards because of report implosion. So it is very important to limit the scope of tree to be inspected. Figure 106 shows TREEEXPLOR control data which is used to limit the scope of the tree.

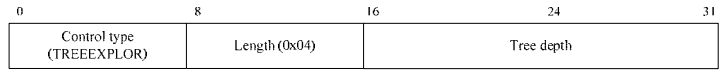


Figure 106 – Control data – TREEEXPLOR

a) *Control type* – denotes the type of control data. Its value shall be set to 0x0B (see Table 4);

b) *Length* – denotes the length of control data. The value shall be set to 0x04 which means 4-byte;

c) *Tree depth* – shall be set to the value to specify the scope of tree inspection.

8.4.12 GROUPINFO control data

8.4.12.1 This control data is used to provide group information to MNs.

서식 있음: 글꼴: 굵게 없음

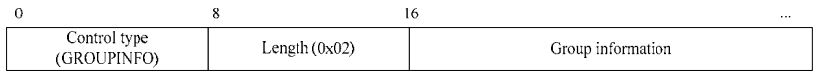


Figure 107 – Control data – GROUPINFO

a) *Control type* – denotes the type of control data. Its value shall be set to 0x0C (see Table 4);

b) *Length* – denotes the total length of control data. The value shall be set to 0x02 which means 2-byte;

c) *Group information* – denotes the information of serving group. Group information may include the following sub-control data:

8.4.12.2 To describe session information, GI_SVC sub-control data is used. The GI_SVC sub-control data are shown in Figure 108. The sub-control data that may follow the GROUPINFO control data format shown in Figure 107.

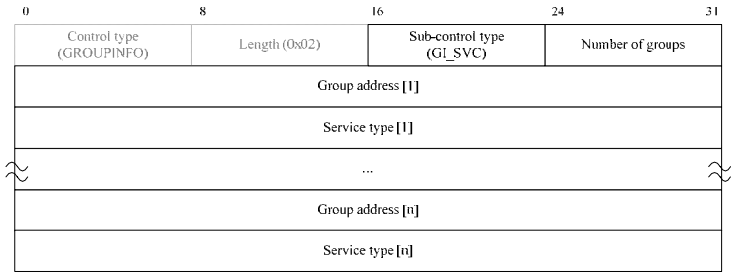


Figure 108 – Sub-control data – GI_SVC

Each field has the following meaning and value:

a) *Sub-control type* – denotes the type of sub-control data. Its value shall be set to 0x01 (see Table 8);

b) *Number of groups* – denotes the number of groups in service;

c) *Group address [n]* – shall be set to the n-th group address in service;

d) *Service type [n]* – shall be set to the n-th group information (stored in character).

삭제됨: Table 8 shows sub-control data types for GROUPINFO control data. Table 8 Table 8 shows sub-control data types for GROUPINFO control data. Table 8

8.4.13 LIFETIME control data

8.4.13.1 This control data is used to register time between terminal and MA. And this control message is used for mADVERTISE and mREGISTREQ messages.

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게 없음

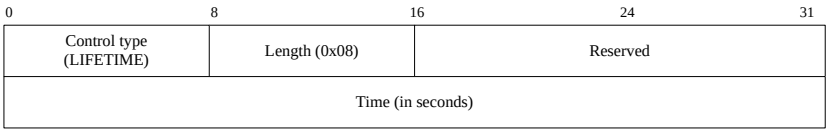


Figure 109 – Control data – LIFETIME

a) *Control type* – denotes the type of control data. Its value shall be set to 0x0D (see Table 4);

b) *Length* – denotes the total length of the control data. The value shall be set to 0x08 which means 8-byte;

c) *Reserved* – Reserved for the further use;

d) *Time* – denotes maximum register time or request time (in second).

8.4.14 OMAINFO control data

8.4.14.1 This control data is used to request its leave through new MMA to old MMA and this control data is used for mREGISTREQ message. Each field has following meaning and value:

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게 없음

서식 있음: 글꼴: 굵게 없음

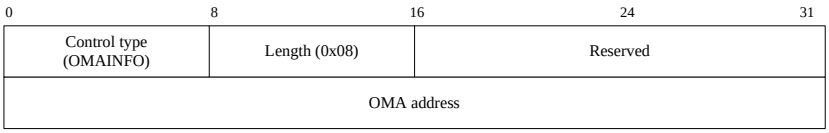


Figure 110 – Control data – OMAINFO

a) *Control type* – denotes the type of control data. Its value shall be set to 0x0E (see Table 4);

b) *Length* – denotes the total length of the control data. The value shall be set to 0x08 which means 8-byte;

c) *Reserved* – Reserved for the further use;

c) *OMA Address* – shall be set to the IP address of previous MMA which had served MN.

8.4.15 IDENTIFICATION control data

8.4.15.1 This control data is used to request its registration and to acknowledge message authentication and this control message is used for mREGISTREQ and mREGISTANS messages.

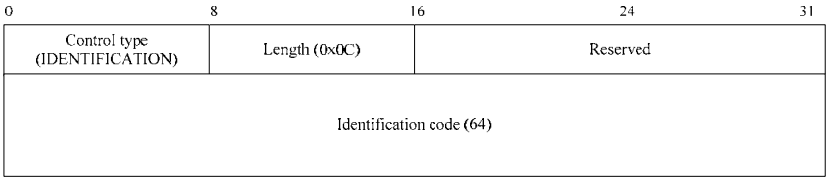


Figure 111 – Control data – IDENTIFICATION

a) *Control type* – denotes the type of control data. Its value shall be set to 0x0F (see Table 4);

b) *Length* – denotes the total length of the control data. Its value shall be set to 0x0C which mean 12-byte;

c) *Identification Code* – shall be set to a 64-bit identification code that identifies a message.

8.4.16 CTRLCH control data

8.4.16.1 This control data is to give information on channel for control. And this control message is used for mADVERTISE message.

서식 있음: 글꼴: 굵게 없음

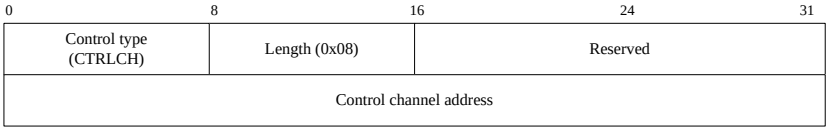


Figure 112 – Control data – CTRLCH

a) *Control type* – denotes the type of control data. Its value shall be set to 0x11 (see Table 4);

b) *Length* – denotes the total length of the control data. Its value shall be set to 0x08 which means 8-byte;

c) *Control channel address* – shall be set to the 32-bit Class D group address for control channel.

8.4.17 DATACH control data

8.4.17.1 DATACH control data is to give information on channel for data and is used for mREGISTANS message.

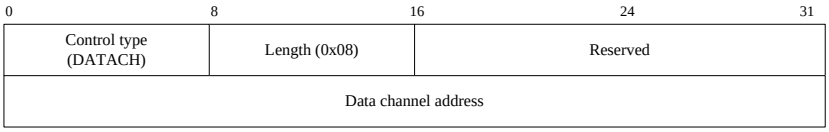


Figure 113 – Control data – DATACH

a) *Control type* – denotes the type of control data. Its value shall be set to 0x12 (see Table 4);

b) *Length* – denotes the total length of the control data. Its value shall be set to 0x08 which means 8-byte;

c) *Data channel address* – shell be set to the 32-bit Class D group dataa for data channel.

8.4.18 URGENTJOIN control data

8.4.18.1 This control data is used by new MMA to attach to MN's old MMA for seamless handover. This type of join is necessary for MMA to save time from finding PMA and this control message is used for RELREQ message.

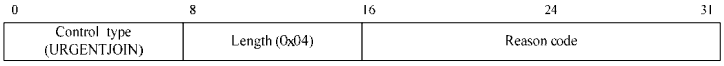


Figure 114 – Control data – URGENTJOIN

a) *Control type* – denotes the type of control data. Its value shall be set to 0x13 (see Table 4);

b) *Length* – denotes the total length of the control data. Its value shall be set to 0x04 which means 4-byte;

c) *Reason code* – shall be set to the reason code that indicates urgent join (see Table 12).

8.4.19 MNLEAVE control data

8.4.19.1 This control data is used by new MMA to tell its MN's leave to the leaving MN's old MMA. The MNLEAVE control data is necessary when the MN performs handover. It is necessary to delete the state information of the leaving MN for preventing useless broadcasting when there is no MN in the MMA region. The MNLEAVE control data is used for RELREQ control message.

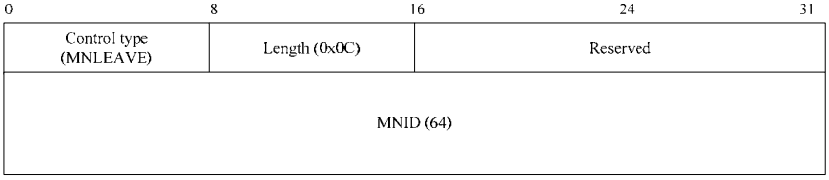


Figure 115 – Control data – MNLEAVE

a) *Control type* – denotes the type of control data. Its value shall be set to 0x14 (see Table 4);

b) *Length* – denotes the total length of the control data. Its value shall be set to 0x0C which means 12-byte;

c) *MNID* – shall be set to the MNID of leave requested MN.

9 Parameters

9.1 MMC-3 identifiers

9.1.1 Session ID

Session ID (SID) is generated with a combination of the local IP address of the Session Manager (SM) and the group address of the session. The SM allocates a group address to a new session when it is requested to create a session. The group address is created as a unique value for the session without duplication with any session it manages.

Figure 116 illustrates the format of SID in MMC-3.

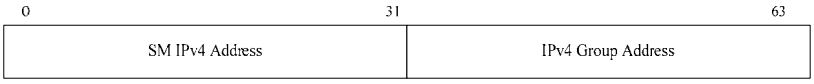


Figure 116 – Format of Session ID

9.1.2 MAID and MNID

MAID consists of the local IP address, port number, and serial number as Figure 117 shows. The local IP address is the IP address of MA. An MA in a MMC-3 session may have to open several ports for the session. The port number used for generation of its MAID is a listening port number opened when the MA starts to run MMC-3 in order to receive control messages from SM or other MAs.

Each MA can be identified by its port number in a multi-user system. It is, however, not possible to identify each MA inside of a Network Address Translation (NAT) based network, where it may show the same IP address for multiple MAs to the communication peer outside of the network. To handle this case, SM generates a unique MAID as it fills in a unique value in the serial number field when it receives a NAT address from an MA, and returns the ID to the MA.

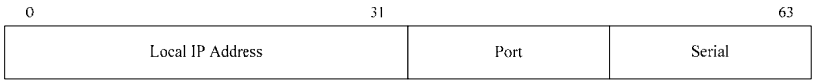


Figure 117 – Format of MAID

The following Figure 118 is the simple algorithm that the current version of MMC-3 uses to generate a unique MAID.

```
If the IP address in the received MAID is a NAT address
    Search for its NAT_address_list;
    if there already exists the same address
            serial_number++;
        else
            add the list into NAT_address_list
            serial_number++;
    MAID = IP_address + port_number + serial_number;
return MAID;
```

Figure 118 – Simple algorithm to generate a unique MAID

Actually MNID is same as MAID. The only difference between MNID and MAID is the user of ID. The MNID is used by Mobile node; on the other hand the MAID is used by Multicast agent. To distinguish Mobilde node from Multicast agent, MMC-3 uses two different terminologies for identification of Mobile node and Multicast agent.

9.2 Parameters used in MMC-3

This section describes the encoding rules for MMC-3 and covers the following:

a) MMC-3 control message types
b) MMC-3 node types
c) Control data type
d) Sub-control data type

9.2.1 MMC-3 control message types

Table 2 lists the types of MMC-3 control message and the corresponding encoded values.

Table 2 – MMC-3 control message types

Message type	Code (8 bits)
SUBSREQ	0x01
SUBSANS	0x02
PPROBREQ	0x03

삭제됨: 0x02
삭제됨: 0x03
삭제됨: 0x04

PPROBANS	0x04
HSOLICIT	0x05
HANNOUNCE	0x06
HLEAVE	0x07
RELREQ	0x08
RELANS	0x09
STREQ	0x0A
STANS	0x0B
STCOLREQ	0x1A
STCOLANS	0x1B
LEAVREQ	0x0C
LEAVANS	0x0D
HB	0x10
TERMREQ	0x0E
TERMANS	0x0F
FAILCHECK	0x1C
mADVERTISE	0x21
mSOLICIT	0x22
mREGISTREQ	0x23
mREGISTANS	0x24
mLEAVREQ	0x25
mLEAVANS	0x26
mTERMNOTI	0x27

삭제됨: 0x05
삭제됨: 0x06
삭제됨: 0x07
삭제됨: 0x08
삭제됨: 0x09
삭제됨: 0x0C
삭제됨: 0x12
삭제됨: 0x13
삭제됨: 4
삭제됨: 5
삭제됨: 0x16
삭제됨: 0x17
삭제됨: 8
삭제됨: 0x19
삭제됨: 1A
삭제됨: TREEEXPLOR... [9]

9.2.2 Node types

Table 3 lists the MMC-3 nodes and corresponding encoded values.

Table 3 – MMC-3 node types

Node type	Code (4 bits)
SM	0x1
MCS	0x2
MA	0x3
MMA	0x4
MN	0x6

삭제됨: 8
삭제됨: 0
삭제됨: SMA
삭제됨: 0
삭제됨: 0
삭제됨: 0
삭제됨: 0

9.2.3 Control data types

Table 4 lists the codes of MMC-3 control data type.

Table 4 – MMC-3 control data types

Control type	Code (8 bits)
AUTH	0x00
RP_COMMAND	0x01
SI_COMMAND	0x02
DATAPROFILE	0x03
NEIGHBORLIST	0x04
REASON	0x05
RESULT	0x06
ROOTPATH	0x07
SYSINFO	0x08
TIMESTAMP	0x09
TREEEXPLOR	0x0B
GROUPINFO	0x0C
LIFETIME	0x0D
OMAINFO	0x0E
IDENTIFICATION	0x0F
DATACH	0x11
CTRLCH	0x12
URGENTJOIN	0x13
MNLEAVE	0x14

삭제됨: 1

삭제됨: 2

삭제됨: CANDIDA... [10]

9.2.4 Sub-control data types

A single control data may include zero or more sub-control data. This section presents the codes of MMC-3 sub-control data.

SYSINFO control data is used for describing information related to MA. Table 5 lists the possible sub-control data type and its encoded value and meaning. The four most significant bits of the encoded code specify the category of the information, with the lowest four bits specifying the detailed items such as bandwidth, packets, and bytes.

Since combinational command code for a certain sub-control data type is not used, 16-bit command code can only represent 16-type. The combinational command code is used when multiple of system informations are requested. For example, command code 0x00 0D is used to request SI_DELAY and SI_ROOM_CMA.

Table 5 – MMC-3 sub-control data types (SYSINFO)

Type	Code (8 bit)	Command code (16 bits)	Meaning
SI_IP	0x11	0x00 01	IP address of MA.
SI_UPTIME	0x12	0x00 02	Time of MA's uptime.
SI_DELAY	0x13	0x00 04	Status of delay as perceived by MA from ROOT.
SI_ROOM_CMA	0x14	0x00 08	The room for CMAs.

서식 있는 표

SI_PROV_BW	0x15	0x00 10	Maximum incoming / outgoing bandwidth of MA's network interface card.
SI_POS_BW	0x25	0x00 20	The possible forwarding bandwidth that MA can afford.
SI_SND _BW	0x35	0x00 40	Total bandwidth consumed by PMA to serve its CMAs.
SI_SND_PACKET	0x36	0x00 80	Total number of packets sent by MA from startup.
SI_SND _BYTES	0x37	0x01 00	Total number of bytes sent by MA from startup.
SI_RCV_BW	0x45	0x02 00	Bandwidth perceived by MA between its PMA.
SI_RCV_PACKET	0x46	0x04 00	Number of packets received by MA from startup.
SI_RCV_BYTES	0x47	0x08 00	Number of bytes received by MA from startup.
SI_TREE_CONN	0x68	0x10 00	PMA and CMA(s) of MA.
SI_TREE_MEM	0x69	0x20 00	List of tree members.
SI_MN_MEM	0x79	0x40 00	List of MNs which are managed by a specific MMA.

삭제됨: SI_PERC_[illegible] … [11]

NEIGHBORLIST control data is used for describing information related to the MMC-3 neighbors. Table 6 lists the possible sub-control data type and its encoded value and meaning.

Table 6 – MMC-3 sub-control data types (NEIGHBORLIST)

Sub-control Type	Code	Meaning
NL_MAID	1	List of MAs.

ROOTPATH control data is used to describe the path between two end-points. The path consists of MAs passing through between the two end-points along the HybridTree. Table 7 lists the possible sub-control data type and their encoded value and meaning.

삭제됨:

서식 있음: 글꼴: 기울임꼴

Table 7 – MMC-3 sub-control data types (ROOTPATH)

서식 있는 표

Type	Code (8 bits)	Command code (16 bits)	Meaning
RP_ID	0x11	0x00 01	The following ROOTPATH contains only the MAID of each hop (8 bytes each).
RP_ BW	0x12	0x00 02	The following ROOTPATH contains only the bandwidth by hop (4 bytes each).
RP_ DL	0x14	0x00 04	The following ROOTPATH contains only the delay perceived by each hop (4 bytes each).
RP_ID _BW	0x13	0x00 03	The following ROOTPATH contains the MAID and bandwidth of each hop (12 bytes each).
RP_ID _DL	0x15	0x00 05	The following ROOTPATH contains the MAID and corresponding delay of each hop (12 bytes each).
RP_ID_BW_DL	0x17	0x00 07	The followed ROOTPATH contains the MAID, bandwidth, and delay of each hop (16 bytes each).
RP_PSEUDO	0x10	0x00 08	The following ROOTPATH is a pseudo-ROOTPATH for fault recovery (N/A).

삭제됨: 8 bytes each

Table 8 shows sub-control data types for GROUPINFO control data.

Table 8 – MMC-3 sub-control data types (GROUPINFO)

Sub-control Type	Code	Meaning
GI_SVC	0x01	Information about serving group

9.3 Encoding rules to represent values used in MMC-3

9.3.1 Authentication algorithm for MMC-3

AUTH control data is used to specify the authentication algorithm to be used. Table 9 lists the possible authentication algorithms for MMC-3 and their encoded value and reference.

Table 9 – Authentication algorithm (AUTH_ALG)

Type	Code (8 bits)	Reference
HMAC-SHA1	0x01	IETF RFC 2104
HMAC-MD5	0x02	IETF RFC 2104
MD5	0x03	IETF RFC 1321

9.3.2 Reason for leaving

Table 10 lists the various reasons for leaving of MA. The eight most upper bits specify the main cause of leaving, with the eight lowest bits specifying the detailed reasons for leaving. Through the code for the reason for leaving, MA can express the reason for leaving explicitly.

삭제됨: four

삭제됨: four

Table 10 – Code for reason for leaving

Category	Value (16 bits)	Meaning
Leave	0x01 00	MA's own leaving
	0x02 00	MCS leaving
Kick out	0x03 00	SM kick out
	0x03 01	PMA kick out
Parent switching	0x04 00	Parent switching by MA

삭제됨: 8

삭제됨: 1

삭제됨: 1

삭제됨: 1

삭제됨: SMA

삭제됨: 2

삭제됨: 2

삭제됨: 4

9.3.3 Reason for termination

Table 11 lists the reason for session termination. The eight most significant bits specify the main reason for session termination, with the eight lowest bits specifying the reasons.

삭제됨: four

삭제됨: four

Table 11 – Code for reason for termination

Category	Value (16 bits)	Meaning
Normal session termination	0xF1 00	Session is terminated normally.
Abnormal session termination	0xF2 00	Session is terminated abnormally for no reason.
	0xF2 01	Session is terminated abnormally by user request.

서식 있는 표

삭제됨: 8

삭제됨: E0

삭제됨: F0

삭제됨: F1

9.3.4 Reason for urgent join

Table 12 lists the reason for urgent join.

Table 12 – Code for reason for urgent join

Category	Value (16 bits)	Meaning
Urgent join for handover	0xA0 00	Urgent join is needed because of MN handover

삭제됨: 8
서식 있는 표
삭제됨: F2

9.3.5 Result code

Table 13 lists the results. These codes are included in the return message to specify the result of a specific request.

Table 13 – Result codes

Value (16 bits)	Meaning
0x01 00	OK
0x02 00	System problem
0x03 00	Administrative problem

삭제됨: 8

9.4 Timers and their parameters

9.4.1 Parameters for MMA advertisement

Table 14 – Parameters for MMA advertisement

Value	Name	Default value	Description
T_MMAADV	MMA advertisement timer	5	This timer should be kept by each MMA to issue a periodic mADVERTISE control message. At every assigned time period, it reminds MMA to issue an mADVERTISE control message. The default value for the MMA advertisement timer is 5 seconds, although it can be changed arbitrarily be each MMA.

9.4.2 Parameters for connection continuity

Table 15 – Parameters for connection continuity

Value	Name	Default value	Description
T_REGIST	Regist timer	10	This timer should be kept by each MMA to recognize of MN leave. If MMA does not receive mREGISTREQ message from specific MN until Regist timer is expired, MMA considers that the MN leaved the session. The default value for the Regist timer is 10 seconds, although it can be changed arbitrarily be each MMA.

Editor's note: Definition of threshold value for parent switching should be added.

서식 있음: 글꼴: 11 pt, (한글) 한국어
서식 있음: 표준
서식 있음: (한글) 한국어
서식 있음: 글꼴: 기울임꼴
서식 있음: 글꼴: 기울임꼴, (한글) 한국어
서식 있음: 글꼴: 기울임꼴
서식 있음: 글꼴: 기울임꼴
서식 있음: 글꼴: 기울임꼴, (한글) 한국어
서식 있음: 글꼴: 기울임꼴, (한글) 한국어

9.5 MMC data profile

<TBD>

Attachment – List of open issues in June 2009

This Attachment describes the list of open issues identified by the editors.

Priority H: Serious problem, M: Medium level problem, L: low level problem (mainly editorial issues)

Issue No.	Priority	Document clause number	Open issues	Work needed	action proposal
1	H	7.3.3 & 7.4.5	Handover	Need to define precise procedure & protocol for handovers	Contributions are invited
2	M	6.5 & 9.5	Real-time data delivery procedure	Need to define precise procedure & protocol for real-time data delivery	Contributions are invited
3	M	6.5 & 9.5	Reliable data delivery procedure	Need to define precise procedure & protocol for reliable data delivery	Contributions are invited
4	H	9.5	MMC data profile	Need to define data profile for data delievery	Contributions are invited

페이지 5: [1] 삭제됨 Lee 2009-04-23 PM 1:53:00

페이지 24: [2] 서식 있음	Lee	2009-04-24 PM 2:33:00

제목 5, 다음 단락과의 사이에 페이지 나누기, 현재 단락을 나눔

페이지 24: [3] 삭제됨	Lee	2009-04-28 PM 2:25:00

where the multicast data delivery scheme is used

페이지 24: [4] 삭제됨	Lee	2009-05-08 PM 3:05:00

with an empty HMA candidate list to the local network

페이지 32: [5] 삭제됨	Lee	2009-05-08 PM 7:49:00

This clause only describes mobility support function for MN. The other functions are equivalent to clause 7.2.

페이지 32: [6] 삭제됨	Lee	2009-05-04 PM 8:11:00

can exist anywhere in the network

페이지 32: [7] 삭제됨	Lee	2009-05-04 PM 8:27:00

The mADVERTISE message is broadcasted every MMA advertisement time so that the MN would be aware of the MMA connectivity and join the multicast session anytime.

페이지 32: [8] 삭제됨 Lee 2009-05-04 PM 9:25:00

which belongs to the MMA

페이지 80: [9] 삭제됨 Lee 2009-05-26 PM 3:15:00

TREEEXPLOR	0x1B

페이지 81: [10] 삭제됨 Lee 2009-05-26 AM 11:37:00

CANDIDATEHMA	0x0A

페이지 82: [11] 삭제됨 Lee 2009-05-26 AM 11:43:00

SI_PERC_QOS	0x26		The QoS perceived by each MA