IEEE P802.21/D8.0

Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services

Sponsor

**LAN MAN Standards Committee
of the
IEEE Computer Society**

Abstract: This standard specifies IEEE 802 media access-independent mechanisms that optimize handovers between heterogeneous IEEE 802 systems and between IEEE 802 systems and cellular systems.
Keywords: media independent handover

This page intentionally left blank

Introduction

This standard defines extensible media access independent mechanisms that enable the optimization of handovers between heterogeneous IEEE 802 systems and may facilitate handovers between IEEE 802 systems and cellular systems.

Working Group Officers

At the time this supplement to the standard was submitted to Sponsor Ballot, the working group had the following officers.

Vivek Gupta, *Chair*
Michael Williams, Vice Chair
Xiaoyu Liu, Secretary
Qiaobing Xie, Technical Editor

The following individuals served as the working group Assistant Editors:

Yuu-Heng Alice Cheng
David Cypher
Junghoon Jee
Marc Meylemans

Proposals for Media Independent Handover solutions were made by the following individuals (co-authors of the accepted proposal are marked with an asterisk)

Farooq Anjum*
Stefan Berg*
Yogesh Bhatt*
Alistair Buttar*
Alan Carlton*
Subir Das*
Ashutosh Dutta*
Les Eastwood*
Natraj Ekambaram*
Stefano Faccin*
Victor Fajardo*
Peretz Feder*
Andrea Francini*
Yuri Goldstein*
Prasad Govindarajan*
Wolfgang Groeting*
Vivek Gupta*
Youn-Hee Han*
Fang Hao*
Eleanor Hepworth*
Michael Hoghooghi*
Cheng Hong*
Prakash Iyer*
David Johnston*
Maged Kaki*
Ronny Kim*
Benjamin Koh*
Kalyan Koora*
Yongwon Kwak*
Jeff Keating*
Hong-Yon Lach*
Peter Larsson*
Jin Lee*
Sungjin Lee*
Michael Lee*
Xiaoyu Liu*
Guang Lu*
Sunil Madhani*
Mani Mahalingam*
Stephen McCann*
Ray Miller*
Padraig Moran*
Nat Natarajan*
Eric Njedjou*
Yoshihiro Ohba*

Ulises Olvera-Hernandez*
Chihyun Park*
Soohong Daniel Park*
Mathieu Peresse*
Daniel Orozco-Perez*
Ajay Rajkumar*
Sampath Rangarajan*
Marian Rudolf*
Reijo Salminen*
Sameer Sharma*
Ajoy Singh*
Kenichi Taniuchi*
Yousif Targali*
Michael Williams*
Qiaobing Xie*
Raziq Yaqub*
Tao Zhang*
Yao Zhonghui*
Juan Carlos Zuniga*

Others who made presentations to the Media Independent Handover Working Group, and contributions to this document, were:

Sanjeev Athayle
Gabor Bajko
Farooq Bari
Yuu-Heng Alice Cheng
David Cypher
David Famolari
Nada Golmie
Junxiang Guo
James Han
Kentaro Ishizu
Junghoon Jee
Hongseok Jeon
Eunah Kim
Masahiro Kuroda
Jing Liu
Changmin Park
Yan Peng
Takashi Sakakura
Srinivas Sreemanthula
Lucian R. Suciu
Miriam Tauil
Albert Vidal
Jesse Walker
Bin Xia
Yi Zhou
Hongcheng Zhuang

[NOTE - Please send other names to the Editor. Need to update list and re-balance when list is done]

The following members of the balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

To be supplied by IEEE

Editor’s Foreword

[Notes]

[Throughout this document all notes such as this one are temporary notes inserted by the Editors for a variety of purposes; these notes and the Editor’s Forward will all be removed prior to publication and are not part of the normative text.]

[**Comments and participation in IEEE 802.21 standards development**

Comments on this draft are encouraged. **PLEASE NOTE: All issues related to IEEE standards presentation style, formatting, spelling, etc. are routinely handled between the IEEE 802.21 Editor and the IEEE Staff Editors prior to publication, after balloting and the process of achieving agreement on the technical content of the standard is complete.** Readers are urged to devote their valuable time and energy only to comments that materially affect either the technical content of the document or the clarity of that technical content. Comments should not simply state what is wrong, but also what might be done to fix the problem.

Full participation in the development of this draft requires individual attendance at IEEE 802.21 meetings. Information on IEEE 802.21 activities, working documents, and email distribution lists can be found on the IEEE 802.21 website:

http://ieee802.org/21

Comments on this document may be sent to the IEEE 802.21 mailing reflector, to the Editor, or to the Chairs of the IEEE 802.21 Working Group.

Vivek Gupta

Chair, IEEE 802.21 (Media Independent Handover)

Email: vivek.g.gupta@intel.com

Michael Williams

Vice Chair, IEEE 802.21 (Media Independent Handover)

Email: michael.g.williams@nokia.com

Qiaobing Xie

Technical Editor, IEEE 802.21 (Media Independent Handover)

Email: Qiaobing.Xie@motorola.com

PLEASE NOTE: Comments whose distribution is restricted in any way cannot be considered and may not be acknowledged.]

[The draft text and accompanying information

This document currently comprises:

— A temporary cover page, preceding the Editor's Forewords. This cover page will be removed following working group approval of this draft, i.e. prior to sponsor ballot.
— IEEE boilerplate text.
— The Editor's foreword, including this text.
— A title page for the propose standard including an Abstract and Keywords. This title page will be retained following approval.
— A record of participants

Scope of the Proposed Project:

This standard defines extensible 802 media access independent mechanisms that enable the optimization of handover between heterogeneous 802 systems and may facilitate handover between 802 systems and cellular systems.

Purpose of the Proposed Project:

The purpose is to improve the user experience of mobile devices by facilitating handover between 802 networks whether or not they are of different media types, including both wired and wireless, where handover is not otherwise defined and to make it possible for mobile devices to perform seamless handover where the network environment supports it. These mechanisms may also be usable for handovers between 802 networks and non 802 networks.

The joint harmonized contribution was confirmed during the May 2005 meeting and the first WG draft was produced in the July 2005 meeting.]

Table of Content

1. Introduction 1

1.1 Scope 1
1.2 Purpose 1
1.3 Overview 1
1.4 Assumptions 3
1.5 Media independence 4

2. Normative references 5

3. Definitions 7

4. Abbreviations and acronyms 10

5. General architecture 12

5.1 Introduction 12
5.1.1 General 12
5.1.2 Service continuity 12
5.1.3 Application class 12
5.1.4 Quality of service 12
5.1.5 Network discovery 13
5.1.6 Network selection 13
5.1.7 Power management 13
5.1.8 Handover policy 13
5.2 General design principles 13
5.2.1 MIHF design principles 13
5.2.2 QoS design principles 14
5.3 MIHF service overview 14
5.3.1 General 14
5.3.2 Media Independent event service 15
5.3.2.1 General 15
5.3.2.2 Event origination 15
5.3.2.3 Event destination 15
5.3.2.4 Event service flow 15
5.3.2.5 Event service use cases and functions 15
5.3.3 Media Independent command service 15
5.3.3.1 General 15
5.3.3.2 Command origination 15
5.3.3.3 Command destination 16
5.3.3.4 Command flow 16
5.3.3.5 Command service use cases 16
5.3.4 Media Independent information service 16
5.4 Media independent handover reference framework 17
5.4.1 General 17
5.4.2 MIHF communication model 18
5.4.3 A Deployment example for the MIH services 20
5.5 MIHF reference models for link-layer technologies 21
5.5.1 General 21
5.5.2 MIHF reference model for IEEE 802.3 24
5.5.3 MIHF reference model for IEEE 802.11 24

5.5.4 MIHF reference model for IEEE 802.16 25
5.5.5 MIHF reference model for 3GPP.... 26
5.5.6 MIHF reference model for 3GPP2.... 27
5.6 Service access points (SAPs).... 28
5.6.1 General.... 28
5.6.2 Media dependent SAPs.... 29
5.6.2.1 General.... 29
5.6.2.2 MIH_LINK_SAP.... 29
5.6.2.3 MIH_NET_SAP.... 30
5.6.2.4 MLME_SAP.... 30
5.6.2.5 C_SAP.... 30
5.6.2.6 MIH_3GLINK_SAP.... 30
5.6.2.7 LSAP.... 30
5.6.2.8 CS_SAP.... 30
5.6.3 Media independent SAPs.... 30
5.6.3.1 MIH_SAP.... 30
5.7 MIH protocol.... 31
5.7.1 General.... 31
5.7.2 Transport Considerations.... 31

6. MIHF services.... 32

6.1 General.... 32
6.2 Service management.... 32
6.2.1 General.... 32
6.2.2 Service management primitives.... 32
6.2.3 MIH capability discovery.... 32
6.2.4 MIH registration.... 33
6.2.5 MIH event subscription.... 34
6.2.6 Network communication.... 34
6.3 Media independent event service.... 34
6.3.1 Introduction.... 34
6.3.2 Event subscription.... 36
6.3.2.1 General.... 36
6.3.2.2 Link events subscription.... 36
6.3.2.3 MIH events subscription.... 36
6.3.3 Event service flow model.... 37
6.3.4 Link events.... 37
6.3.4.1 General.... 37
6.3.4.2 Characteristics of events.... 38
6.3.5 MIH events.... 39
6.4 Media independent command service.... 39
6.4.1 Introduction.... 39
6.4.2 Command service flow model.... 42
6.4.3 Command list.... 43
6.4.3.1 MIH commands.... 43
6.4.3.2 Link commands.... 44
6.5 Media independent information service.... 44
6.5.1 Introduction.... 44
6.5.2 Access information service before authentication.... 45
6.5.3 Restricting query response size.... 46
6.5.4 Information elements.... 46
6.5.5 Definition of information element name space.... 49
6.5.6 Information element representation and query methods.... 50

6.5.6.1 Introduction 50
6.5.6.2 Binary representation and TLV query 50
6.5.6.3 RDF representation and SPARQL query 54
6.5.7 Information service schema 54
6.5.7.1 General 54
6.5.7.2 The MIIS RDF schema 54
6.5.8 Information service flow 55

7. Service access points (SAPs) and primitives 57

7.1 Introduction 57
7.2 SAPs 57
7.2.1 General 57
7.2.2 Media dependent SAPs 58
7.2.2.1 MIH_LINK_SAP 58
7.2.2.2 MIH_NET_SAP 58
7.2.3 Media independent SAPs 59
7.2.3.1 MIH_SAP 59
7.3 MIH_LINK_SAP primitives 60
7.3.1 Link_Detected.indication 60
7.3.1.1 Function 60
7.3.1.2 Semantics of service primitive 60
7.3.1.3 When generated 61
7.3.1.4 Effect on receipt 61
7.3.2 Link_Up.indication 61
7.3.2.1 Function 61
7.3.2.2 Semantics of service primitive 61
7.3.2.3 When generated 62
7.3.2.4 Effect on receipt 62
7.3.3 Link_Down.indication 62
7.3.3.1 Function 62
7.3.3.2 Semantics of service primitive 62
7.3.3.3 When generated 62
7.3.3.4 Effect on receipt 63
7.3.4 Link_Going_Down.indication 63
7.3.4.1 Function 63
7.3.4.2 Semantics of service primitive 63
7.3.4.3 When generated 63
7.3.4.4 Effect on receipt 64
7.3.5 Link_Event_Rollback.indication 64
7.3.5.1 Function 64
7.3.5.2 Semantics of service primitive 64
7.3.5.3 When generated 64
7.3.5.4 Effect on receipt 64
7.3.6 Link_Parameters_Report.indication 65
7.3.6.1 Function 65
7.3.6.2 Semantics of service primitive 65
7.3.6.3 When generated 65
7.3.6.4 Effect on receipt 65
7.3.7 Link_PDU_Transmit_Status.indication 65
7.3.7.1 Function 65
7.3.7.2 Semantics of service primitive 66
7.3.7.3 When generated 66
7.3.7.4 Effect on receipt 66

7.3.8 Link_Handover_Imminent.indication ... 67
7.3.8.1 Function ... 67
7.3.8.2 Semantics of service primitive ... 67
7.3.8.3 When generated ... 67
7.3.8.4 Effect on receipt ... 67
7.3.9 Link_Handover_Complete.indication ... 67
7.3.9.1 Function ... 67
7.3.9.2 Semantics of service primitive ... 68
7.3.9.3 When generated ... 68
7.3.9.4 Effect on receipt ... 68
7.3.10 Link_Event_Subscribe ... 68
7.3.10.1 Link_Event_Subscribe.request ... 68
7.3.10.2 Link_Event_Subscribe.confirm ... 69
7.3.11 Link_Event_Unsubscribe ... 69
7.3.11.1 Link_Event_Unsubscribe.request ... 69
7.3.11.2 Link_Event_Unsubscribe.confirm ... 70
7.3.12 Link_Configure_Thresholds ... 71
7.3.12.1 Link_Configure_Thresholds.request ... 71
7.3.12.2 Link_Configure_Thresholds.confirm ... 71
7.3.13 Link_Capability_Discover ... 72
7.3.13.1 Link_Capability_Discover.request ... 72
7.3.13.2 Link_Capability_Discover.confirm ... 72
7.3.14 Link_Get_Parameters ... 73
7.3.14.1 Link_Get_Parameters.request ... 73
7.3.14.2 Link_Get_Parameters.confirm ... 74
7.3.15 Link_Action ... 75
7.3.15.1 Link_Action.request ... 75
7.3.15.2 Link_Action.confirm ... 76
7.4 MIH_SAP primitives ... 76
7.4.1 MIH_Capability_Discover ... 77
7.4.1.1 MIH_Capability_Discover.request ... 77
7.4.1.2 MIH_Capability_Discover.indication ... 78
7.4.1.3 MIH_Capability_Discover.response ... 79
7.4.1.4 MIH_Capability_Discover.confirm ... 80
7.4.2 MIH_Register ... 81
7.4.2.1 MIH_Register.request ... 81
7.4.2.2 MIH_Register.indication ... 81
7.4.2.3 MIH_Register.response ... 82
7.4.2.4 MIH_Register.confirm ... 83
7.4.3 MIH_DeRegister ... 83
7.4.3.1 MIH_DeRegister.request ... 83
7.4.3.2 MIH_DeRegister.indication ... 84
7.4.3.3 MIH_DeRegister.response ... 85
7.4.3.4 MIH_DeRegister.confirm ... 85
7.4.4 MIH_Event_Subscribe ... 86
7.4.4.1 MIH_Event_Subscribe.request ... 86
7.4.4.2 MIH_Event_Subscribe.confirm ... 87
7.4.5 MIH_Event_Unsubscribe ... 88
7.4.5.1 MIH_Event_Unsubscribe.request ... 88
7.4.5.2 MIH_Event_Unsubscribe.confirm ... 88
7.4.6 MIH_Link_Up ... 89
7.4.6.1 MIH_Link_Up.indication ... 89
7.4.7 MIH_Link_Down ... 90
7.4.7.1 MIH_Link_Down.indication ... 90

7.4.8 MIH_Link_Going_Down 91
7.4.8.1 MIH_Link_Going_Down.indication 91
7.4.9 MIH_Link_Detected 92
7.4.9.1 MIH_Link_Detected.indication 92
7.4.10 MIH_Link_Parameters_Report 93
7.4.10.1 MIH_Link_Parameters_Report.indication 93
7.4.11 MIH_Link_Event_Rollback 94
7.4.11.1 MIH_Link_Event_Rollback.indication 94
7.4.12 MIH_Link_PDU_Transmit_Status 94
7.4.12.1 MIH_Link_PDU_Transmit_Status.indication 94
7.4.13 MIH_Link_Handover_Imminent 95
7.4.13.1 MIH_Link_Handover_Imminent.indication 95
7.4.14 MIH_Link_Handover_Complete 96
7.4.14.1 MIH_Link_Handover_Complete.indication 96
7.4.15 MIH_Get_Link_Parameters 97
7.4.15.1 MIH_Get_Link_Parameters.request 97
7.4.15.2 MIH_Get_Link_Parameters.confirm 98
7.4.16 MIH_Link_Configure_Thresholds 99
7.4.16.1 MIH_Link_Configure_Thresholds.request 99
7.4.16.2 MIH_Link_Configure_Thresholds.confirm 100
7.4.17 MIH_Link_Actions 101
7.4.17.1 MIH_Link_Actions.request 101
7.4.17.2 MIH_Link_Actions.confirm 101
7.4.18 MIH_Net_HO_Candidate_Query 102
7.4.18.1 General 102
7.4.18.2 MIH_Net_HO_Candidate_Query.request 102
7.4.18.3 MIH_Net_HO_Candidate_Query.indication 103
7.4.18.4 MIH_Net_HO_Candidate_Query.response 104
7.4.18.5 MIH_Net_HO_Candidate_Query.confirm 105
7.4.19 MIH_MN_HO_Candidate_Query 107
7.4.19.1 MIH_MN_HO_Candidate_Query.request 107
7.4.19.2 MIH_MN_HO_Candidate_Query.indication 108
7.4.19.3 MIH_MN_HO_Candidate_Query.response 109
7.4.19.4 MIH_MN_HO_Candidate_Query.confirm 111
7.4.20 MIH_N2N_HO_Query_Resources 112
7.4.20.1 MIH_N2N_HO_Query_Resources.request 112
7.4.20.2 MIH_N2N_HO_Query_Resources.indication 113
7.4.20.3 MIH_N2N_HO_Query_Resources.response 114
7.4.20.4 MIH_N2N_HO_Query_Resources.confirm 116
7.4.21 MIH_Net_HO_Commit 117
7.4.21.1 MIH_Net_HO_Commit.request 117
7.4.21.2 MIH_Net_HO_Commit.indication 118
7.4.21.3 MIH_Net_HO_Commit.response 119
7.4.21.4 MIH_Net_HO_Commit.confirm 120
7.4.22 MIH_MN_HO_Commit 121
7.4.22.1 MIH_MN_HO_Commit.request 121
7.4.22.2 MIH_MN_HO_Commit.confirm 122
7.4.23 MIH_N2N_HO_Commit 122
7.4.23.1 MIH_N2N_HO_Commit.request 122
7.4.23.2 MIH_N2N_HO_Commit.indication 123
7.4.23.3 MIH_N2N_HO_Commit.response 124
7.4.23.4 MIH_N2N_HO_Commit.confirm 125
7.4.24 MIH_MN_HO_Complete 126
7.4.24.1 MIH_MN_HO_Complete.request 126

7.4.24.2 MIH_MN_HO_Complete.indication ... 127
7.4.24.3 MIH_MN_HO_Complete.response ... 128
7.4.24.4 MIH_MN_HO_Complete.confirm ... 128
7.4.25 MIH_N2N_HO_Complete ... 129
7.4.25.1 MIH_N2N_HO_Complete.request ... 129
7.4.25.2 MIH_N2N_HO_Complete.indication ... 130
7.4.25.3 MIH_N2N_HO_Complete.response ... 130
7.4.25.4 MIH_N2N_HO_Complete.confirm ... 131
7.4.26 MIH_Get_Information ... 132
7.4.26.1 MIH_Get_Information.request ... 132
7.4.26.2 MIH_Get_Information.indication ... 133
7.4.26.3 MIH_Get_Information.response ... 135
7.4.26.4 MIH_Get_Information.confirm ... 136
7.5 MIH_NET_SAP primitives ... 137
7.5.1 MIH_TP_Data ... 137
7.5.1.1 MIH_TP_Data.request ... 137
7.5.1.2 MIH_TP_Data.indication ... 138
7.5.1.3 MIH_TP_Data.confirm ... 138

8. Media Independent Handover Protocol ... 141

8.1 Introduction ... 141
8.2 MIH protocol description ... 141
8.2.1 MIH protocol acknowledgement operation ... 141
8.2.2 MIH protocol transaction state diagram ... 142
8.2.2.1 State machines ... 143
8.2.2.2 Notational conventions used in state diagrams ... 143
8.2.2.3 Global variables ... 145
8.2.2.4 Global procedures ... 146
8.2.2.5 Global constants ... 146
8.2.2.6 Timers ... 146
8.2.2.7 Transaction source and destination state machines ... 147
8.2.2.8 ACK related state machines ... 149
8.2.3 Other considerations ... 151
8.2.3.1 Congestion control and load management ... 151
8.2.3.2 Reliability ... 152
8.2.3.3 Fragmentation ... 152
8.2.3.4 MIHF discovery ... 152
8.3 MIH protocol identifiers ... 154
8.3.1 MIHF ID ... 154
8.3.2 Transaction ID ... 154
8.4 MIH protocol frame format ... 154
8.4.1 General frame format ... 154
8.4.1.1 MIH protocol header fields ... 155
8.5 Message parameter TLV encoding ... 156
8.6 MIH protocol messages ... 156
8.6.1 MIH messages for service management ... 157
8.6.1.1 MIH_Capability_Discover request ... 157
8.6.1.2 MIH_Capability_Discover response ... 157
8.6.1.3 MIH_Register request ... 158
8.6.1.4 MIH_Register response ... 158
8.6.1.5 MIH_DeRegister request ... 159
8.6.1.6 MIH_DeRegister response ... 159
8.6.1.7 MIH_Event_Subscribe request ... 159

8.6.1.8 MIH_Event_Subscribe response 160
8.6.1.9 MIH_Event_Unsubscribe request 160
8.6.1.10 MIH_Event_Unsubscribe response 161
8.6.2 MIH messages for event service 161
8.6.2.1 MIH_Link_Up indication 161
8.6.2.2 MIH_Link_Down indication 161
8.6.2.3 MIH_Link_Going_Down indication 162
8.6.2.4 MIH_Link_Detected indication 162
8.6.2.5 MIH_Link_Parameters_Report indication 163
8.6.2.6 MIH_Link_Event_Rollback indication 163
8.6.2.7 MIH_Link_Handover_Imminent indication 163
8.6.2.8 MIH_Link_Handover_Complete indication 164
8.6.3 MIH messages for command service 164
8.6.3.1 MIH_Get_Link_Parameters request 164
8.6.3.2 MIH_Get_Link_Parameters response 165
8.6.3.3 MIH_Link_Configure_Thresholds request 165
8.6.3.4 MIH_Link_Configure_Thresholds response 165
8.6.3.5 MIH_Link_Actions request 166
8.6.3.6 MIH_Link_Actions response 166
8.6.3.7 MIH_Net_HO_Candidate_Query request 166
8.6.3.8 MIH_Net_HO_Candidate_Query response 167
8.6.3.9 MIH_MN_HO_Candidate_Query request 168
8.6.3.10 MIH_MN_HO_Candidate_Query response 168
8.6.3.11 MIH_N2N_HO_Query_Resources request 169
8.6.3.12 MIH_N2N_HO_Query_Resources response 169
8.6.3.13 MIH_Net_HO_Commit request 170
8.6.3.14 MIH_Net_HO_Commit response 171
8.6.3.15 MIH_N2N_HO_Commit request 171
8.6.3.16 MIH_N2N_HO_Commit response 172
8.6.3.17 MIH_MN_HO_Complete request 172
8.6.3.18 MIH_MN_HO_Complete response 172
8.6.3.19 MIH_N2N_HO_Complete request 173
8.6.3.20 MIH_N2N_HO_Complete response 173
8.6.4 MIH messages for information service 174
8.6.4.1 MIH_Get_Information request 174
8.6.4.2 MIH_Get_Information response 174

Annexes

Annex A (normative) Type identifiers for information elements 177

Annex B (normative) Data type definition 178

B.1 General 178
B.2 Basic data types 178
B.2.1 Encoding rule for data type LIST(DATATYPE) 179
B.3 Derived data types 180
B.3.1 General 180
B.3.2 General data types 180
B.3.3 Data types for addresses 181
B.3.4 Data types for link identification and manipulation 182
B.3.5 Data types for QoS 191
B.3.6 Data types for location 193
B.3.7 Data types for IP configuration 196
B.3.8 Data types for information elements 197

B.3.9 Data types for information service query 202
B.3.10 Data types for information service response 206
B.3.11 Data type for MIHF identification 206
B.3.12 Data type for MIH capabilities 207
B.3.13 Data type for MIH registration 209
B.3.14 Data types for handover operation 209
B.3.15 Data types for MIH_NET_SAP primitives 210

Annex C (normative) MIH protocol message code assignments 211

Annex D (normative) MIIS basic schema 215

Annex E (informative) Bibliography 231

Annex F (informative) Example UML representation of basic schema 233

Annex G (informative) Making User Extensions to MIIS schema 235

Annex H (informative) Handover procedures 237

H.1 Mobile-initiated handover procedure 237
H.2 Network-initiated handover procedure 243
H.3 Network-initiated handover from 802.16 to 3G 248
H.3.1 Network-initiated handover from 802.16 to 3G (Single PoS) 248
H.3.2 Network-initiated handover from 802.16 to 3G (Multiple PoS) 251
H.4 Handover from 3G to WLAN 256
H.4.1 Network-initiated handover from 3G to WLAN 256
H.4.2 Mobile-initiated handover from 3G to WLAN 260
H.5 Example handover flow chart between 802.11 and 802.16 264
H.6 Example handover flow charts for Mobile IP 268
H.6.1 Mobile-initiated handover procedure 268
H.6.2 Network-initiated handover procedure 273
H.7 Example handover flow chart for Proxy Mobile IPv6 278
H.7.1 Network-initiated handover procedures 278
H.7.2 Mobile-initiated handover procedures 281
H.8 Network Selection in 802.11 (WLAN) using 802.21 284
H.9 Handover flow diagram for abnormal cases 286
H.9.1 Abnormal case at the stage of resource availability check 286
H.9.2 Abnormal case at the stage of handover resource reservation 287

Annex I (informative) Mapping MIH messages to reference points 290

Annex J (informative) Quality of service examples 292

J.1 Generic IEEE 802.21 QoS flow diagram 293
J.2 Generic IEEE 802.21 QoS parameter mappings 294
J.3 Example of deriving generic IEEE 802.21 QoS parameters 297
J.3.1 Packet loss rate 297
J.3.2 Packet error rate 298
J.3.3 Average transfer delay 298
J.3.4 Packet transfer delay jitter 299

Annex K (informative) Media specific mapping for SAPs 301

K.1 MIH_LINK_SAP mapping to specific technologies 301
K.2 C_SAP primitives referenced by IEEE 802.21 304

Annex L (informative) Information service examples........ 305

L.1 Example query for RDF_DATA........ 305
L.2 Example query for RDF_SCHEMA_URL 305
L.3 Example query for RDF_SCHEMA 305
L.4 SPARQL query example and usage of RDF schema 306
L.4.1 RDF example........ 306
L.4.2 SPARQL query example 310
L.4.3 SPARQL query 320

Annex M (normative) Mappings from MIH_LINK_SAP to media-specific SAPs 322

M.1 802.3 322
M.2 802.11 322
M.3 802.16 322
M.4 3GPP and 3GPP2 323

List of Figures

Figure 1 —MIH services and their initiation ... 3
Figure 2 —MIHF communication model ... 18
Figure 3 —Example of network model with MIH services ... 20
Figure 4 —General MIHF reference model and SAPs ... 22
Figure 5 —Types of MIHF relationship ... 23
Figure 6 —MIH reference model for IEEE 802.3 ... 24
Figure 7 —MIH reference model for IEEE 802.11 ... 25
Figure 8 —MIH reference model for IEEE 802.16 ... 26
Figure 9 —MIH reference model for 3GPP systems ... 27
Figure 10 —MIH reference model for 3GPP2 systems ... 27
Figure 11—Relationship between different MIHF SAPs ... 29
Figure 12—MIH registration flow ... 33
Figure 13 —Link events and MIH events ... 35
Figure 14 —Remote MIH events ... 35
Figure 15 —MIH events subscription and flow ... 37
Figure 16 —Link commands and MIH commands ... 40
Figure 17 —Remote MIH command ... 41
Figure 18 —Command service flow ... 42
Figure 19—Depicting a list of neighboring networks with information elements. ... 49
Figure 20 —TLV representation of information elements ... 51
Figure 21 —MIIS information flow ... 55
Figure 22—State machines interactions ... 143
Figure 23—Transaction Timers state machine ... 147
Figure 24—Transaction source state machine ... 148
Figure 25—Transaction destination state machine ... 149
Figure 26—ACK requestor state machine ... 150
Figure 27—ACK responder state machine ... 151
Figure 28 —MIH protocol general frame format ... 154
Figure 29 —MIH protocol header format ... 155
Figure F-1—UML representation of basic schema ... 233
Figure F-2—UML representation of basic schema (cont.) ... 234
Figure G-1—Extended schema ... 235
Figure H-1—Mobile-initiated handover procedure ... 238
Figure H-1a—Mobile-initiated handover procedure (cont.) ... 239
Figure H-1b—Mobile-initiated handover procedure (cont.) ... 240
Figure H-1c—Mobile-initiated handover procedure (cont.) ... 242
Figure H-2—Network-initiated handover procedure ... 244
Figure H-2a—Network-initiated handover procedure (cont.) ... 245
Figure H-2b—Network-initiated handover procedure (cont.) ... 246
Figure H-2c—Network-initiated handover procedure (cont.) ... 247
Figure H-3—Network-initiated handover from 802.16 to 3G (Single PoS) ... 248
Figure H-3a—Network-initiated handover from 802.16 to 3G (Single PoS) (cont.) ... 249
Figure H-3b—Network-initiated handover from 802.16 to 3G (Single PoS) (cont.) ... 250
Figure H-4—Network-initiated handover from 802.16 to 3G (Multiple PoS) ... 251
Figure H-4a—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.) ... 252
Figure H-4b—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.) ... 253
Figure H-4c—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.) ... 254
Figure H-4d—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.) ... 255
Figure H-5—Network-initiated handover from 3G to WLAN ... 256
Figure H-5a—Network-initiated handover from 3G to WLAN (cont.) ... 257

Figure H-5b—Network-initiated handover from 3G to WLAN (cont.) 258
Figure H-5c—Network-initiated handover from 3G to WLAN (cont.) 259
Figure H-6—Mobile-initiated handover from 3G to WLAN 260
Figure H-6a—Mobile-initiated handover from 3G to WLAN (cont.) 261
Figure H-6b—Mobile-initiated handover from 3G to WLAN (cont.) 262
Figure H-6c—Mobile-initiated handover from 3G to WLAN (cont.) 263
Figure H-7—Example handover flow chart between 802.11 and 802.16 264
Figure H-7a—Example handover flow chart between 802.11 and 802.16 (cont.) 265
Figure H-7b—Example handover flow chart between 802.11 and 802.16 (cont.) 266
Figure H-7c—Example handover flow chart between 802.11 and 802.16 (cont.) 267
Figure H-8—Mobile-initiated handover procedure 268
Figure H-8a—Mobile-initiated handover procedure (cont.) 269
Figure H-8b—Mobile-initiated handover procedure (cont.) 270
Figure H-8c—Mobile-initiated handover procedure (cont.) 271
Figure H-8d—Mobile-initiated handover procedure (cont.) 272
Figure H-9—Network-initiated handover procedure 273
Figure H-9a—Network-initiated handover procedure (cont.) 274
Figure H-9c—Network-initiated handover procedure (cont.) 276
Figure H-9d—Network-initiated handover procedure (cont.) 277
Figure H-10—Network-initiated handover procedures 279
Figure H-10a—Network-initiated handover procedures (cont.) 280
Figure H-11—Mobile-initiated handover procedures 282
Figure H-11a—Mobile-initiated handover procedures (cont.) 283
Figure H-12—Network selection in WLAN with 802.11 and 802.21 284
Figure H-13—Use case: query SSPN list 285
Figure H-14—Abnormal case at the stage of resource availability check 286
Figure H-15—Abnormal case at the stage of handover resource reservation 287
Figure J-1—An example flow for setting application QoS requirements 294
Figure L-1—-Graphic representation of example network (network 1) 309
Figure L-2—Graphic representation of example network (network 2 and 3) 310
Figure L-3—The serching area of the location based query 321

List of Tables

Table 1—Summary of reference points 19
Table 2—Service management primitives 32
Table 3—Link events 38
Table 4—MIH events 39
Table 5—MIH commands 43
Table 6—Naming convention for MIH handover command primitives 44
Table 7—Link commands 44
Table 8—Information elements 47
Table 9—Information elements name space 50
Table 10—MIH_LINK_SAP primitives 58
Table 11—MIH_NET_SAP primitive 58
Table 12—MIH_SAP primitives 59
Table 13—State machine symbols 144
Table 14—Global state machine variables 145
Table 15—Description of MIH protocol header fields 155
Table A-1—Type identifiers for information elements 177
Table B-1—Basic data types 178
Table B-2—General data types 180
Table B-3—Data types for address 181
Table B-4—Data types for links 182
Table B-5—Link actions 189
Table B-6—Link action attributes 190
Table B-7—Link Down reason code 190
Table B-8—Link Going Down reason code 191
Table B-9—Data types for QoS 192
Table B-10—Data types for location 193
Table B-11—Value field format of PoA location information (geospatial location) 194
Table B-12—Data types for IP configuration 196
Table B-13—Data types for information elements 197
Table B-14—Network type and revision representation 201
Table B-15—Data types for binary query 203
Table B-16—Data type for RDF query 205
Table B-17—Data type for binary information query response 206
Table B-18—Data type for RDF information query response 206
Table B-19—Data type for MIHF identification 207
Table B-20—Data type for MIH capabilities 207
Table B-21—Data type for MIH registration 209
Table B-22—Data type for handover operation 210
Table B-23—Data type for MIH_NET_SAP primitives 210
Table C-1—AID assignment 211
Table C-2—Type values for TLV encoding 212
Table I-1—Mapping MIH messages to reference points 290
Table J-1—Example QoS parameter mapping table for IEEE 802.11K 295
Table J-2—An example of a QoS parameter mapping table 296
Table J-3—Example QoS parameter mapping table for 3GPP 297
Table K-1—MIH_Link_SAP/802.16 primitives mapping 301
Table K-2—MIH_LINK_SAP/802.11/802.3ah/802.1ag primitives mapping 302
Table K-3—MIH_LINK_SAP/3GPP/3GPP2 primitives mapping 303
Table K-4—C_SAP primitives referenced by 802.21 304

IEEE P802.21/D8.0

Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services

1. Introduction

1.1 Scope

This standard defines extensible 802 media access independent mechanisms that enable the optimization of handover between heterogeneous 802 systems and may facilitate handover between 802 systems and cellular systems.

1.2 Purpose

The purpose is to improve the user experience of mobile devices by facilitating handover between 802 networks whether or not they are of different media types, including both wired and wireless, where handover is not otherwise defined and to make it possible for mobile devices to perform seamless handover where the network environment supports it. These mechanisms may also be usable for handovers between 802 networks and non 802 networks.

1.3 Overview

This standard provides link layer intelligence and other related network information to upper layers to optimize handovers between heterogeneous networks. This includes media types specified by Third Generation (3G) Partnership Project (3GPP), 3G Partnership Project 2 (3GPP2), and both wired and wireless media in the IEEE 802 family of standards. In this standard, unless otherwise noted, *media* refers to method/mode of accessing a telecommunication system (e.g., cable, radio, satellite), as opposed to sensory aspects of communication (e.g., audio, video).

The following items are not within the scope of this standard:

1) Intra-technology handover (except for handovers across extended service sets (ESSs) in case of IEEE 802.11);

2) Handover policy;

3) Security mechanisms;

4) Enhancements that are specific to particular link layer technologies and required to support this standard will be carried out by those respective link-layer technology standards;

5) Higher layer (layer 3 and above) enhancements that are required to support this standard.

The purpose of this standard is to enhance the experience of mobile users by facilitating handovers between heterogeneous networks. The standard addresses the support of handovers for both mobile and stationary users. For mobile users, handovers may occur due to changes in wireless link conditions due to, for example, the user's movement. For the stationary user, handovers may become imminent when the surrounding network environment changes, making one network more attractive than another.

This standard supports another important aspect of optimized handover - link adaptation. A user may choose an application which requires a higher data rate than available on the current link, necessitating a link adaptation to provide the higher rate, or necessitating a handover if the higher rate is unavailable on the current link.

In all such cases service continuity should be maintained to the extent possible during handover. As an example, when making a network transition during a phone call the handover procedures should be executed in such a way that any perceptible interruption to the conversation will be minimized.

This standard supports cooperative use of information available at the mobile node and within the network infrastructure. The mobile node is well-placed to detect available networks. The network infrastructure is well-suited to store overall network information, such as neighborhood cell lists, location of mobile nodes, and higher layer service availability. Both the mobile node and the network may make decisions about connectivity. In general, both the mobile node and the network points of attachment such as base stations and access points may be multi-modal (i.e., capable of supporting multiple radio standards and simultaneously supporting connections on more than one radio interface).

The overall network may include pico cells such as IEEE 802.15, micro cells such as IEEE 802.11, and macro cells (such as 3GPP, 3GPP2, or IEEE 802.16) with overlapping coverage. The handover process may be initiated by measurement reports and triggers supplied by the link layers on the mobile node. The measurement reports may include metrics such as signal quality, synchronization time differences, and transmission error rates. Specifically the standard consists of the following elements:

a) A framework that enables service continuity while a mobile node (MN) transitions between heterogeneous link-layer technologies. The framework relies on the presence of a mobility management protocol stack within the network elements that support the handover. The framework presents media independent handover (MIH) reference models for different link layer technologies.

b) A set of handover-enabling functions within the protocol stacks of the network elements and a new entity created therein called the MIH Function (MIHF).

c) A media independent handover Service Access Point (called the MIH_SAP) and associated primitives are defined to provide MIH Users with access to the services of the MIHF. The MIHF provides the following services:

   1) The Media Independent Event service which detects changes in link layer properties and initiates appropriate events (triggers) from both local and remote interfaces.

   2) The Media Independent Command service provides a set of commands for the MIH Users to control link properties that are relevant to handover and switch between links if required.

3) The Media Independent Information service provides the information about different networks and their services thus enabling more effective handover decision making across heterogeneous networks.

d) The definition of new link layer SAPs and associated primitives for each link-layer technology. The new primitives help the MIHF collect link information and control link behavior during handovers. If applicable, the new SAPs may be recommended as amendments to the standards for the respective link-layer technology.

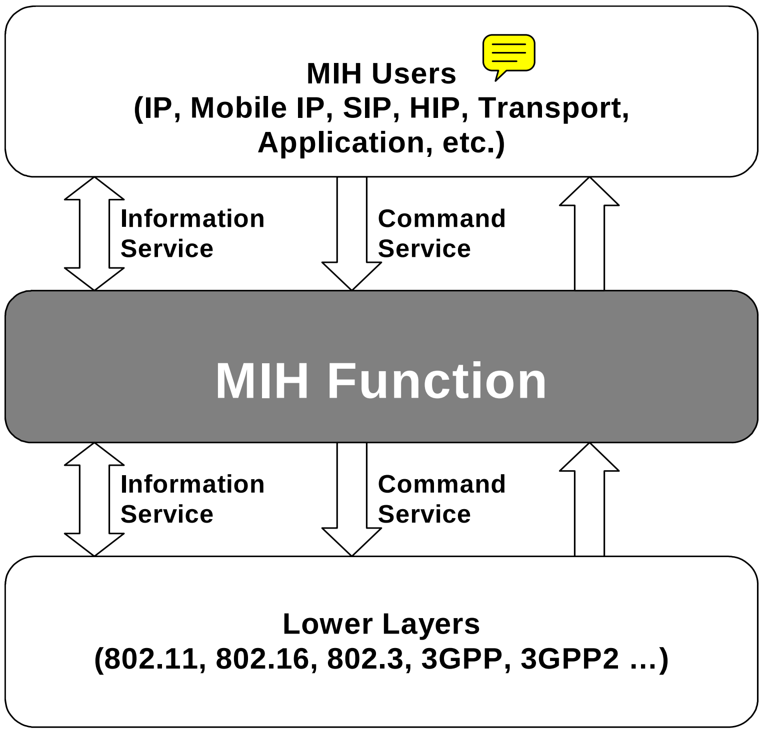


Figure 1 — MIH services and their initiation

Figure 1 shows the placement of the MIHF within the protocol stack of an MN or network entity. The MIHF provides services to the MIH Users through a single media independent interface (the MIH service access point) and obtains services from the lower layers through a variety of media dependent interfaces (media-specific SAPs).

1.4 Assumptions

The following assumptions have been made in the development of this standard:

1) The mobile node is capable of supporting multiple link-layer technologies, which may be wireless or wired;

2) The MIHF is a logical entity, whose definition is independent of its deployment location which may be on the mobile node or in the network;

3) The MIHF, regardless whether it is located on the mobile node or in the network, may receive and transmit information about the configuration and condition of access networks around the MN. This information may originate at different layers of the protocol stack within the MN or at various network elements.

i) When the information originates at a remote network element, the MIHF on the local network element obtains it through MIH message exchanges with a peer MIHF instance that resides in the remote network element.

ii) When the information originates at lower layers of the protocol stack within an MN or network entity, the MIHF on that entity obtains it locally through the service primitives of the SAPs that define the interface of the MIHF with the lower layers.

1.5 Media independence

The intent of this standard is to provide generic link layer intelligence independent of the specifics of mobile nodes or radio networks. As such this standard is intended to provide a generic interface between the link layer users in the mobility-management protocol stack and existing media-specific link layers, such as those specified by 3GPP, 3GPP2 and the IEEE 802 family of standards.

This standard defines SAPs and primitives that provide generic link layer intelligence. Individual media-specific technologies thereafter need to enhance their media-specific SAPs and primitives to satisfy the generic abstractions of this standard. Suitable amendments are required to existing link layer (MAC/PHY) standards of different media-specific technologies such as IEEE 802.3, IEEE 802.11, IEEE 802.16, 3GPP, and 3GPP2 to satisfy the requirements of generic link layer intelligence identified by this standard.

2. Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

3GPP TS 23.003 (2007-09), Numbering, addressing and identification (Release 7).

3GPP TS 25.008, Digital cellular telecommunication system (Phase 2+); Radio subsystem link control.

3GPP TS 25.215 (2007-11), Physical layer - Measurements (FDD) (Release 7).

3GPP TS 25.401 (2007-09), UTRAN overall description (Release 7).

3GPP TS 25.413 (2007-09), UTRAN Iu interface RANAP signalling (Release 7).

3GPP2 C.S0004-D (2004-02), Signaling Link Access Control (LAC) Standard for cdma2000 Spread Spectrum Systems.

IEEE Draft Std 802.16Rev2/D0d (2007-08), Information Technology- Telecommunications and information exchange between system-Local and metropolitan area networks-Specific Requirements-Part 16: Air Interface for Fixed Broadband Wireless Access Systems.

IEEE Std 802.16™-2004 [ISO/IEC 8802-16: 2004], Information Technology- Telecommunications and information exchange between system-Local and metropolitan area networks-Specific Requirements-Part 16: Air Interface for Fixed Broadband Wireless Access Systems.

IEEE Std 802.16E™-2005 [ISO/IEC 8802-16: 2005], Information Technology- Telecommunications and information exchange between system-Local and metropolitan area networks-Specific Requirements-Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems (Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands).

IETF RFC 1661 (1994-07), The Point-to-Point Protocol (PPP).

IETF RFC 2865 (2000-06), Remote Authentication Dial In User Service (RADIUS).

IETF RFC 3344 (2002-08), IP Mobility Support for IPv4.

IETF RFC 3588 (2003-09), Diameter Base Protocol.

IETF RFC 3775 (2004-06), Mobility Support in IPv6.

IETF RFC 3825 (2004-07), Dynamic Host Configuration Protocol Option for Coordinate-based Location Configuration Information.

IETF RFC 4068 (2005-07), Fast Handovers for Mobile IPv6.

IETF RFC 4119 (2005-12), A Presence-based GEOPRIV Location Object Format.

IETF RFC 4140 (2005-08), Hierarchical Mobile IPv6 Mobility Management (HMIPv6).

IETF RFC 4555 (2006-06), IKEv2 Mobility and Multihoming Protocol (MOBIKE).

IETF RFC 4776 (2006-11), Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information.

IETF RFC 4857 (2007-06), Mobile IPv4 Regional Registration.

IETF RFC 4881 (2007-06), Low-Latency Handoffs in Mobile IPv4.

ISO 3166-1 (1997), Codes for the representation of names of countries and their subdivisions – Part 1: Country codes.

W3C Recommendation, RDF/XML Syntax Specification.

W3C Recommendation, Resource Description Framework (RDF) – Concepts and Abstract Syntax.

W3C Recommendation, SPARQL Query Language for RDF.

3. Definitions

For the purpose of this standard, the following terms and definitions apply. The Authoritative Dictionary of IEEE Standards Terms should be referenced for terms not defined in this clause.

3.1 access router: An IP router residing in an access network and connected to one or more access points. An access router offers IP connectivity to mobile nodes.

3.2 candidate point of attachment (candidate PoA): A Point of Attachment (PoA) under evaluation to which the link may be switched.

3.3 handoff: *See:* **handover.**

3.4 handover: The process by which a mobile node obtains facilities and preserves traffic flows upon occurrence of a link switch event. The mechanisms and protocol layers involved in the handover may vary with the type of the link switch event (i.e., with the type of the serving and target point of attachment and the respective subnet associations). Different types of handover are defined based on the way facilities for supporting traffic flows are preserved. *Syn:* **handoff.** *See also* **hard handover; soft handover; seamless handover**.

3.5 hard handover: Handover where facilities for supporting traffic flows are subject to complete unavailability between their disruption on the serving link and their restoration on the target link (break-before-make).

3.6 handover policies: A set of rules that contribute to shaping the handover decision for a mobile node.

3.7 home subscriber network: Network managed by operator with whom the subscriber has a business relationship (subscription). *See also* **visited network; serving network.**

3.8 horizontal handovers: This involves mobile nodes moving between Point of Attachments of the same link type (in terms of coverage, data rate and mobility), such as UMTS to UMTS or WLAN to WLAN. *Syn:* **intra-technology handovers.**

3.9 inter-technology handovers: *See*: **vertical handovers.**

3.10 intra-technology handovers: *See*: **horizontal handovers.**

3.11 link: A communication channel through which nodes may communicate for the exchange of L2 protocol data units. Each link is associated with two endpoints and has a unique identifier.

3.12 link layer: Conceptual layer of control or processing logic that is responsible for maintaining control of the data link. The data link layer functions provide an interface between the higher-layer logic and the data link.

3.13 link indication: Link state information provided by the link layer to higher layers.

3.14 link switch: The process by which a mobile node changes the link that connects it to the network. Changing a link implies changing the remote link endpoint and therefore the point of attachment of the mobile node.

3.15 lower layers: The layers located at OSI Level 2 and below across different link-layer technology standards supported by this standard. For example, the IEEE 802.11 Lower Layers are the MAC Sublayer and the PHY, while the 3GPP Lower Layers are L1/MAC/RLC/PDCP in the case of W-CDMA FDD/TDD, L1/LAPDm in the case of GSM CS, and L1/MAC/RLC in the case of GPRS/EGPRS, respectively. The term

“Lower Layers” also includes Logical Link Control Layers such as IEEE 802.2 Logical Link Control (LLC) or 3GPP Radio Link Control (RLC). The MIHF uses the services provided by these layers.

3.16 MIH discovery protocol – A protocol for discovering MIH entities.

3.17 MIH network entity: Network Entity with MIHF capability.

3.18 MIH point of service (MIH PoS): Network-side MIHF instance that exchanges MIH messages with an MN-based MIHF. The same MIH Network Entity includes an MIH PoS for each MIH-enabled MN with which it exchanges MIH messages. A single MIH PoS may host more than one MIH service. The same MIH Network Entity may include multiple MIH Points of Service that may provide different combinations of MIH services to the respective MNs based on subscription or roaming conditions. Note that for a network entity comprising of multiple interfaces, the notion of MIH PoS is associated with the network entity itself and not with just one of its interfaces.

3.19 MIH node: An MIHF capable entity (MN or Network).

3.20 MIH non-PoS: An MIH network entity that can directly exchange MIH messages with other MIH network entities but cannot *directly* exchange MIH messages with any MIH enabled MN.

3.21 MIH transport protocol: A protocol for transporting MIH protocol messages between a pair of MIH entities.

3.22 MIH Users: Entities that use the services provided by the MIHF. MIH Users use the MIH_SAP to interact with the MIHF.

3.23 media independent handover function (MIHF): A functional implementation of MIH services.

3.24 MIHF pairing: The communication relationship that exists between distinct MIHF instances when they exchange MIH messages or MIH information.

3.25 MIHF transaction: A combination of an MIH Request or Indication message and the corresponding MIH Response message (if applicable) that are exchanged between two MIH peers. It is required to match each MIH request message that is sent by the initiator with the corresponding MIH response message. Acknowledgement messages associated with MIH message exchange are also part of the transaction.

3.26 mobile-controlled handover: The mobile node has the primary control over the handover process.

3.27 mobile-initiated handover: The mobile node initiates the handover process by indicating to the network that the handover is necessary or desired.

3.28 mobile node (MN): Communication node that may change its Point of Attachment from one link to another.

3.29 mobile node association: The connectivity state where the mobile node is ready to exchange user data (like TCP/UDP packets) with the network point of attachment.

3.30 network detection: The process by which a mobile node collects information on networks in its locality, identifies the different points of attachment, and ascertains the validity of link-layer configuration.

3.31 network entity: Communication node inside the network.

3.32 network-controlled handover: The network has the primary control over the handover process.

3.33 network-initiated handover: The network initiates the handover process by indicating to the mobile node that the handover is necessary or desired.

3.34 network neighborhood: The area of interest in which the network discovery and selection entity seeks to determine the available coverage of a wired/wireless network with identical or different link-layer technologies.

3.35 network point of attachment (network PoA, or PoA): Network side endpoint of a layer 2 link that includes an MN as the other endpoint. *See also* **candidate PoA; serving PoA; target PoA**.

3.36 network selection: The process by which a mobile node or a network entity makes a decision to connect to a specific network (possibly out of many available) based on a policy configured in the mobile node and/or obtained from the network.

3.37 network selector: The entity that undertakes the network selection decision that may lead to a handover.

3.38 operator identifier (operator ID): An identifier of the access or core network provider.

3.39 protocol data unit: Packet produced by a protocol for exchange between the protocol's peer entities.

3.40 seamless handover: Handover associated with a link switch between points of attachment, where the mobile node either experiences no degradation in service quality, security, and capabilities, or experiences some degradation in service parameters that is mutually acceptable to the mobile subscriber and to the network that serves the newly connected interface.

3.41 s**ervice primitive**: Conceptual abstraction describing the interaction between an upper layer user and the present layer in the provisioning of a service. The abstraction resides in the exclusive specification of the service provided and not in the means by which the service is provided.

3.42 serving network: Network that provides services to the user. The serving network may be a home subscriber network or a visited network. *See also* **visited network; home subscriber network.**

3.43 serving point of attachment (serving PoA): PoA of the current link being used by the MN.

3.44 soft handover: Handover where facilities for supporting traffic flows are continuously available while the mobile-node link-layer connection transfers from the serving point of attachment to the target point of attachment. The network allocates transport facilities to the target point of attachment prior to the occurrence of the link switch event (make-before-break).

3.45 target point of attachment (target PoA): A Candidate PoA that has been selected to become the new serving PoA.

3.46 vertical handovers: This involves MNs moving between PoAs of different link types, such as from UMTS to WLAN. *Syn:* **inter-technology handovers**

3.47 visited network: Network managed by an operator other than the subscriber's home operator and in which the subscriber is receiving service. *See also* **home subscriber network; serving network.**

4. Abbreviations and acronyms

The following abbreviations and acronyms are used in this standard.

3G	third generation
3GPP	3rd Generation Partnership Project
3GPP2	3rd Generation Partnership Project 2
AAA	authentication, authorization, and accounting
ACK	acknowledgement
AP	access point
AR	access router
AS	access stratum
BS	base station
BSC	base station controller
BSSID	basic service set identifier
BTS	base transceiver station
ESS	extended service set
CN	core network
CoA	care-of address
CoS	class of service
CS	convergence sublayer
CSME	convergence sublayer management entity
DHCP	dynamic host configuration protocol
FA	foreign agent
GGSN	gateway GPRS support node
GMM	GPRS mobility management
GPRS	general packet radio service
GSM	global system for mobile communication
HA	home agent
HESSID	homogenous extended service set ID
HIP	host identity protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	internet protocol
ISP	internet service provider
ITU	International Telecommunications Union
L1	layer 1 (PHY)
L2	layer 2 (MAC and/or LLC)
L3MMP	layer 3 mobility management protocol
LAN	local area network
LbyR	location by reference
LCP	location configuration protocol
LLC	logical link control
LSAP	logical link control service access point
LTE	long term evolution
MAC	medium access control
MAP	mobility anchor point
MICS	media independent command services
MIES	media independent event services
MIH	media independent handover
MIHF	media independent handover function
MIIS	media independent information service
MIP	mobile IP
MLME	MAC layer management entity

MMS	multimedia message service
MN	mobile node
MPLS	multi-protocol label switching
MS	mobile station
MSS	mobile subscriber station
MSDU	medium access control (MAC) service data unit
MSME	MAC common part sublayer management entity
MT	mobile terminal
N/A	not applicable
NAS	network access server
NAK	negative acknowledgement
NCMS	network control and management system
PCS	physical coding sublayer
PDSN	packet data serving node
PHY	physical layer
PLMN	public land mobile network
PLME	physical layer management entity
PoA	point of attachment
PoS	point of service
PPP	point-to-point protocol
QoS	quality of service
RAT	radio access technology
RDF	resource description framework
RFC	request for comment
RLC	radio link control
RNC	radio network controller
RRC	radio resource control
SAE	system architecture evolution
SAP	service access point
SDO	standards development organization
SDU	service data unit
SGSN	serving GPRS support node
SINR	signal over interference plus noise ratio
SLA	service level agreement
SM	session management
SME	station management entity
SNDCP	sub-network data convergence protocol
SNR	signal-to-noise ratio
SS	subscriber station
STA	station
TCP	transmission control protocol
TLV	type-length-value
UDP	user datagram protocol
UE	user equipment
UMTS	universal mobile telecommunications system
WLAN	wireless local area network
WMAN	wireless metropolitan area network
WPAN	wireless personal area network
XML	extensible mark-up language

5. General architecture

5.1 Introduction

5.1.1 General

This standard supports different handover methods. Such methods are generally classified as “hard” or “soft”, depending on whether the handover procedure is “break-before-make” or “make-before-break” with respect to the data transport facilities that support the exchange of data packets between the mobile node and the network.

Handover decision making may involve cooperative use of both mobile node and network infrastructure. Handover control, handover policies and other algorithms involved in handover decision making are generally handled by communication system elements which do not fall within the scope of this standard. However, it is beneficial to describe certain aspects of the overall handover procedure so that the role and purpose of the MIH services in the handover process are clear. The following subclauses give an overview of how the different factors that may affect handovers are addressed within this standard.

5.1.2 Service continuity

Handovers may occur either between two different access networks or between two different points of attachment of a single access network. In such cases, service continuity is defined as the continuation of the service during and after the handover while minimizing aspects such as data loss and disconnection duration during the handover without requiring any user intervention. The change of access network may or may not be noticeable to the end user, but there should be no need for the user to re-establish the service. There may be a change in service quality as a consequence of the transition between different networks due to the varying capabilities and characteristics of the access networks. For example if the quality of service (QoS) supported by the new access network is unacceptable, higher layer entities may decide not to handover or may terminate the current session after the handover based on applicable policies. This standard specifies essential elements which enable service continuity.

5.1.3 Application class

Various applications have different tolerance characteristics for delay and data loss. Application aware handover decisions may be possible by making a provision for such characteristics. For example, when a network transition due to impending handover is made during the pause phase of a conversation in an active voice call, the perceptible interruption in the service is minimized.

5.1.4 Quality of service

The quality of the service (QoS) experienced by an application depends on the accuracy, speed, and availability of the information transfer in the communication channel. This standard provides support for fulfilling application QoS requirements during handover.

There are two aspects of QoS to consider in the context of IEEE 802.21. Firstly, there is the QoS experienced by an application during a handover. Secondly, there is the QoS considered as part of a handover decision. This standard includes mechanisms that support both aspects of QoS towards enabling seamless mobility; however the MIHF alone cannot guarantee seamless mobility. Depending on the QoS requirements of the end-to-end application, seamless mobility implies minimizing the latency, and potential packet loss incurred during a handover so as to minimize the end-to-end delay and loss perceived by the application. Seamless mobility also implies the timely assessment of network conditions, such as the monitoring of packet loss on the current link and signal strength on the link of both current and target networks, in order to optimize the handover decision and its execution.

The MIH QoS model defines parameters that are used to set the requirements and assess the performance of packet transfers between a source and its destinations. When used in threshold-setting commands (such as MIH_Link_Configure_Thresholds), these parameters describe the QoS requirements of the MIH User. On the other hand, when used in parameter-reporting events (such as MIH_Link_Parameters_Report) and parameter-extraction commands (such as MIH_Get_Link_Parameters), they characterize current network conditions. Therefore, depending on their usage these parameters can represent either static QoS requirements or dynamic network measurements.

5.1.5 Network discovery

This standard defines the information that helps in network discovery and specifies the means by which such information may be obtained and be made available to the MIH Users. The network information may include information about link type, link identifier, link availability, link quality, etc.

5.1.6 Network selection

Network selection is the process by which a mobile node or a network entity selects a network (possibly out of many available) to establish network-layer connectivity. The selection may be based on various criteria such as required QoS, cost, user preferences, or the network operator's policies. This standard specifies means by which such information can be made available to the MIH Users to enable effective network selection.

5.1.7 Power management

This standard allows the MN to discover different types of networks (e.g. 802.11, 802.16 and 3GPP networks), avoiding powering-up of multiple radios and/or excessive scanning at the radios. Thus this standard minimizes power consumed by mobile devices in discovery of potential handover candidates. Specific power management mechanisms deployed are dependent on individual link-layer technologies.

5.1.8 Handover policy

The primary role of the MIHF is to facilitate handovers and provide intelligence to the network selector entity. The MIHF aids the network selector entity with the help of the Event Service, Command Service, and Information Service. The network selector entity and the handover policies that control handovers are outside the scope of this standard.

5.2 General design principles

5.2.1 MIHF design principles

This standard is based on the following general design principles.

a) MIHF is a logical entity that facilitates handover decision making. MIH Users make handover decisions based on inputs from the MIHF.

b) MIHF provides abstracted services to higher layers. The service primitives defined by this interface are based on the technology-specific protocol entities of the different access networks. The MIHF communicates with the lower layers of the mobility-management protocol stack through technology-specific interfaces.

c) Higher layer mobility management protocols specify handover signaling mechanisms for vertical handovers. Additionally, different access network technologies have defined handover signaling mechanisms to facilitate horizontal handover. The definition of such handover signaling mechanisms is outside the scope of this standard except in the case of handovers across ESSs in 802.11.

The role of this standard is to serve as a handover facilitating service and to maximize the efficiency of such handovers by providing appropriate link layer intelligence and network information.

d) The standard shall provide support for remote events. Events are advisory in nature. The decision whether to cause a handover or not based on these events is outside the scope of this standard.

e) The standard supports transparent operation with legacy equipment. IEEE 802.21 standard compatible equipment should be able to co-exist with legacy equipment.

5.2.2 QoS design principles

In the context of this standard it is assumed that applications communicate via a communication channel that is considered to be composed of several connected segments, each under a possibly different but cooperative administrative authority. Examples of such channels (e.g., for IP traffic) have been detailed in ITU-T Recommendation Y.1540.

It is generally accepted that, based on the required accuracy of information transfer, applications can be grouped into a small number of behavioral sets [Y.1540] called Classes of Service (CoS). Support for differentiation via Classes of Service is pervasive in many of the IEEE 802 based standards (IEEE 802.11, IEEE 802.1q, IEEE 802.16, etc.).

It is assumed that the classes of service definitions used within this standard conform to ITU-T recommendation Y.1540.

5.3 MIHF service overview

5.3.1 General

This standard defines services that enhance handovers between heterogeneous access links.

1. A Media Independent Event Service (MIES) which provides event classification, event filtering and event reporting corresponding to dynamic changes in link characteristics, link status, and link quality.
2. A Media Independent Command Service (MICS) which enables MIH Users to manage and control link behavior relevant to handovers and mobility.
3. A Media Independent Information Service (MIIS) which provides details on the characteristics and services provided by the serving and neighboring networks. The information enables effective system access and effective handover decisions.

The MIHF provides asynchronous and synchronous services through well-defined SAPs for link layers and MIH Users. In the case of a system with multiple network interfaces of arbitrary type, the MIH Users may use the Event Service, Command Service and Information Service provided by MIHF to manage, determine, and control the state of the underlying interfaces.

These services provided by MIHF help the MIH Users in maintaining service continuity, service adaptation to varying quality of service, battery life conservation, network discovery, and link selection. In a system containing heterogeneous network interfaces of IEEE 802 types and cellular (3GPP, 3GPP2) types, the MIHF may help the MIH Users to implement effective procedures to couple services across heterogeneous network interfaces. MIH Users may utilize services provided by the MIHF across different entities to query resources required for a handover operation between heterogeneous networks.

MIH services in mobile nodes facilitate seamless handovers between heterogeneous networks. An MIH User such as a mobility management protocol (e.g., Mobile IP) could be supported for handover and seamless session continuity. This shall not preclude other protocols in addition to Mobile IP and even other MIH Users from making use of MIH services to optimize handovers.

5.3.2 Media Independent event service

5.3.2.1 General

Events may indicate changes in state and transmission behavior of the physical, data link and logical link layers, or predict state changes of these layers. The Event Service may also be used to indicate management actions or command status on the part of the network or some management entity.

5.3.2.2 Event origination

Events may originate from the MIHF (MIH Events) or any lower layer (Link Events) within the protocol stack of a mobile node or network node.

5.3.2.3 Event destination

The destination of an event may be the MIHF or any upper layer entity. The recipient of the event may be located within the node that originated the event or within a remote node. The destination of an event is established with a subscription mechanism that enables a mobile node or network node to subscribe its interest in particular event types.

5.3.2.4 Event service flow

In the case of local events, messages often propagate from the lower layers (e.g., PHY, MAC) to the MIHF and from MIHF to any upper layer. In case of remote events, messages propagate from the MIHF in one protocol stack to the MIHF in the peer protocol stack. One of the protocol stacks may be present in a mobile node while the other may be present in a fixed network entity. This network entity may be a point of attachment or any node not directly connected to the other protocol stack.

5.3.2.5 Event service use cases and functions

The event service may be used to detect the need for handovers. For example, an indication that the link will cease to carry MAC SDUs at some point in the near future may be used by MIH Users to prepare a new point of attachment ahead of the current point of attachment ceasing to carry frames. This has the potential to reduce the time needed to handover between attachment points.

Events may carry additional context data such as an L2 identifier or L3 identifier. A Link_Up event may also carry a new IP address acquisition indication which may inform the upper layers of the need to initiate a layer 3 handover.

5.3.3 Media Independent command service

5.3.3.1 General

The command service enables higher layers to control the physical, data link, and logical link layers (also known as “lower layers”). The higher layers may control the reconfiguration or selection of an appropriate link through a set of handover commands.

5.3.3.2 Command origination

Commands may be invoked from MIH Users (MIH Commands), as well as from the MIHF itself (Link Commands), see Figure 16.

5.3.3.3 Command destination

The destination of a command may be the MIHF or any lower layer. The recipient of a command may be located within the protocol stack that originated the command, or within a remote protocol stack.

5.3.3.4 Command flow

In the case of local commands, messages often propagate from the MIH Users (e.g., policy engine) to the MIHF and then from MIHF to lower layers. In the case of remote commands, messages may propagate from MIH Users via MIHF in one protocol stack to the MIHF in a peer protocol stack (with the use of the MIH Protocol). One of the protocol stacks may be present in a mobile node while the other may be present in a fixed network entity. This network entity may be a point of attachment or any node not directly connected to the other protocol stack.

5.3.3.5 Command service use cases

The commands generally carry the upper layer decisions to the lower layers on the local device entity or at the remote entity. For example the command service may be used by the policy engine of an entity in the network to request a mobile node to switch between links (remote command to lower layers on mobile node protocol stack).

This standard facilitates both mobile-initiated and network-initiated handovers. Handovers may be initiated by changes in the wireless environment which may lead to the selection of a network that supports a different access technology other than the serving network.

During network selection, the mobile node and the network need to exchange information about available candidate networks and select the best network. The network selection policy engine may end up selecting a different network than the current one which may necessitate an inter-technology handover. Network selection and handover initiation are outside the scope of mobility management protocols such as mobile IP (MIP) and SIP. Once a new network has been selected and handover has been initiated, mobility management protocols handle packet routing aspects such as address update and transfer of packet delivery to the new network.

This standard supports a set of media independent commands which help with network selection under different conditions. These commands allow both the mobile node and the network to initiate handovers and exchange information about available networks and negotiate the best available network under different conditions. Please refer to the flow diagrams in Annex H for more information. These commands do not affect packet routing aspects and can be used in conjunction with other mobility management protocols such as MIP and SIP to perform inter-technology handovers.

5.3.4 Media Independent information service

The Media Independent Information Service (MIIS) provides a framework and corresponding mechanisms by which an MIHF entity may discover and obtain network information existing within a geographical area to facilitate the handovers.

Additionally or alternatively, the neighboring network information discovered and obtained by this framework and mechanisms can also be used in conjunction with user and network operator policies for optimum initial network selection and access (attachment), or network re-selection in idle mode.

MIIS primarily provides a set of information elements (IEs), the information structure and its representation, and a query/response type of mechanism for information transfer. This contrasts with the asynchronous push model of information transfer for the event service. The information may be present in some information server from where the MIHF in the MN may access it. The definition of the information server is outside the scope of this standard. Additionally or alternatively, information may be present locally in the MN, and may be learned by the MN or pre-provisioned, or both. The definition of and indexing of such a local database, as well as the regime for maintaining it or accessing it, are outside the scope of this standard

The information may be made available via both lower, as well as higher layers. Information may be made available at L2 through both a secure and a non-secure port. Information available through the non-secure port would allow a network selection decision to be made before incurring the overhead of authentication and the establishment of a secure L2 connection with the network.

In certain scenarios information accessible at L2 may not be available or sufficient to make an intelligent handover decision. In such cases, information accessible only via higher layers is required. Thus this standard enables both L2 and L3 transport options for this information and expects the transport to provide security for this information.

MIIS typically provides static link layer parameters such as channel information, the MAC address and security information of a PoA. Information about available higher layer services in a network may also help in more effective handover decision making before the mobile node actually attaches to any particular network.

The information provided by MIIS conforms to the structure and semantics specified within this standard. MIIS specifies a common (or media independent) way of representing this information across different technologies by using a standardized format such as extensible mark-up language (XML) or binary encoding. A structure of information is defined as a schema.

MIIS provides the ability to access information about all networks in a geographical area from any single L2 network, depending on how the IEEE 802.21 MIIS service is implemented. MIIS either relies on existing access media specific transports and security mechanisms or L3 transport and L3 security mechanisms to provide access to the information. How this information is developed and deployed in a given network is outside the scope of the standard. Typically, in a heterogeneous network composed of multiple media types, the network selector or higher layer mobility management will collect information from different media types and assemble a consolidated view to facilitate its inter-media handover decision.

Some networks such as the cellular networks already have an existing means of detecting a list of neighborhood base stations within the vicinity of an area via the broadcast control channel. Some IEEE standards define similar means and support MNs in detecting a list of neighborhood access points within the vicinity of an area via either beaconing or via the broadcast of MAC management messages. MIIS defines a unified mechanism to the higher layer entities to provide handover candidate information in a heterogeneous network environment by a given geographical location. However, the algorithm for deciding what information to provide is out of scope. In the larger scope, the objective is to help the higher layer mobility protocol to acquire a global view of the heterogeneous networks to effect seamless handover across these networks.

5.4 Media independent handover reference framework

5.4.1 General

The following subclauses describe the key points with regards to communication between different MIHF entities in the MN and the network. The reference points in this subclause are for illustration only. This subclause does not define any specific deployed network system architecture.

5.4.2 MIHF communication model

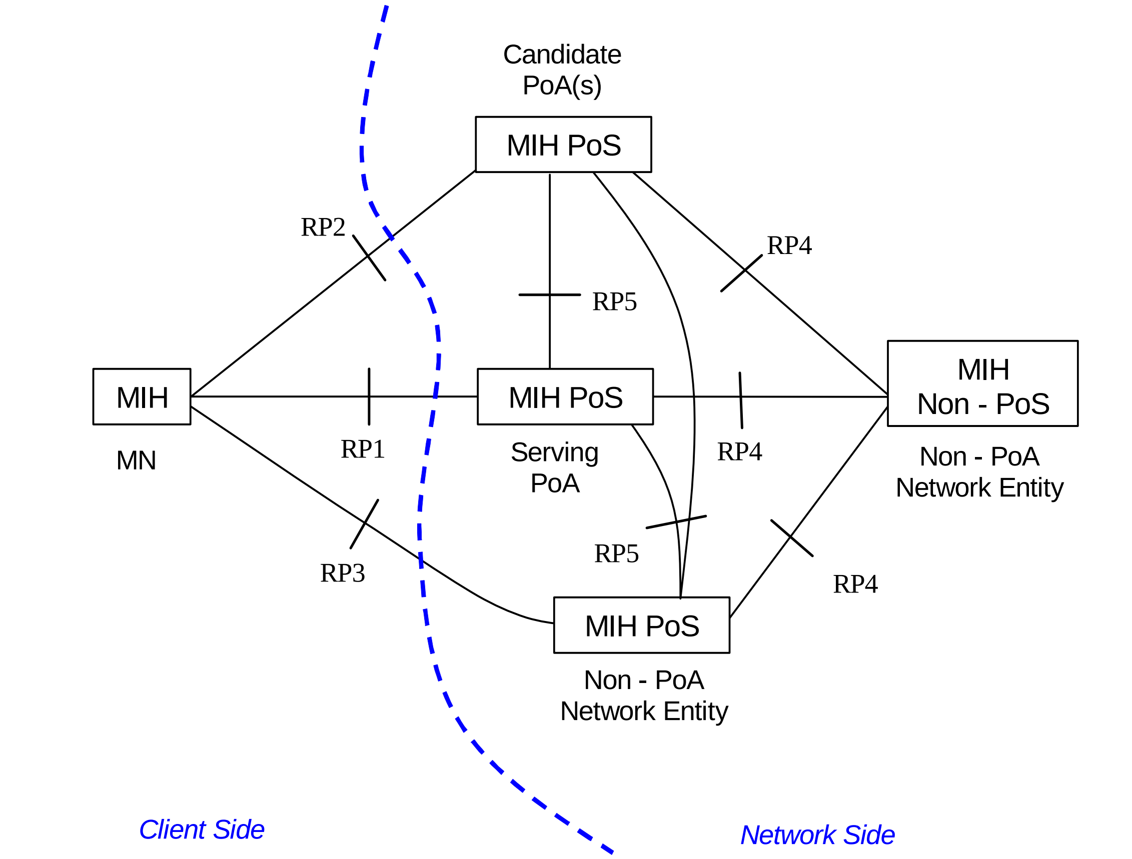


Figure 2 — MIHF communication model

MIHF functions communicate with each other for various purposes. The mobile node exchanges MIH information with its MIH Point of Service. The MIHF in any Network Entity becomes an MIH PoS when it communicates directly with an MN-based MIHF. An MIH Network Entity may not have a direct connection to the MN and therefore does not constitute an MIH PoS for that particular MN. The same MIH Network Entity may still act as MIH PoS for a different MN.

MIHF communication might not take place on all L2 interfaces of an MIH-capable MN. As an example, on an MIH-capable MN with three L2 interfaces namely IEEE 802.11, IEEE 802.16, and IEEE 802.3, the IEEE 802.3 interface might be used only for system administration and maintenance operations, while the IEEE 802.11 and IEEE 802.16 interfaces might engage in the provisioning of MIHF services. The MN may use L2 transport for exchanging MIH information with an MIH PoS that resides in the same Network Entity as its Network PoA. The MN may use L3 transport for exchanging MIH information with an MIH PoS that may not reside in the same Network Entity as its Network PoA. The framework supports use of either L2 or L3 mechanisms for communication among MIH network entities.

Figure 2 shows the MIH communication model. The model shows MIHFs in different distinctive roles and the communication relationships amongst them. The communication relationship shown in Figure 2 applies only to MIHFs. It is important to note that each of the communication relationships in the communication model does not imply a particular transport mechanism. Rather, a communication relationship only intends to show that MIHF related information passing is possible between the two distinctive MIHFs. Moreover, each communication relationship shown in the diagram may encompass different types of interfaces, different transport mechanisms used (e.g., L2, L3), and different MIHF service related content being passed (e.g., MIIS, MICS, or MIES).

The communication model assigns different roles to the MIHF depending on its position in the system.

1) MIHF on the MN
2) MIH PoS on the Network Entity that includes the serving PoA of the MN
3) MIH PoS on the Network Entity that includes a candidate PoA for the MN
4) MIH PoS on a Network Entity that does not include a PoA for the MN
5) MIH non-PoS on a Network Entity that does not include a PoA for the MN.

The communication model also identifies the following reference points between different instances of MIHFs.

1) **Reference point RP1**: Reference point RP1 refers to MIHF procedures between the MIHF on the MN and the MIH PoS on the Network Entity of its serving PoA. RP1 may encompass communication interfaces over both L2 and L3 and above. MIHF content passed over RP1 may be related to MIIS, MIES, or MICS.
2) **Reference point RP2**:Reference point RP2 refers to MIHF procedures between the MIHF on the MN and the MIH PoS on the Network Entity of a candidate PoA. RP2 may encompass communication interfaces over both L2 and L3 and above. MIHF content passed over RP2 may be related to MIIS, MIES, or MICS.
3) **Reference point RP3**: Reference point RP3 refers to MIHF procedures between the MIHF on the MN and the MIH PoS on a non-PoA Network Entity. RP3 may encompass communication interfaces over L3 and above and possibly L2 transport protocols like Ethernet bridging, or multi-protocol label switching (MPLS). MIHF content passed over RP3 may be related to MIIS, MIES, or MICS.
4) **Reference point RP4**: Reference point RP4 refers to MIHF procedures between an MIH PoS in a Network Entity and an MIH non-PoS instance in another Network Entity. RP4 may encompass communication interfaces over L3 and above. MIHF content passed over RP4 may be related to MIIS, MIES, or MICS.
5) **Reference point RP5**: Reference point RP5 refers to MIHF procedures between two MIH PoS instances in distinct Network Entities. RP5 may encompass communication interfaces over L3 and above. MIHF content passed over RP5 may be related to MIIS, MIES, or MICS.

Table 1—Summary of reference points

Reference point	Description
RP1	Between the MIHF on an MN and an MIH PoS on the Network Entity of the serving PoA.
RP2	Between the MIHF on an MN and an MIH PoS on the Network Entity of the candidate PoA.
RP3	Between the MIHF on an MN and an MIH PoS on a non-PoA network entity.
RP4	Between an MIHF PoS and an MIH non-PoS instance in distinct Network Entities.
RP5	Between two MIH PoS instances in distinct Network Entities.

All reference point definitions are within the scope of this standard.

5.4.3 A Deployment example for the MIH services

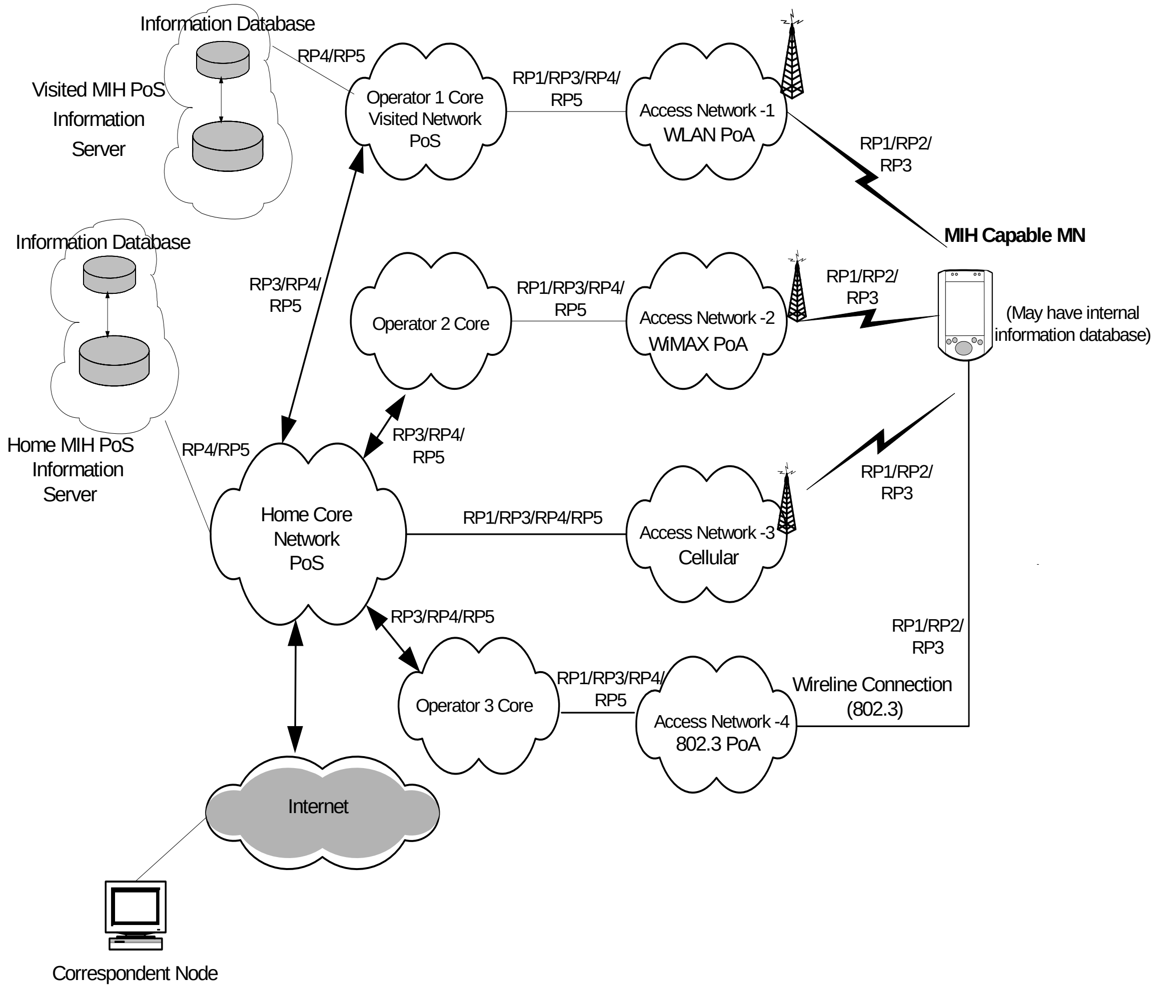


Figure 3 — Example of network model with MIH services

A network model including MIH services is shown in Figure 3 to better illustrate the MIH Reference Points. Moving from right to left, the model includes an MIH-capable mobile node (MN, far right) that supports multiple wired and wireless access technologies. The model assumes that the serving network either operates multiple link-layer technologies or allows its user to roam into other networks when a service level agreement (SLA) in support of inter-working has been established.

The model illustrates access networks that are connected in some loose, serial way to a given core network (i.e., Core Operator 1, 2, or 3). Also depicted is an access network that is more tightly coupled (Access Network-3). Each Core Operator network (1, 2, or 3) might represent a service provider, corporate intranet provider, or just another part of the visited or home access. In this depicted model the provisioning provider is operating Access Network-3, which couples the terminal to the core (labeled Home Core Network) via RP1. At any given point in time, the subscriber's serving network may be the home subscriber network or a visited network.

The network providers offer MIH services in their access networks (Access Network-1 to 4) in order to facilitate heterogeneous handovers into their networks. Each access technology either advertises its MIH capability or responds to MIH service discovery. Each service provider for these access networks allows access to one or more MIH Points of Service (PoS) node(s). These PoS nodes may provide some or all of the

MIH services as determined during the MIH capabilities discovery. The PoS location may vary based on the operator deployment scenario and the technology-specific MIH architecture.

An MIH PoS may reside next to, or co-located with, the point of attachment (PoA) node in the access network (e.g., Access Network 1, 2, 4). Alternatively the PoS may reside deeper inside the access or core networks (e.g., Access Network 3). As shown in Figure 3, the MIH entity in the MN can communicate with MIH network entities using reference points RP1, RP2, or RP3 over any of the available access network. If the PoA in the serving access network has a co-located MIHF, the RP1 reference point terminates at the PoA which is also the PoS (MN to Access Network 1, 2, 4 of the model can all be RP1). In that case an RP3 reference point would be terminated at any non-PoA (illustrated by MN connectivity to Access Networks 1, 2, 4). MIH events may originate at both sides of an active RP1 link. The MN is typically the first node to react to these events.

The interaction of visited and home subscriber networks could be either for control and management purposes or for data transport purposes. It is also possible that due to roaming or SLA agreements, the home subscriber network may allow the MN to access the public Internet directly through a visited network. As illustrated, two MIH network entities may communicate with each other via RP4 or RP5 reference points. The MIH capable PoA may also communicate with other MIH network entities via RP4 and RP5 reference points. The MIH capable MN could have an MIH communication with other PoA in the candidate access networks via RP2 reference point to obtain Information Services about the candidate network.

With regard to the MIH Information Service, visited providers can offer access to their information server located in an MIH PoS node (upper far left). The operator provides the MIIS to mobile nodes so they can obtain pertinent information including, but not limited, to new roaming lists, costs, provider identification information, provider services, priorities and any other information that would enable the selection and utilization of these services. As illustrated, it is possible for the mobile node to be pre-provisioned with MIIS data by its provider. Also possible is for the mobile node to obtain MIH Information Services from any access network of its provider or visited networks that maintain SLA agreements with the provisioner. MIIS could also be available from another overlapping or nearby visited network, using that network's MIIS point of service. The serving network may utilize RP4 and RP5 interfaces to access other MIH entities. As an example, in Figure 3 the home subscriber network may access its own MIH information server or core operator 1 (visited network) MIH information server.

5.5 MIHF reference models for link-layer technologies

5.5.1 General

The MIHF provides asynchronous and synchronous services through well-defined Service Access Points for MIH Users. The following subclauses describe the reference models for various link-layer technologies with MIH functionality.

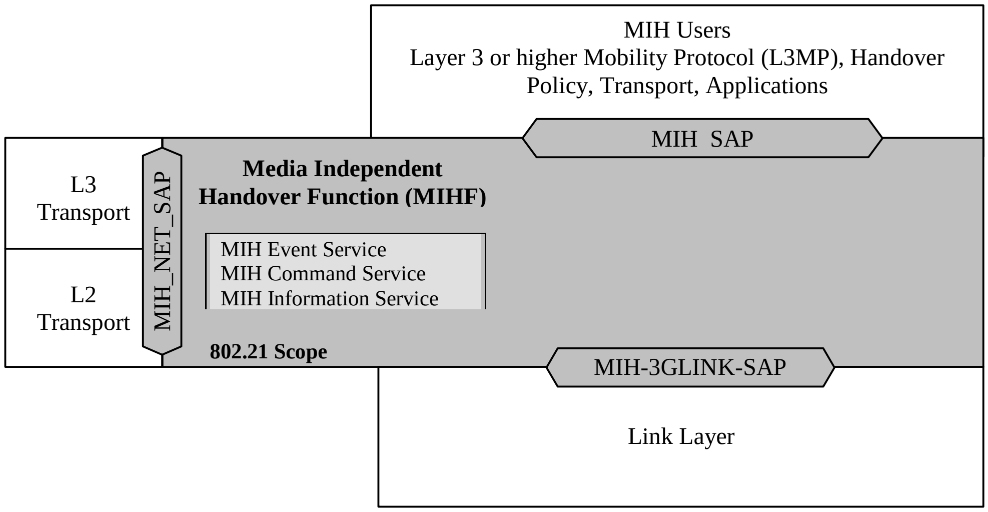


Figure 4 — General MIHF reference model and SAPs

Figure 4 illustrates the position of the MIHF in a protocol stack and the interaction of the MIHF with other elements of the system. All exchanges between the MIHF and other functional entities occur through service primitives, grouped in Service Access Points (SAPs). The media agnostic General MIH Reference Model includes the following SAPs:

a) MIH_SAP: Media independent interface of MIHF with the upper layers of the protocol stack.
b) MIH_LINK_SAP: Abstract media dependent interface of MIHF with the lower layers of the media-specific protocol stacks.
c) MIH_NET_SAP: Abstract media dependent interface of MIHF which provides transport services over the data plane on the local node, supporting the exchange of MIH information and messages with the remote MIHF. For all transport services over L2, the MIH_NET_SAP uses the primitives specified by the MIH_LINK_SAP.

In the media-specific reference models, the media independent SAP (MIH_SAP) always maintain the same name and same set of primitives. The media dependent SAP (which is a technology specific instantiation of the MIH_LINK_SAP), assumes media-specific names and sets of primitives, often reusing names and primitives that already exist in the respective media-specific existing lower-layer SAPs. Primitives defined in MIH_LINK_SAP result in amendments to media-specific SAPs due to additional functionality defined for interfacing with the MIHF. All communications of the MIHF with the lower layers of media-specific protocol stacks take place through media-specific instantiations of MIH_LINK_SAP. Refer to Annex K for the media-specific amendments.

The message exchanges between peer MIHF instances, in particular the type of transport that they use, are sensitive to several factors, such as the nature of the network nodes that contain the peer MIHF instances (whether or not one of the two is a Mobile Node or a PoA), the nature of the access network (whether IEEE 802 or 3G cellular), and the availability of MIH capabilities at the PoA.

Figure 5 presents a summary of the types of relationships that may exist between the MIHF and other functional components in the same network node.

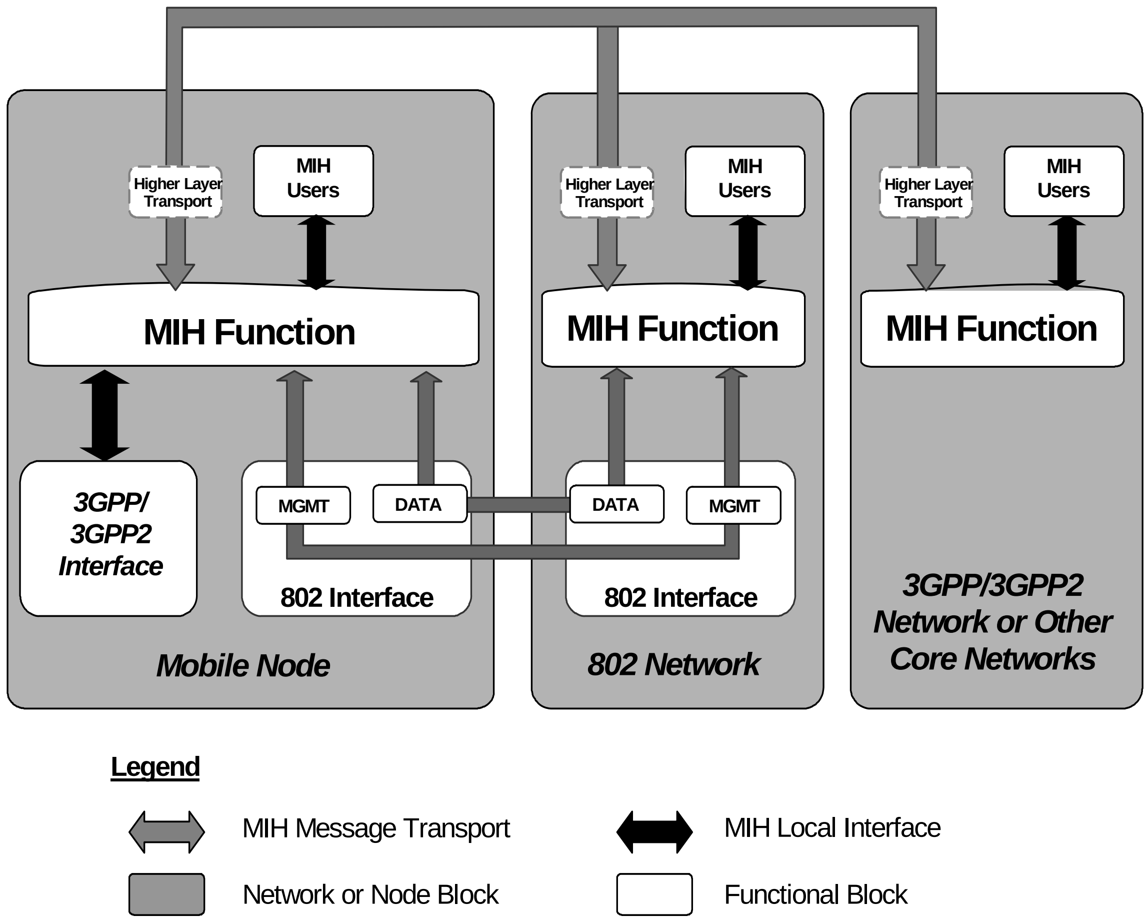


Figure 5 — Types of MIHF relationship

The general MIH reference model in Figure 4 enables a simple representation of the broad variety of MIHF relationships shown in Figure 5. In the model, a mobility-management protocol stack is logically identified within each network node that includes an MIHF instance. The abstraction provided makes it easy to isolate and represent the MIH relationships with all pre-existing functional entities within the same network node. Such relationships are both internal (with functional entities that, just like the MIHF, share the logical inclusion in the mobility-management protocol) and external (with functional entities that belong to other planes).

Figure 5 shows how an MIH-enabled Mobile Node may communicate with an MIH-enabled network. The gray arrows show the MIH signaling over the network, whereas the black arrows show local interactions between the MIHF and lower and higher layers. For a more detailed view of local interactions, please refer to technology-specific reference models and Service Access Point in the following subclauses.

When connected to an IEEE 802 network, a Mobile Node may directly use L2 for exchanging MIH signaling, as the peer MIHF may be embedded in a PoA. The MN may be able to do this for certain IEEE 802 networks even before being authenticated with the network. However, the Mobile Node may also use L3 for exchanging MIH signaling, for example in cases where the peer MIHF is not located in the PoA, but deeper in the network.

When connected to a 3GPP or 3GPP2 network, a Mobile Node may use L3 transport to conduct MIH signaling.

5.5.2 MIHF reference model for IEEE 802.3

The MIHF reference model for IEEE 802.3 is illustrated in Figure 6. The transport of MIHF services is supported over the data plane by use of existing primitives defined by the LSAP. There are no amendments specified in IEEE 802.3 to support any link services defined over the MIH_LINK_SAP in this specification.

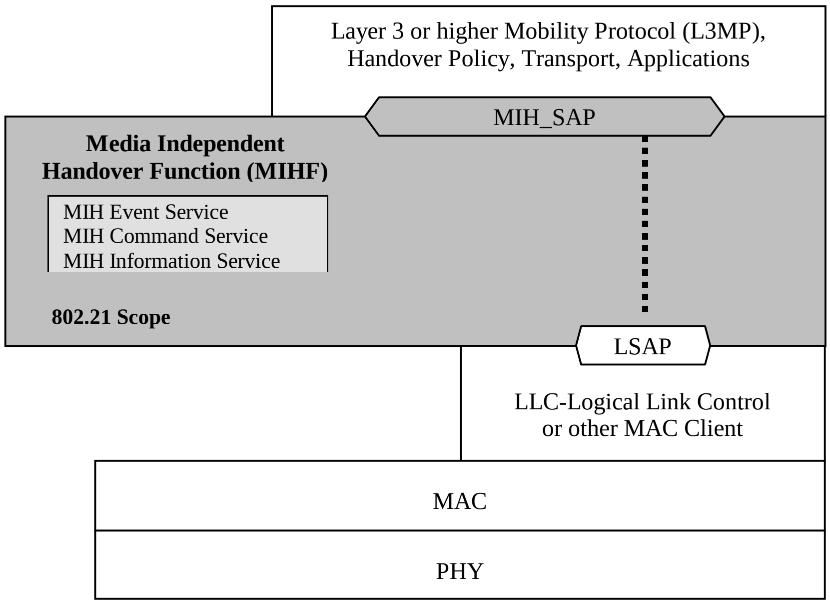


Figure 6 — MIH reference model for IEEE 802.3

5.5.3 MIHF reference model for IEEE 802.11

Figure 7 shows the MIHF reference model for IEEE 802.11 stations and network PoAs. The payload of MIHF services over IEEE 802.11 may be carried either in the data frames by using existing primitives defined by the LSAP or encapsulated in the management frames (such as Action frames) by using primitives defined in the MLME_SAP.

It should be noted that sending MIHF payload over the LSAP is allowed only in *State 3* (i.e., after successful authentication and association of the station to the AP), whereas transporting MIHF payload over the MLME_SAP is permitted in all states. Moreover, when in *State 1* (i.e., before the station has authenticated and associated with the AP), only MIH Information Service and MIH Capability Discovery messages can be transported over the MLME_SAP.

The MIH_SAP specifies the interface of the MIHF with MIH Users.

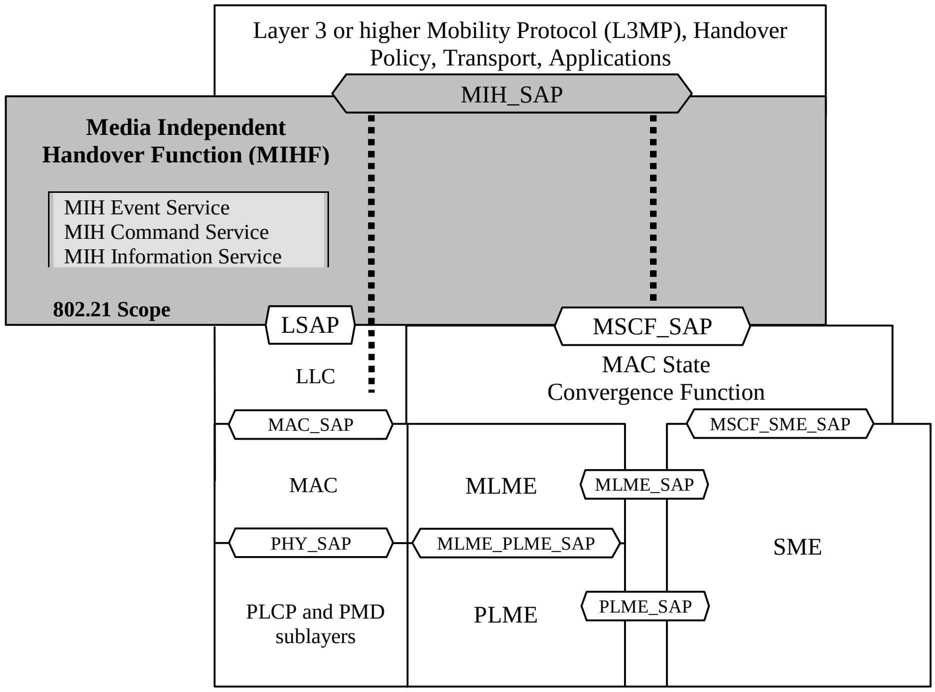


Figure 7 — MIH reference model for IEEE 802.11

5.5.4 MIHF reference model for IEEE 802.16

Figure 8 shows the MIHF for IEEE 802.16 based systems. The M_SAP and C_SAP service access points are common between the MIHF and network control and management system (NCMS). These SAPs specify the interaction between the MIHF and the control and management plane entities, respectively, and they also help in transporting MIH messages to peer MIHF entities. The C_SAP specifies the interface between the MIHF and the management plane and allows MIHF payload to be encapsulated in management messages (such as MOB_MIH-MSG defined in [B21]). Thus primitives specified by C_SAP may be used to transfer packets before and after a Mobile Node has entered a network with a base station (BS), whereas the convergence sublayer (CS) SAP may be used to transfer packets after network entry has been accomplished with a

BS. The MIH_SAP specifies the interface of the MIHF with other higher layer entities such as transport, handover policy engine, and layer 3 mobility protocol.

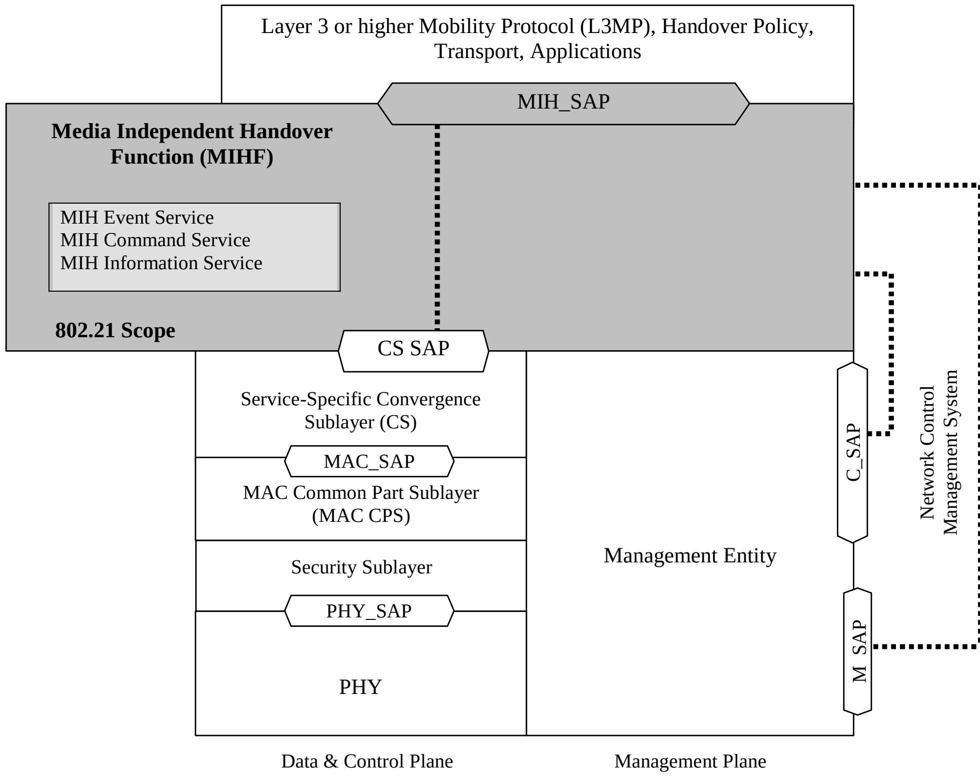


Figure 8 — MIH reference model for IEEE 802.16

In the above model, C_SAP and M_SAP provide link services defined by MIH_LINK_SAP, C_SAP provides services before network entry, while CS_SAP provides services over the data plane after network entry.

5.5.5 MIHF reference model for 3GPP

Figure 9 illustrates the interaction between the MIHF and the 3GPP based systems. The MIHF services are specified by the MIH_3GLINK_SAP. However no new primitives or protocols need to be defined in the 3GPP specification for accessing these services. The MIHF services may be easily mapped to existing 3GPP signaling functions (see Table K-3). The architectural placement of the MIHF shall also be decided by the 3GPP standard. Figure 9 is for illustrative purposes only and shall not constrain implementations.

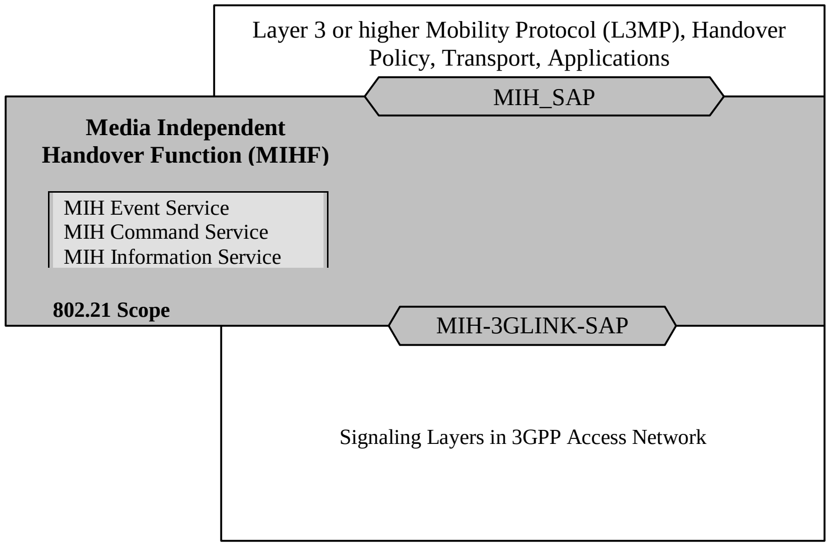


Figure 9 — MIH reference model for 3GPP systems

5.5.6 MIHF reference model for 3GPP2

Figure 10 illustrates the interaction between IEEE 802.21 services and 3GPP2 based systems. IEEE 802.21 services are accessed through the MIH-3GLINK-SAP. Note however that no new primitives or protocols need to be defined within the 3GPP2 specification. Instead, a mapping between IEEE 802.21 Link Layer primitives and 3GPP2 primitives as defined in IETF RFC 1661 and 3GPP2 C.S0004-D is already established. Primitive information available from Upper Layer Signaling and Point-to-Point Protocol (PPP) can be directly used by mapping LAC SAP and PPP SAP primitives to IEEE 802.21 service primitives in order to generate an event.

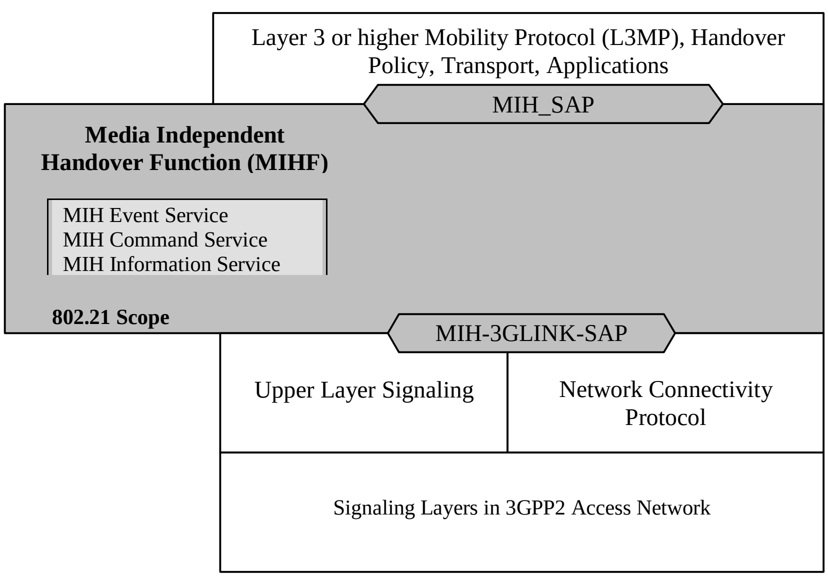


Figure 10 — MIH reference model for 3GPP2 systems

This mapping is illustrated in Table K-3 which provides an example of how 3GPP and 3GPP2 primitives can be mapped to IEEE 802.21 primitives. For example events received from the Upper Layer Signaling through the LAC layer SAP such as “L2.Condition.Notification” can be mapped and generated through the MIH-3GLINK-SAP as a Link_Up, Link_Down, or Link_Going_Down. Likewise events generated at the PPP SAP within the PPP layer, such as LCP-Link-Up or IPCP_LINK_OPEN, could be mapped and generated through the MIH-3GLINK-SAP as a Link_Up event.

It is noteworthy that there will be no direct communication between the 3GPP2 PHY and MAC layers with the MIHF. The architectural placement of any MIHF shall be left for the 3GPP2 standards development organization (SDO). Figure 10 is for illustrative purposes only and shall not constrain implementations.

5.6 Service access points (SAPs)

5.6.1 General

The MIHF interfaces with other layers and functional planes using Service Access Points (SAPs). Each SAP consists of a set of service primitives that specify the interactions between the service user and provider.

The specification of the MIHF includes the definition of SAPs that are media independent and recommendations to define or extend other SAPs that are media dependent. Media independent SAPs allow the MIHF to provide services to the upper layers of the mobility-management protocol stack, the network management plane, and the data bearer plane. The MIH_SAP and associated primitives provide the interface from MIHF to the upper layers of the mobility-management protocol stack. Upper layers need to subscribe with the MIHF as users to receive MIHF generated events and also for link layer events that originate at layers below the MIHF but may be passed on to MIH Users through the MIHF. MIH Users may directly send commands to the local MIHF using the service primitives of the MIH_SAP. Communication between two MIHFs relies on MIH protocol messages.

Media dependent SAPs allow the MIHF to use services from the lower layers of the mobility management protocol stack and their management planes. All inputs (including the events) from the lower layers of the mobility-management protocol stack into the MIHF are provided through existing media-specific SAPs such as MAC SAPs, PHY SAPs, and logical link control (LLC) SAPs. Link Commands generated by the MIHF to control the PHY and MAC layers during the handover are part of the media-specific MAC/PHY SAPs and are already defined elsewhere. Figure 11 shows the key MIHF related SAPs for different networks.

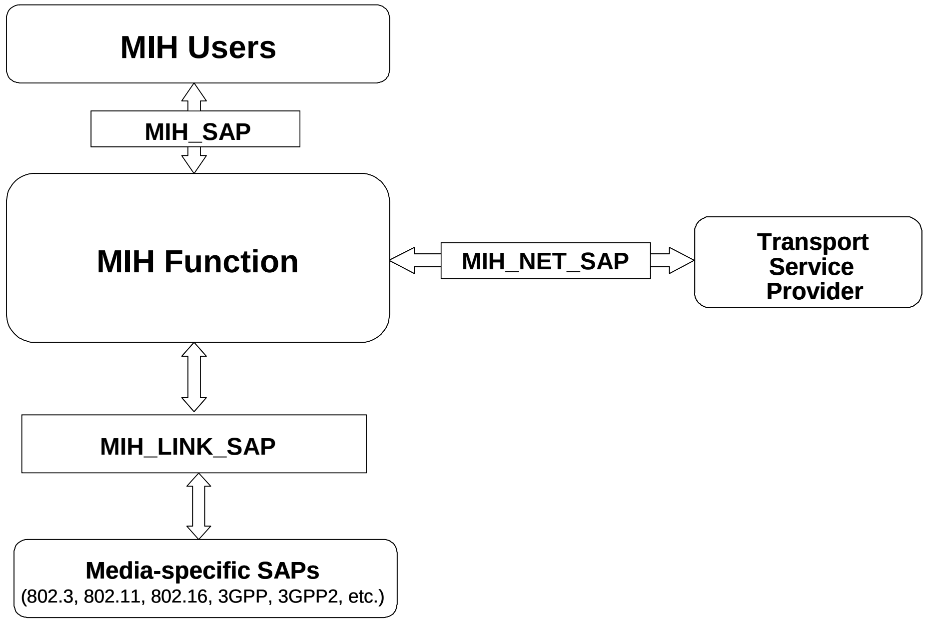


Figure 11—Relationship between different MIHF SAPs

Figure 11shows the key MIHF-related SAPs for different networks:

a) The MIH_SAP specifies a media independent interface between the MIHF and upper layers of the mobility management protocol stack. The upper layers need to subscribe with the MIHF as users to receive MIHF-generated events and also for link layer events that originate at layers below the MIHF but may be passed on to MIHF users through the MIHF. MIHF users may directly send commands to the local MIHF using the service primitives of the MIH_SAP. For communication between two remote MIHFs, the propagation from the local MIHF to the remote MIHF relies on MIH protocol messages;

b) The MIH_LINK_SAP specifies an abstract media dependent interface between the MIHF and lower layers media-specific protocol stacks of technologies such as IEEE 802.3, IEEE 802.11, IEEE 802.16, 3GPP, and 3GPP2. For different link layer technologies media-specific SAPs provide the functionality of MIH_LINK_SAP. Amendments are suggested to the respective media-specific SAPs to provide all the functionality as described by MIH_LINK_SAP;

c) The MIH_NET_SAP specifies an abstract media dependent interface of the MIHF which provides transport services over the data plane on the local node, supporting the exchange of MIH information and messages with remote MIHFs.

5.6.2 Media dependent SAPs

5.6.2.1 General

Each link layer technology specifies its own technology-dependent SAPs. For each link layer technology, the MIH_LINK_SAP maps to the technology-specific SAPs.

5.6.2.2 MIH_LINK_SAP

This SAP defines the abstract media dependent interface between MIHF and different link layer technologies. Amendments are suggested for different layer technology-specific SAPs based on the definition of this particular SAP.

5.6.2.3 MIH_NET_SAP

MIH_NET_SAP defines the abstract media dependent interface of the MIHF which provides transport services over the data plane on the local node, supporting the exchange of MIH information and messages with remote MIHFs. For L2 this SAP uses the primitives provided by MIH_LINK_SAP.

5.6.2.4 MLME_SAP

This SAP defines the interface between the MIHF and the management plane of an IEEE 802.11 network. This SAP may be used for sending MIH messages between the MIHF and local link layer entities, as well as between peer MIHF entities. Messages based on management frames may be sent in the unauthenticated state as well.

5.6.2.5 C_SAP

The C_SAP, defined in the IEEE 802.16 standard, provides the interface between the MIHF and the IEEE 802.16 control plane. This SAP is used for MIH exchanges between the MIHF and the lower layers of the management plane (as part of the IEEE 802.16 instantiation of the MIH_LINK_SAP).

5.6.2.6 MIH_3GLINK_SAP

This SAP works as an umbrella that defines the interface between the MIHF and the different protocol elements of the cellular systems. The existing service primitives or media-specific SAPs as defined in 3GPP and 3GPP2 specifications may be directly mapped to MIHF services, and hence no new primitives need to be defined in these specifications. Table K-3 lists this mapping.

5.6.2.7 LSAP

The LLC SAP (LSAP), defined in the IEEE 802.2 standard, provides the interface between the MIHF and the Logical Link Control sublayer in IEEE 802.3 and IEEE 802.11 networks. This SAP is used for local MIH exchanges between the MIHF and the lower layers of the IEEE 802.3 interface (as the IEEE 802.3 instantiation of the MIH_LINK_SAP) and for the L2 transport of MIH messages across IEEE 802.3 and IEEE 802.11access links.

5.6.2.8 CS_SAP

The CS_SAP, defined in the IEEE 802.16 standard, provides the interface between the MIHF and the service-specific Convergence Sublayer in IEEE 802.16 networks. This SAP is used for the L2 transport of MIH messages across IEEE 802.16 access links.

5.6.3 Media independent SAPs

5.6.3.1 MIH_SAP

This SAP defines the media independent interface between the MIHF and MIH Users such as an upper layer mobility protocol or a handover function which might reside at higher layers or a higher layer transport entity as well. The definition of this SAP is required to define the scope and functionality of the MIHF.

5.7 MIH protocol

5.7.1 General

MIH Protocol defines the format of messages (i.e., MIHF packet with header and payload) that are exchanged between remote MIHF entities and the media independent mechanisms that support the delivery of these messages.

5.7.2 Transport Considerations

The selection of the transport mechanism for MIH messages is dependent on the access technology that connects the MN to the network and the location of the MIH PoS.

There are two kinds of transport mechanisms: the first one is the lower layer transport (L2) and the second one is the higher layer transport (L3). Within the lower layer (L2) there may be alternate means (e.g., L2 management frames or L2 data frames) for transporting MIH messages.

If the MN cannot reach the MIH PoS via the lower layer (L2) transport mechanisms, then higher layer (L3) transport mechanisms shall be used.

6. MIHF services

6.1 General

The MIHF provides the Media Independent Event Service, the Media Independent Command Service, and the Media Independent Information Service which facilitate handovers across heterogeneous networks. This clause provides a general description of these services. These services are managed and configured through service management primitives, as discussed in 6.2.

6.2 Service management

6.2.1 General

Prior to providing the MIH services from one MIHF to another, the MIH entities need to be configured properly. This is done through the following service management functions:

- MIH Capability Discovery
- MIH Registration
- MIH Event Subscription

In order to know the services that are supported by an MIH peer, the MIH node performs MIH Capability Discovery. The MIH node may perform MIH Capability Discovery with different MIH peers in order to decide which one to register with. The following subclauses define each of these methods in more details.

6.2.2 Service management primitives

Table 2 defines the set of service management primitives.

Table 2—Service management primitives

Service Management Primitive	(L) ocal, (R) emote	Comments
MIH_Capability_Discover	L, R	Discover the capabilities of a local or peer MIHF.
MIH_Register	R	Register with a peer MIHF.
MIH_DeRegister	R	Deregister from a peer MIHF.
MIH_Event_Subscribe	L, R	Subscribe for one or more MIH events with a local or remote MIHF.
MIH_Event_Unsubscribe	L, R	Unsubscribe for one or more MIH events from a local or remote MIHF.

6.2.3 MIH capability discovery

The MIH Capability Discovery procedure is used by an MIHF entity to discover the peer MIHF's capabilities in terms of MIH Services (Event Service, Command Service, and Information Service). MIH Capability Discovery may be performed either through the MIH protocol or through media-specific Layer 2 broadcast messages (i.e., IEEE 802.11 beacons, IEEE 802.16 DCD).

6.2.4 MIH registration

MIH registration provides a mechanism for two peer MIHFs to identify and communicate with each other. The MIH registration process is symmetric, i.e., once one of the two MIHFs has initiated MIH registration and the other has responded, the two MIHF entities can symmetrically request services from each another.

For example, in a network controlled inter-technology handover framework, MIH registration can allow an MN to select an MIH PoS among many that may be available. This mechanism can work as a trigger for network MIH entities to be aware of an MN's presence. This mechanism can be an optional mechanism that enables two MIH entities to create an MIH pairing based on the policies configured in the MN or based on the information from the capability discovery. From the MN's point of view, the MIH pairing can also be seen as providing authorization for a certain network control node to send MIH commands to leverage inter-technology access resources and improve the overall user experience. MIH network entities may not be able to provide any MIH services to the MN without prior MIH registration.

As part of the MIH Service Management function, the use of MIH Registration is optional. For example, even without performing an MIH Registration, an MIHF entity in an MN may access Information Services from an MIH entity in the network without link authentication. In such an unauthenticated state only a subset of information may be available.

Figure 12 shows a reference communication message flow for MIH registration. The source MIH node performs a peer MIH discovery to determine the end point address of the peer MIHF. The source MIH node then performs MIHF capability discovery to determine what MIH services are supported by the peer MIHF. This is followed by the MIH registration to create the pairing between the two MIHFs. On successful registration, the MIHF entities can send requests for specific event subscriptions or send other MIHF commands to request specific actions to be taken.

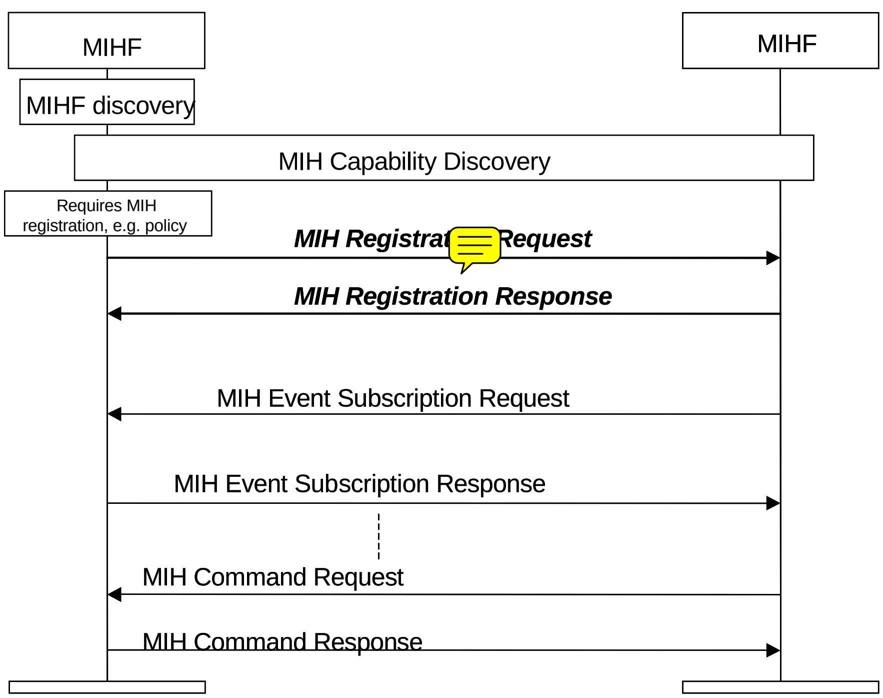


Figure 12—MIH registration flow

6.2.5 MIH event subscription

The MIH Event Subscription mechanism allows an interested MIH User to subscribe for a particular set of events that may originate from a local or peer MIHF. See Subclause 6.3.2.3 for a more detailed description of MIH event subscription.

6.2.6 Network communication

The Network Communication functions provide transport services over the data plane on the local node, supporting the exchange of MIH information and messages between the local and remote MIHF. For transport services over L2, MIH_NET_SAP utilizes the primitives specified by the MIH_LINK_SAP. For transport services over L3 the primitives are specified by MIH_NET_SAP. Please refer to Clause 7.5 for more details on MIH_NET_SAP.

6.3 Media independent event service

6.3.1 Introduction

In general, handovers may be initiated either by the mobile node or by the network. Events that may be relevant to handover may originate from MAC, PHY or MIHF at the mobile node, at the network PoA, or at the PoS. Thus, the source of these events may be either local or remote. A transport protocol is needed for supporting remote events. Security is another important consideration in such transport protocols.

Multiple higher layer entities may be interested in these events at the same time. Thus these events may need to have multiple destinations. Higher layer entities may subscribe to receive event notifications from a particular event source. The MIHF may help in dispatching these events to multiple destinations.

These events are treated as discrete events. As such there is no general event state machine. However, in certain cases a particular event may have state information associated with it, such as the Link_Going_Down event. In such cases the event may be assigned an *identifier* and other related events, such as Link_Event_Rollback, may be associated with the corresponding event using this identifier. Event notifications are generated asynchronously. Thus, all MIH Users and MIHFs that want to receive event notifications need to subscribe to particular events.

From the recipient's perspective these events are mostly "advisory" in nature and not "mandatory". In other words, the recipient is *not* obligated to act on these events. Layer 3 and above entities may also need to deal with reliability and robustness issues associated with these events. Higher layer protocols and other entities may prefer to take a more cautious approach when events originate remotely as opposed to when they originate locally. These events may also be used for horizontal handovers.

The Event Service is broadly divided into two categories, Link Events and MIH Events. Both Link and MIH Events may traverse from a lower to a higher layer. Link Events are defined as events that originate from event source entities below the MIHF and may terminate at the MIHF. Entities generating Link Events include, but are not limited to, various IEEE 802-defined, 3GPP-defined, and 3GPP2-defined interfaces. Within the MIHF, Link Events may be further propagated, with or without additional processing, to MIH Users that have subscribed for the specific events. MIH events are defined as events created within the MIHF, or Link Events that are propagated by the MIHF to the MIH Users. The MIH user cannot distinguish propagated link events from MIHF internally created events. This relationship is shown in Figure 13.

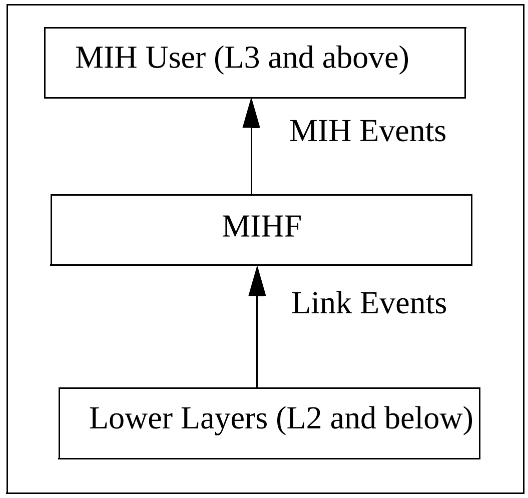


Figure 13 — Link events and MIH events

Local events are propagated across different layers within the local protocol stack of a device. All link events (from lower layer to MIHF) are local in nature. Remote events are indications that traverse across the network medium from one MIHF to a peer MIHF. MIH events may be local or remote. They traverse the medium to the local MIHF and are then dispatched to MIH Users that have subscribed to these events within the local protocol stack as shown in Figure 14.

This is with the assumption that the local upper layer entities have subscribed for the remote event. Link events that are received by the MIHF may also be sent to a remote entity as an MIH event.

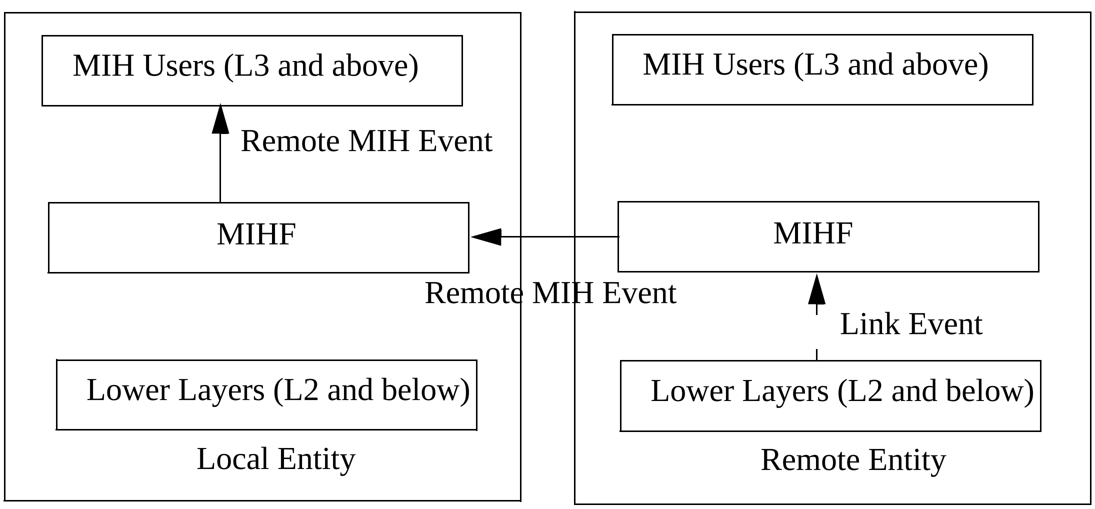


Figure 14 — Remote MIH events

6.3.2 Event subscription

6.3.2.1 General

Event Subscription provides a mechanism for upper layer entities to selectively receive events. Event Subscription may be divided into Link Events Subscription and MIH Events Subscription. Link Events Subscription is performed by the MIHF with the event source entities in order to determine the events that each event source (link) is able to provide. MIH Event Subscription is performed by upper layer entities with the MIHF to select the events to receive. It is possible for upper layer entities to subscribe for all existing events or notifications that are provided by the event source entity even if no additional processing of the event is done by the MIHF.

6.3.2.2 Link events subscription

It may be possible during initialization that the MIHF actively searches for pre-existing interfaces, devices and modules that serve as link event sources in the Event service. In addition to the link event source entities that are present during the bootstrapping stage, allowances may be made for devices such as hot-plugged interfaces or an external module. The exact description and implementation of such mechanisms are out of scope of the standard. The MIHF may then subscribe individually with each of these link layers based on user preferences.

6.3.2.3 MIH events subscription

MIH Users may specify a list of events for which they wish to receive notifications from the MIHF. For an MIH event that can originate both locally and remotely, an MIH User may specify whether it is subscribing for the local event only, remote event only, or both. MIH Users may specify additional parameters during the subscription process in order to control the behaviour of the Event Service. Examples of these additional parameters include threshold values and requests to receive events as a bundle. MIH Users may also query on the availability of events from the MIHF without prior subscription. This query may have additional parameters. An example would be a query to retrieve a list of all events for IEEE 802.11 defined interfaces only. If the MIH event that an MIH User is subscribing for is not supported or is not available, the MIHF shall be able to reject the subscription and notify the MIH User.

6.3.3 Event service flow model

Figure 15 shows the event flow model for link events and MIH events.

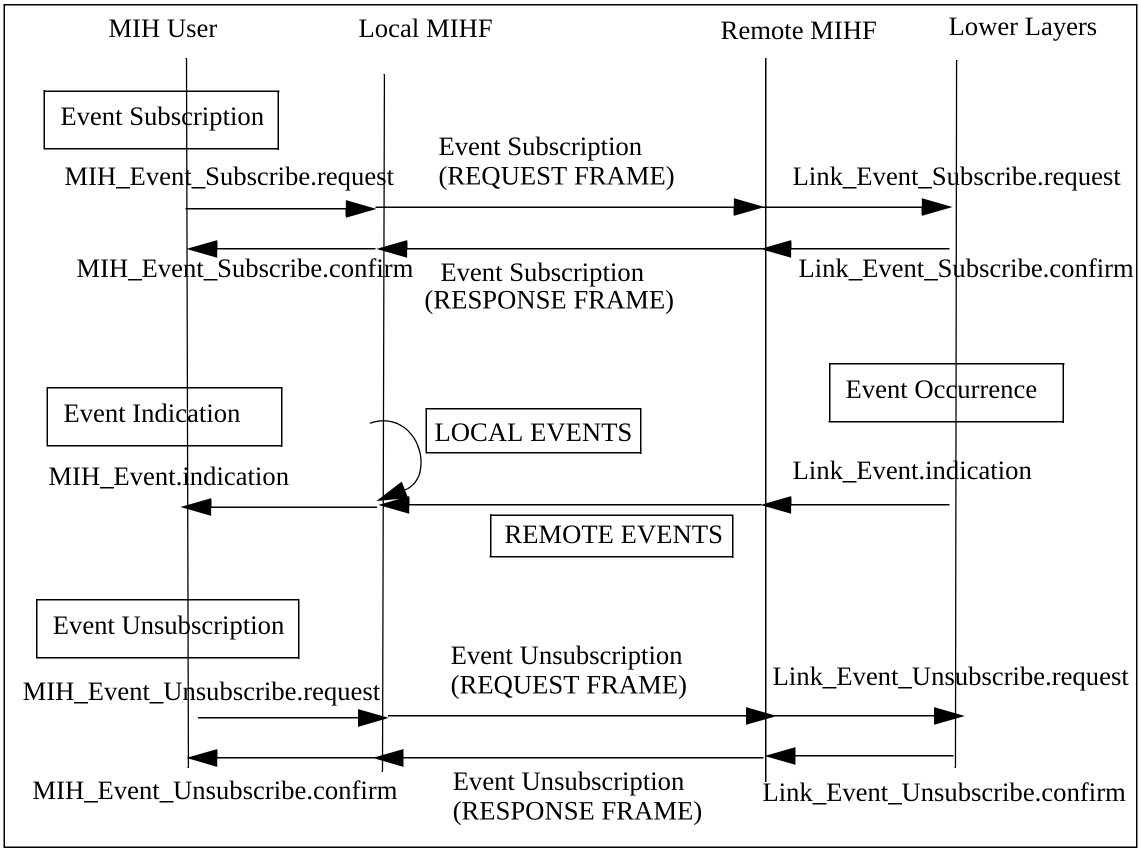


Figure 15 — MIH events subscription and flow

6.3.4 Link events

6.3.4.1 General

The Media Independent Event Service supports several categories of link events:

1) **MAC and PHY State Change events**: These events correspond to changes in MAC and PHY state. For example Link_Up event is an example of a state change event.

2) Link Parameter events: These events are due to changes in link layer parameters. For example, the primitive Link_Parameters_Report is a Link Parameter event.

3) **Predictive events**: Predictive events convey the likelihood of a change in the link properties in the near future based on past and present conditions. For example, decay in signal strength of a WLAN network may indicate a loss of link connectivity in the near future. In case predictive events are incorrect they may be retracted.

4) **Link Handover events**: These events inform upper layers about the occurrence of L2 handovers/link switches if supported by the given media type.[1]

5) **Link Transmission events**: These events indicate the link layer transmission status (e.g., success or failure) of upper layer PDUs. This information may be used by upper layers to improve buffer management for minimizing the upper layer data loss due to a handover.

For example, the occurrence of a handover of an MN from one access network to another will result in the tear-down of the old link layer connection between the MN and the source access network and the establishment of a new link layer connection between the MN and the target access network. When this occurs, some upper layer PDUs may become “trapped” at the old link - including PDUs that had been queued at the old link but never been transmitted before the link was torn-down (i.e., unsent PDUs), and PDUs that has been transmitted over the old link but never been fully acknowledged by the upper layer receiver before the link was torn-down (i.e., unacked PDUs). These “trapped” PDUs will be discarded when the old link is torn-down. As a result, unless the upper layer sender attempts to retransmit them over the new link connection, these upper layer PDUs will never reach the receiver.

However, before the retransmission can happen, the upper layer sender first needs to be notified about which upper layer PDUs are “trapped” at the old link. Link Transmission events facilitate this process by providing the upper layer a local indication on whether a particular upper layer PDU has been successfully transmitted or not.

The following set of link layer events is defined:

Table 3—Link events

Link event name	Link event type	Description
Link_Detected	State Change	New link has been detected
Link_Up	State Change	L2 connection is established and link is available for use
Link_Down	State Change	L2 connection is broken and link is not available for use
Link_Event_Rollback	State Change	Previous link event needs to be rolled back
Link_Parameters_Report	Link Parameters	Link parameters have crossed specified threshold
Link_Going_Down	Predictive	Link conditions are degrading and connection loss is imminent
Link_Handover_Imminent	Link Handover	L2 handover is imminent based on changes in link conditions
Link_Handover_Complete	Link Handover	L2 link handover to a new PoA has been completed
Link_PDU_Transmit_Status	Link Transmission	Indicate transmission status of a PDU

6.3.4.2 Characteristics of events

The Link Detected event is generated on the MN when the first PoA of an access network is detected. This event is not generated when subsequent PoAs of the same access network are discovered. In general when a

[1]The mechanism that triggers and executes a link layer handover/switch (also referred as an L2 handover) is specified within the corresponding media-specific standard and out of scope of this standard.

link event occurs due to a change in link condition it is not known at that instant if this would lead to intra-technology handover or inter-technology handover. That determination is done higher up in the protocol stack by the network selection entity based on variety of other factors. As such certain link layer events such as Link_Going_Down may lead to either intra-technology or inter-technology handovers. The network selection entity may try to maintain the current connection, by first trying intra-technology handovers and only later on resort to inter-technology handovers. Link_Up and Link_Down are discrete events and there is no state machine associated between them. The Link_Parameters_Report event occurs whenever the link parameters cross pre-specified thresholds.

6.3.5 MIH events

The following set of MIH events is defined:

Table 4—MIH events

MIH event name	MIH event type	(L) ocal (R) emote	Description
MIH_Link_Detected	State Change	L, R	New link has been detected.
MIH_Link_Up	State Change	L, R	L2 connection is established and link is available for use.
MIH_Link_Down	State Change	L, R	L2 connection is broken and link is not available for use.
MIH_Link_Event_Rollback	State Change	L, R	Previous link event needs to be rolled back.
MIH_Link_Parameters_Report	Link Parameters	L, R	Link parameters have crossed a specified threshold and need to be reported.
MIH_Link_Going_Down	Predictive	L, R	Link conditions are degrading and connection loss is imminent.
MIH_Link_Handover_Imminent	Link Handover	L, R	L2 handover is imminent based on either the changes in the link conditions or additional information available in the network. For example, the network may decide that an application requires a specific QoS that can be best provided by a certain access technology.
MIH_Link_Handover_Complete	Link Handover	L, R	L2 link handover to a new PoA has been completed.
MIH_Link_PDU_Transmit_Status	Link Transmission	L	Indicate transmission status of a PDU.

6.4 Media independent command service

6.4.1 Introduction

Media Independent Command Service (MICS) refers to the commands sent from MIH Users to the lower layers in the reference model. MIH Users may utilize command services to determine the status of links and/or control the multi-mode device for optimal performance. Command services may also enable MIH Users to facilitate optimal handover policies. For example, the network may initiate and control handovers to balance the load of two different access networks.

The link status varies with time and mobile node mobility. Information provided by MICS is dynamic information comprised of link parameters such as signal strength and link speed, where as information provided by MIIS is less dynamic or static in nature and is comprised of parameters such as network operators and higher layer service information. MICS and MIIS information could be used in combination by the mobile node/network to facilitate the handover.

A number of commands are defined in this standard to allow the MIH Users to configure, control, and retrieve information from the lower layers including MAC, Radio Resource Management, and PHY. The commands are classified into two categories: MIH Commands and Link Commands. Figure 16 shows link commands and MIH commands.

The receipt of certain MIH command requests may cause event indications to be generated. The receipt of MIH command requests may indicate a future state change in one of the link layers in the local node. These indications notify subscribed MIH Users of impending link state changes. This allows MIH Users to be better prepared to take appropriate action.

Link Commands originate from the MIHF and are directed to the lower layers. These commands mainly control the behavior of the lower layer entities. Link Commands are local only. Link Commands are issued by the MIHF to lower layers in the local protocol stack. Whenever applicable this standard encourages use of existing media-specific link commands for interaction with specific access networks. New link commands, if required, are defined as recommendations to different link layer technology standards. It is to be noted that although Link Commands originate from the MIHF, these commands are executed on behalf of the MIH Users.

The MIH commands are generated by the MIH Users and sent to the MIHF. MIH commands can be local or remote. Local MIH commands are sent by MIH Users to the MIHF in the local protocol stack.

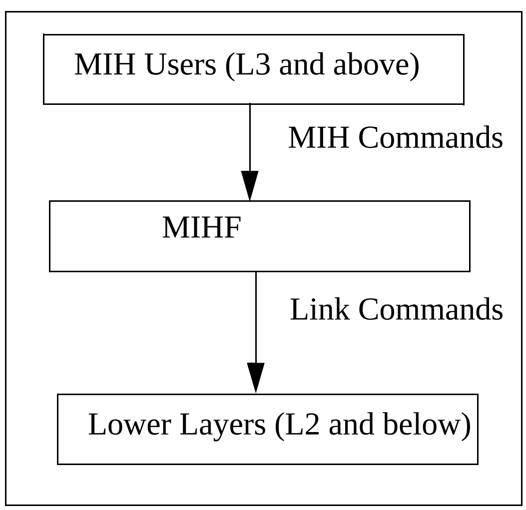


Figure 16 — Link commands and MIH commands

Remote MIH commands are sent by MIH Users to the MIHF in a peer protocol stack. A remote MIH command delivered to a peer MIHF is executed by the lower-layers under the peer MIHF as a link command; or is executed by the peer MIHF itself as an MIH command (as if the MIH command came from an MIH User of the peer MIHF). Often, an MIH indication to a remote MIH User may result from the execution of the MIH command by the peer MIHF. Figure 17 shows remote MIH commands

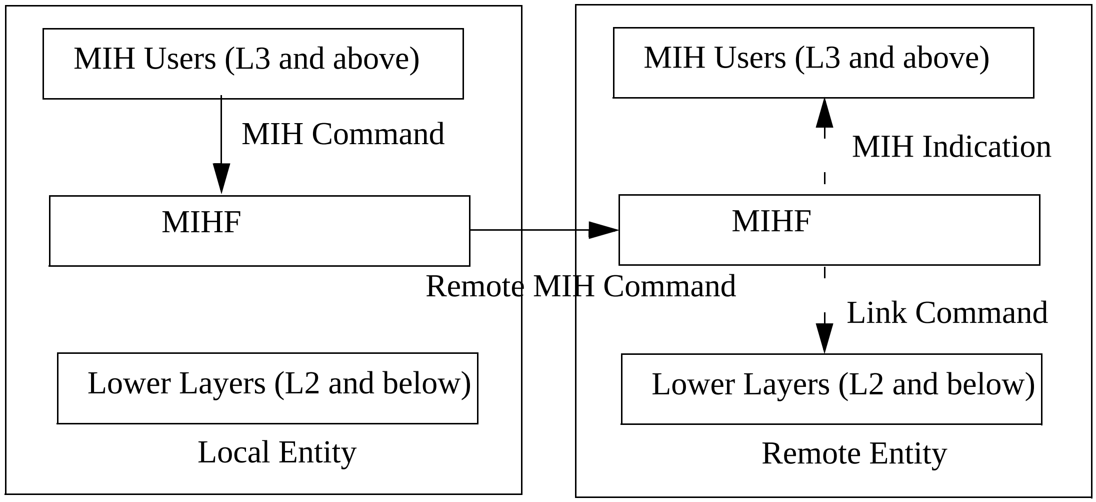
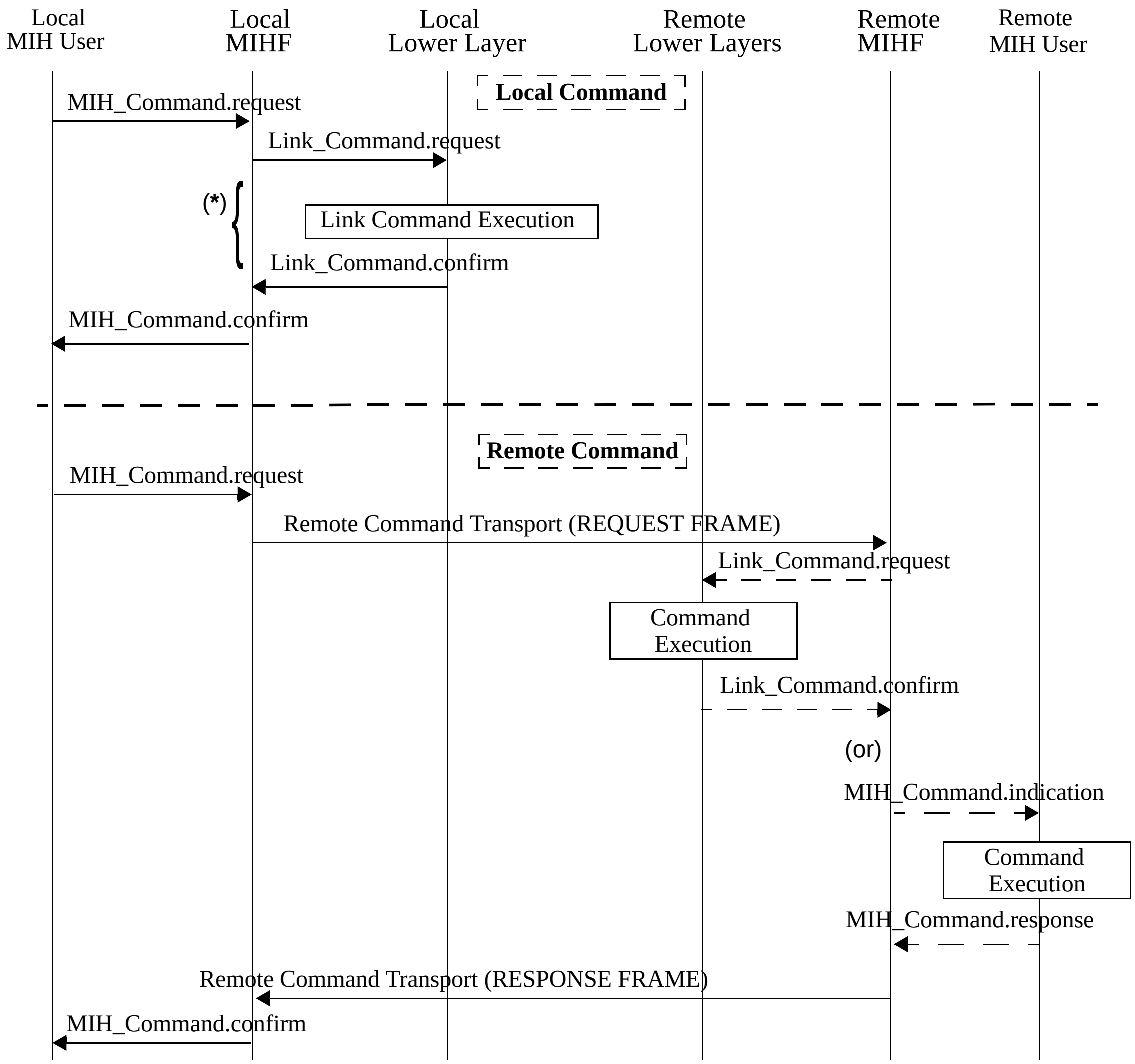


Figure 17 — Remote MIH command

6.4.2 Command service flow model



(*) There might be no corresponding Link_Command primitives and one or more media-specific link primitives can be used here.

Figure 18 — Command service flow

Figure 18 shows the flow for a local command and an example of a remote command, respectively. Example handover procedures using the commands defined in this subclause can be found in Annex H. Remote commands may be transported over network layer protocols or link layer protocols.

6.4.3 Command list

6.4.3.1 MIH commands

6.4.3.1.1 General

The following set of MIH Commands is defined:

Table 5—MIH commands

MIH command	(L) ocal, (R) emote	Comments
MIH_Get_Link_Parameters	L, R	Get the status of a link.
MIH_Link_Configure_Thresholds	L, R	Configure link parameter thresholds
MIH_Link_Actions	L, R	Control the behavior of a set of links
MIH_Net_HO_Candidate_Query	R	Network may initiate handover and send a list of suggested networks and associated Points of Attachment.
MIH_MN_HO_Candidate_Query	R	Command used by MN to query and obtain handover related information about possible candidate networks.
MIH_N2N_HO_Query_Resources	R	This command is sent by the serving MIHF entity to the target MIHF entity to allow for resource query, context transfer (if applicable), and handover preparation.
MIH_Net_HO_Commit	R	In this case the network commits to do the handover and sends the choice of selected network and associated PoA.
MIH_MN_HO_Commit	L	Command used by MN to notify the network that a candidate has been committed for handover.
MIH_N2N_HO_Commit	R	Command used by a serving network to inform a target network that a mobile node is about to move toward that network.
MIH_MN_HO_Complete	R	Notification from MIHF of the MN to the target or source MIHF indicating the status of handover completion.
MIH_N2N_HO_Complete	R	Notification from either source or target MIHF to the other (i.e. peer) MIHF indicating the status of the handover completion.

6.4.3.1.2 Naming convention for MIH handover commands

Generally, there are three types of MIH handover command primitives based on the functionality specified for the following scenarios: i) MN to Network, ii) Network to MN, and iii) Network to Network. This classification helps to ensure the specification of right protocol functionality and the relevant parameters for specific use as determined by the origination and the destination points.

Accordingly, these commands have a naming convention that identifies the origination point in the primitive name, as shown in Table 6. This convention shall be followed by the MIHF to ensure that these commands are utilized for the intended purpose. The destination point applies for remote commands only.

Table 6—Naming convention for MIH handover command primitives

Primitive Name Prefix	Originating Point	Destination Point
MIH_MN_HO_***	Mobile Node	Network
MIH_Net_HO_***	Network	Mobile Node
MIH_N2N_HO_***	Network	Network

6.4.3.2 Link commands

The following set of Link commands is defined:

Table 7—Link commands

Link command	Comments
Link_Capability_Discover	Query and discover the list of supported link layer events and link layer commands.
Link_Event_Subscribe	Subscribe to one or more events from a link.
Link_Event_Unsubscribe	Unsubscribe from a set of link layer events.
Link_Configure_Thresholds	Configure thresholds for Link Parameters Report event.
Link_Get_Parameters	Get parameters measured by the active link, such as signal-to-noise ratio (SNR), BER, RSSI
Link_Action	Request actions on a link layer connection

6.5 Media independent information service

6.5.1 Introduction

Media Independent Information Service (MIIS) provides a framework by which an MIHF, residing in the mobile node or in the network, may discover and obtain network information within a geographical area to facilitate network selection and handovers. The objective is to acquire a global view of all the heterogeneous networks relevant to the MN in the area to facilitate seamless roaming across these networks.

Media Independent Information Service includes support for various Information Elements (IEs). Information Elements provide information that is essential for a network selector to make intelligent handover decisions.

Depending on the type of mobility, support for different types of information elements may be necessary for performing handovers. MIIS provides the capability for obtaining information about lower layers such as neighbor maps and other link layer parameters, as well as information about available higher layer services such as internet connectivity.

MIIS provides information about different access networks such as IEEE 802 networks, 3GPP networks and 3GPP2 networks. The MIIS also allows this collective information to be accessed from any single network. For example, by using an IEEE 802.11 access network it may be possible to get information not only about all other IEEE 802 based networks in a particular region but also about 3GPP and 3GPP2 networks. Similarly using a 3GPP2 interface, it may be possible to get access to information about all IEEE 802 and 3GPP networks in a given region. This capability allows the mobile node to use its currently active access network and inquire about other available access networks in a geographical region. Thus a mobile node is freed from the burden of powering up each of its individual radios and establishing network connectivity for the purpose of retrieving heterogeneous network information. MIIS enables this functionality across all available access networks by providing a uniform way to retrieve heterogeneous network information in any geographical area.

The main goal behind the Information Service is to allow mobile node and network entities to discover information that may influence the selection of appropriate networks during handovers. This information is intended to be primarily used by a policy engine entity that may make effective handover decisions based on this information. This Information Service is expected to provide mostly static information, although network configuration changes shall also be accounted for. Other dynamic information about different access networks, such as current available resource levels, state parameters, and dynamic statistics should be obtained directly from the respective access networks. Some of the key motivations behind the Information Service are as follows:

1) Provide information about the availability of access networks in a geographical area. Further, this information could be retrieved using any wireless network, for example, information about a nearby Wi-Fi hotspot could be obtained using a global system for mobile communication (GSM), CDMA, or any other cellular network, whether by means of request/response signaling, or by means of information that is specifically or implicitly broadcast over those cellular networks. Alternatively, this information could be maintained in an internal database on the MN.
2) Provide static link layer information parameters that could help the mobile nodes in selecting the appropriate access network. For example knowledge of whether security and QoS are supported on a particular access network may influence the decision to select such an access network during handovers.
3) Provide information about capabilities of different PoAs in neighbor reports to aid in configuring the radios optimally (to the extent possible) for connecting to available/selected access networks. For example knowing about supported channels by different PoAs may help in configuring the channels optimally as opposed to scanning or beaconing and then finding out this information. However, for the most part, dynamic link layer parameters have to be obtained or selected based on direct interaction with the access networks as the Information Service may not be able to help much in this regard.
4) Provide an indication of higher layer services supported by different access networks and core networks that may aid in making handover decisions. Such information may not be available (or could not be made available) directly from the MAC sublayer or PHY of specific access networks, but could be provided as part of the Information Service. For example in certain cases classification of different networks into categories such as public, enterprise, home, and others may influence a handover decision. Other information here may be more vendor/network specific in nature and could be specified in that form.

6.5.2 Access information service before authentication

It is important to note that, with certain access networks an MN should be able to obtain IEEE 802.21 related information elements before the MN is authenticated with the PoA. These information elements may be used by the handover policy function to determine if the PoA can be selected. In order to enable the information query before authentication, individual link technologies may provide an L2 or media-specific transport or a protocol message exchange that makes this MIIS query exchange possible between the user equipment

(MN) and a certain MIHF in the network. For example in IEEE 802.11, a *State 1* management action frame exchange is available as a transport means for MIIS information query. IEEE 802.16 also allows an MN to query using PKM management message before authentication. The MIHF in the MN discovers the MIH capability support from the PoA through the media-specific broadcast information containing the system capabilities. It should be noted that the pre-authentication query facility is provided only for MIH information query and cannot be used for carrying other MIH protocol services except MIES and/or MICS capability discovery query using MIH_Capability_Discover embedded into L2 management frames. Additionally, any MIHF within the network may request for the set of information elements from a peer MIHF located in the same or a different network using the MIH protocol.

After authentication and attachment to a certain PoA, the MIH protocol may be used for information retrieval by use of data frames specific to that media technology.

In any case, the MIHF should have the knowledge of whether or not a network supports this standard, and may obtain this knowledge by means of media independent or media-specific discovery mechanisms.

6.5.3 Restricting query response size

When sending an information query request, the MIIS client may provide a maximum response size to limit the query response message size. A request may contain multiple queries. If the request contains multiple queries, they shall be in the order of significance to the client. In case the query results exceed the maximum response size, the least significant query results shall be removed from the response. The MIIS server may have its own maximum response size limit configured, that is smaller than is the one specified by the MIIS client request. In this case the response message shall return results in the order of significance to the client up to that limit.

6.5.4 Information elements

The Information Service elements are classified into three groups:

1) General Information and Access Network Specific Information: These information elements give a general overview of the different networks providing coverage within an area. For example, a list of available networks and their associated operators, roaming agreements between different operators, cost of connecting to the network and network security and quality of service capabilities.

2) PoA Specific Information: These information elements provide information about different PoAs for each of the available access networks. These IEs include PoA addressing information, PoA location, data rates supported, the type of PHY and MAC layers and any channel parameters to optimize link layer connectivity. This may also include higher layer services and individual capabilities of different PoAs.

3) Other Information may be vendor/network specific, and could be specified appropriately.

Table 8 represents the list of Information Elements and their semantics. Each Information Element has an abstract data type. The binary and RDF representation of these Information Elements are described in 6.5.6.2 and 6.5.6.3, respectively. The IEs may be retrieved using TLV or SPARQL based query methods. The standard does not recommend or mandate the choice of either method. An IEEE 802.21 implementation that implements the MIIS shall implement at least one method. Vendors or network operators may need to define additional IEs beyond the IEs specified in Table 8. Vendors and network operators can implement new IEs using the Vendor Specific IEs. These IEs will then be available only in vendor or operator specific deployments.

Table 8—Information elements

Name of information element	Description	Data type name
General Information Elements		
TYPE_IE_NETWORK_TYPE	Link types of the networks that are available in a given geographical area.	NETWORK_TYPE
TYPE_IE_OPERATOR_IDENTIFIER	The operator identifier for the access network/core network.	OPERATOR_ID
TYPE_IE_SERVICE_PROVIDER_IDENTIFIER	Identifier for the service provider.	SERVICE_PROVIDER_ID
TYPE_IE_COUNTRY_CODE	Indicate the country.	COUNTRY_CODE
Access Network Specific Information Elements		
TYPE_IE_ACCESS_NETWORK_IDENTIFIER	Identifier for the access network.	ACCESS_NETWORK_ID
TYPE_IE_ACCESS_NETWORK_AUX_ID	An auxiliary access network identifier. As an example for IEEE 802.11 this refers to the HESSID.	ACCESS_NETWORK_AUX_ID
TYPE_IE_ROAMING_PARTNERS	Roaming Partners. Network Operators with which the current network operator has direct roaming agreements.	ROAMING_PARTNERS
TYPE_IE_COST	Cost. Indication of cost for service or network usage.	COST
TYPE_IE_NETWORK_SECURITY	Security characteristics of the link layer.	NETWORK_SECURITY
TYPE_IE_NETWORK_QOS	QoS characteristics of the link layer.	QOS_LIST
TYPE_IE_NETWORK_DATA_RATE	Data Rate. The maximum value of the data rate supported by the link layer of the access network.	DATA_RATE
TYPE_IE_NETWORK_CHANNEL_RANGE	Channel Range/Parameters. Spectrum range supported by the network.	CHANNEL_RANGE
TYPE_IE_NETWORK_IP_CONFIG_METHODS	IP Configuration Methods supported by the access network	IP_CONFIG_METHODS
TYPE_IE_NETWORK_CAPABILITIES	Bitmap of access network capabilities	NETWORK_CAPABILITIES
TYPE_IE_NETWORK_SUPPORTED_LCP	List of location configuration protocols supported by the access network	SUPPORTED_LCP
TYPE_IE_NETWORK_MOBILITY_MGMT_PROT	Type of mobility management protocol supported	IP_MOBILITY_MGMT

Table 8—Information elements

Name of information element	Description	Data type name
PoA Specific Information Elements		
TYPE_IE_POA_MAC_ADDRESS	MAC Address of PoA	MAC_ ADDRESS
TYPE_IE_POA_LOCATION	Geographical location of PoA. Multiple location types are supported including coordinate-based location information, civic address, and cell ID.	LOCATION
TYPE_IE_POA_CHANNEL_RANGE	Channel Range/Parameters. Spectrum range supported by the channel for that PoA.	CHANNEL_RANGE
TYPE_IE_POA_SYESTEM_INFORMA TION	System information supported by the link layer of a given PoA	SYSTEM_INFORMAT ION
PoA Specific Higher Layer Service Information Elements		
TYPE_IE_POA_SUBNET_INFORMATI ON	Information about subnets supported by a typical PoA	IP_SUBNET_INFO
TYPE_IE_POA_IP_ADDRESS	IP Address of PoA.	IP_ADDRESS
Other Information Elements		
Vendor Specific IEs	Vendor Specific Services	N/A

In certain access network deployments, some PoA properties (e.g., data rate, IP configuration methods, capabilities) may be common for all PoAs within that access network. In such a case the common PoA properties are represented as IEs as part of the access network property information.

As an example, Figure 19 shows the layout of different Information Elements and the neighbor map of different networks in a geographical area. Multiple operators can be providing support for a particular network. Thus support for IEEE 802.11 network is provided by both Operator_1 and Operator_2. A single operator may provide support for multiple networks. Thus Operator_1 provides support for IEEE 802.11 and UMTS networks while Operator_3 provides support for IEEE 802.16 and UMTS networks. The General Network Information Elements are specified for each network supported by an operator. Thus in the case of Operator_1, General Network Information is specified for both IEEE 802.11 and UMTS networks, while in the case of Operator_2 it is specified only for an IEEE 802.11 network.

For each network supported by an operator there is a list of supported PoAs. For each PoA the PoA Information Elements are specified. Figure 19 shows this information representation and tree hierarchy for different networks.

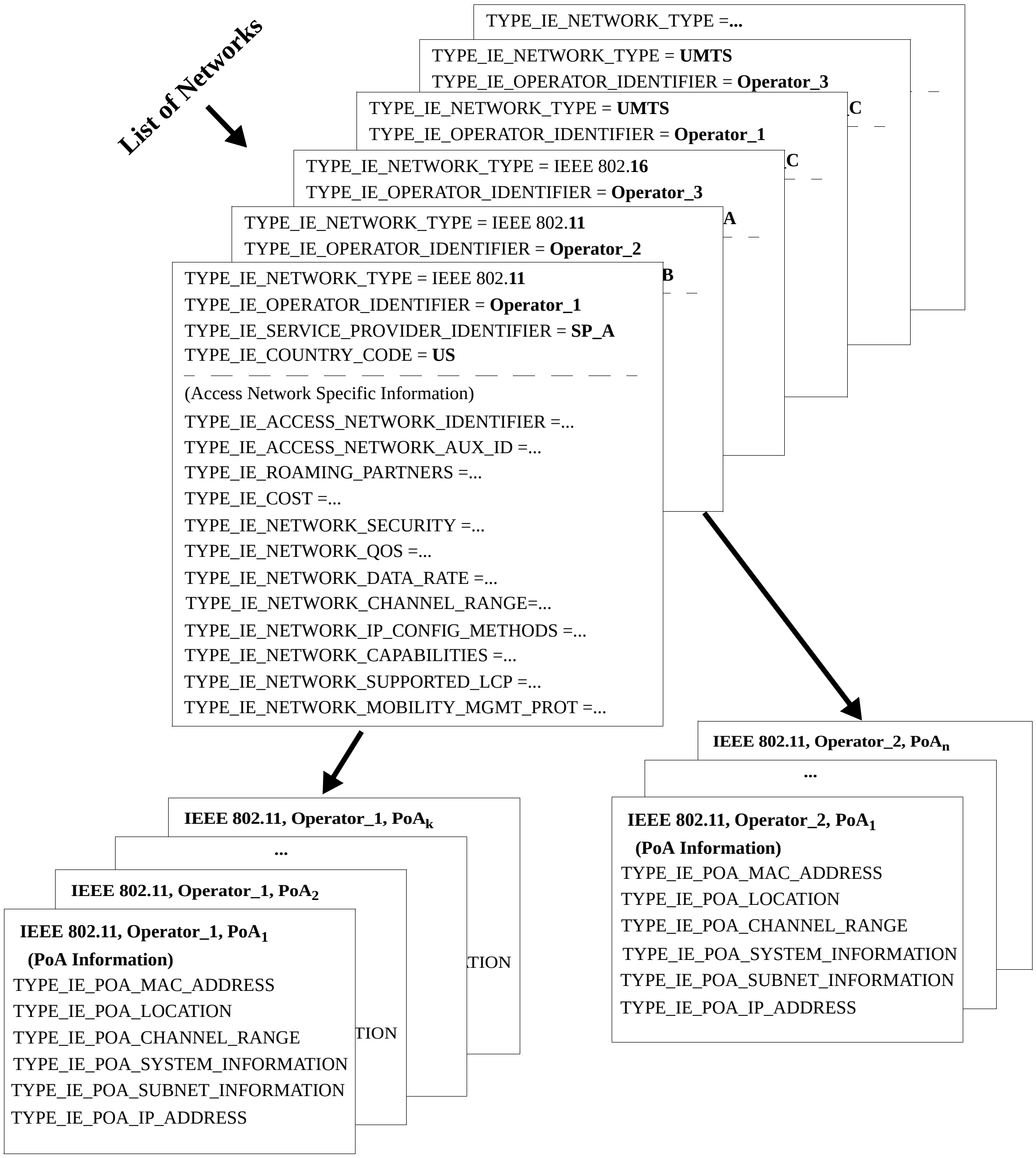


Figure 19—Depicting a list of neighboring networks with information elements.

6.5.5 Definition of information element name space

Each Information Element ID is a 32 bit value. The IEEE 802.21 specific Information Elements are assigned identifiers as per this standard. Please refer to Table A-1 (in Annex A) for more details. Vendors may specify their own IEs using the name space allocated to them. A set of IE name space ranges is also reserved for development and testing. These should not be used in released products. Allocation of additional IE name space and any revisions to this assignment will be handled by future revisions of this standard.

Table 9—Information elements name space

Range	Description	Comments
0x00000000	Reserved	
0x00000001 - 0x1FFFFFFF	IEEE 802.21 IEs	Used for IEEE 802.21 defined IEs. The currently defined IEEE 802.21 IEs are listed in Table A-1 in Annex A.
0x20000000 - 0x2FFFFFFF	Vendor specific IEs	Used for IEs defined by vendors.
0x30000000 - 0x32FFFFFF	Reserved for playpen area.	May be used in development and testing. Should not be used in released products. Avoids collision during development.
0x33000000 – 0xFFFFFFFF	Reserved	For future use

Functional entities shall discard any received IE with an unrecognizable identifier.

6.5.6 Information element representation and query methods

6.5.6.1 Introduction

MIIS defines two methods for representing Information Elements: binary representation and RDF representation (see W3C Recommendation, “Resource Description Framework (RDF) – Concepts and Abstract Syntax” and W3C Recommendation, “RDF/XML Syntax Specification”). MIIS also defines two query methods. For requests using the binary representation, the TLV query method defined in Clause 6.5.5.2 shall be used. For requests using the RDF representation, the SPARQL (see W3C Recommendation, SPARQL Query Language for RDF) query method shall be used. Both methods may be supported by a client or a server.

6.5.6.2 Binary representation and TLV query

In the binary representation method, Information Elements shall be represented and encoded in Type-Length-Value form as follows.

Type (4 octets)	Length (variable octets)	Value (variable octets)
Represents the type or ID of the Information Element	Length of the Value field of the IE	The value of the IE

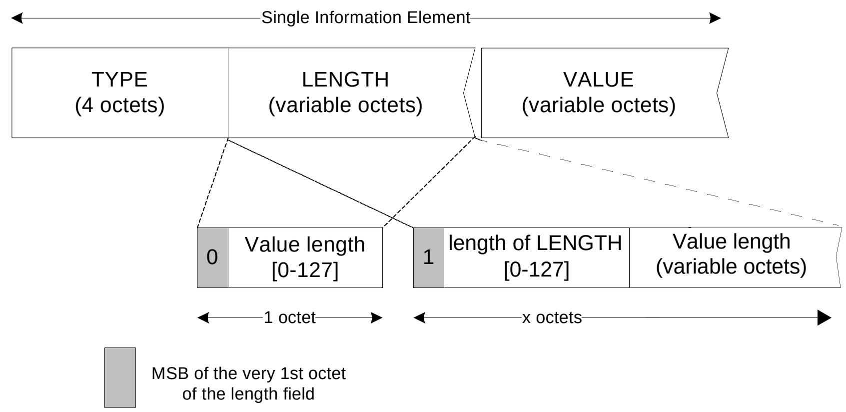


Figure 20 — TLV representation of information elements

The *Length* field shall be interpreted as follows.

Case 1: If the number of octets occupied by the *Value* field is less than 128, the size of the *Length* field is always 1 octet and the MSB of the octet is set to the value ‘0’. The values of the other seven bits of this octet indicate the actual length of the *Value* field.

Case 2: If the number of octets occupied by the *Value* field is exactly 128, the size of the *Length* field is one octet. The MSB of the *Length* octet is set to the value '1' and the other seven bits of this octet are all set to the value ‘0’.

Case 3: If the number of octets occupied by the *Value* field is greater than 128, then the *Length* field is always greater than 1 octet. The MSB of the first octet of the *Length* field is set to the value ‘1’ and the remaining 7 bits of the first octet indicate the number of octets that are appended further. The number represented by the 2nd and subsequent octets of the *Length* field, when added to 128, indicates the total size of the *Value* field, in octets.

6.5.6.2.1 IE containers

In the binary representation method, three Information Element Containers are defined, namely the **List of Neighboring Access Networks Container**, the **Access Network Container**, and the **PoA Container**:

- **List of Neighboring Access Networks Container** - contains a list of heterogeneous neighboring access networks for a given geophysical location.

  A List of Neighboring Access Networks Container shall contain at least one Access Network and may contain one or more Vendor Specific IEs. When more than one Access Network Container is provided in this IE, they should be prioritized in the order of preference from the information server's perspective with first Access Network Container as the top priority and with decreasing priority going down the list. This would enable the receiving entity to utilize this information in the same way as provided in this list for network selection or handover decisions.

Type= *TYPE_IE_CONTAINER_LIST_OF_NETWORKS*	**Length=** *Variable*
Access Network Container #1	
Access Network Container #2 (optional)	

…
Access Network Container #k (optional)
Vendor Specific IE (optional)

- **Access Network Container** - contains all the information depicting an access network.

  When more than one PoA Container is provided in this IE, they should be prioritized in the order of preference from the information server's perspective with first PoA Container as the top priority and with decreasing priority going down the list. This would enable the receiving entity to utilize this information in the same way as provided in this list for network selection or handover decisions

Type=*TYPE_IE_CONTAINER_NETWORK*	**Length**= *Variable*
Network Type IE	
Operator Identifier IE	
Service Provider Identifier IE (optional)	
Access Network Identifier IE (optional)	
Access Network AUX ID IE (optional)	
Roaming Partners IE (optional)	
Cost IE (optional)	
Network Security IE (optional)	
Network QoS IE (optional)	
Network Data Rate IE (optional)	
Network Channel Range IE (optional)	
Network IP Configuration Methods IE (optional)	
Network Capabilities IE (optional)	
List of Supported LCP IE (optional)	
Mobility Management Protocol IE (optional)	
PoA Container #1 (optional)	
PoA Container #2 (optional)	
…	
PoA Container #k (optional)	
Vendor Specific Network IE (optional)	

- **PoA Container** - contains all the information depicting a PoA and may include one or more optional Vendor Specific PoA IEs:

Type=*TYPE_IE_CONTAINER_POA*	**Length**= *Variable*
PoA MAC Address IE	

PoA Location IE
PoA Channel Range IE
PoA System Information IE
PoA Subnet Information IE #1
PoA Subnet Information IE #2 (optional)
...
PoA Subnet Information IE #k (optional)
PoA IP Address IE #1 (optional)
...
PoA IP Address IE #k (optional)
Vendor Specific PoA IE (optional)

TLVs for the component IEs contained in the Access Network Container and PoA Container are defined in Annex B.

6.5.6.2.2 TLV Queries

A TLV query may include the following optional parameters to refine the query.

Querier Location, if provided in the query, can be useful for the Information Server to refine its response. The value field contains either the querier's current location measurement or, when the querier does not have its current location information, an observed MAC address (e.g., from an IEEE 802.11 beacon or some broadcast mechanism for other technologies) which the Information Server will be able to use as a hint to establish an estimate of the client's current location. The querier shall not use both the **MAC Address** and **Location** in the same query. Moreover, the **Neighborhood Radius**, if provided, indicates the radius of the neighborhood, centered at the indicated location, within which all available access networks will be included in the list of neighboring networks. If **Neighborhood Radius** is not present, the Information Server will decide the radius for the search.

If **Querier Location** is not provided, the Information Server shall either get the querier location information through other means or use best estimate of the querier's location to generate the neighboring network information.

Network Type Inclusion, if provided in the query, is used to indicate the neighboring network types the querier wants to include in the response. The querier indicates the network types it wants to include in the query response by setting the corresponding bits to "1". If not provided, the Information Server shall include all available network types in the query response.

Network Inclusion, if provided in the query, is used to indicate the specific access networks the querier wants to include in the query response. If not provided, the Information Server shall include all available access networks in the query response.

Reporting Template, indicates to the information server a template of the list of IEs which shall be included in the information response.

Rules for using **Reporting Template**:

a) If the **Reporting Template** is absent, the entire list of neighboring networks container shall be returned in the response

b) If a container is listed *without* any of its component IEs, the entire container shall be returned in the response. For example, inclusion of *TYPE_IE_CONTAINER_POA* solely returns a list of PoA Containers with all their component IEs.

c) If a container is listed *with* one or more of its component IEs, the container *with only* the listed component IEs shall be returned. For example, inclusion of *TYPE_IE_CONTAINER_NETWORK, TYPE_IE_NETWORK_TYPE and TYPE_IE_OPERATOR_IDENTIFIER* solely returns a list of Network Containers with each containing only Network Type and Operator ID.

d) If a component IE is listed *without* its parent container, the listed component IE shall be returned as an individual IE. For example, inclusion of *TYPE_IE_NETWORK_TYPE and TYPE_IE_COST* solely returns a list of Network Types and a list of Costs. Note, a list of individual IEs out of their context may have very limited usefulness. This is only an example to show the flexible use of Reporting Template.

Rules for generating returned IEs:

Upon receipt of a binary query, the information server will:

a) create the list of neighboring access network information for the given location;

— If a **Network Type Inclusion** is provided in the query, include only the information of the neighboring access networks of the network type(s) indicated in the **Network Type Inclusion**. Otherwise, include information of all available neighboring access networks for the given location.

— If a **Network Inclusion** is provided in the query, include only the information of the neighboring access network(s) indicated in the **Network Inclusion**. Otherwise, include information of all available neighboring access networks for the given location.

b) if no **Reporting Template** is given in the query, send the list of neighboring access network information in a **List of Neighboring Access Networks Container** in an Information Response.

c) if a **Reporting Template** is given in the query, extract the requested IE(s)/Containers from the list of neighboring access network information using the rules described for **Reporting Template** and send them in an Information Response.

6.5.6.3 RDF representation and SPARQL query

The RDF representation of Information Elements shall be represented in XML format. SPARQL shall be used as the query method. The RDF representation and SPARQL query method shall implement the RDF schema as described in Subclause 6.5.7.2.

6.5.7 Information service schema

6.5.7.1 General

A schema is used in the IEEE 802.21 Information Service to define the structure of each information element, as well as the relationship among the information elements. The IEEE 802.21 Information Service schema shall be supported by every MIHF that implements the MIIS to support flexible and efficient information queries.

6.5.7.2 The MIIS RDF schema

The RDF schema definition for MIIS consists of two parts; the basic and the extended schema. An MIIS client or server should be pre-provisioned with the basic schema for ease of implementation of schema-based query. In scenarios where the basic schema is not pre-provisioned, methods such as DHCP may be used to obtain the basic schema.

The MIIS RDF representation method is extensible using the extended schema. The extended schema may be pre-provisioned. The extended schema may be updated, e.g., when a new information element about the network is introduced. The extended schema may be retrieved from the specified URL via the IEEE 802.21 Information Service using the schema query capability without any pre-provision of such extended schema. Alternatively methods such as DHCP may provide the URL of the extended schema. If the extended schema is pre-provisioned, but the updated extended schema defines information differently than in the pre-provisioned extended schema, the implementation shall use the updated extended schema.

The basic schema is defined in Annex D. Annex F is a UML representation of the basic schema. The basic schema contains the schema for information elements defined in Table 8. The extended schema is defined by individual vendors or by network operators and may contain the schema for vendor-specific information elements or network operator specific information.

6.5.8 Information service flow

Figure 21 describes an Information Service flow. The MIIS within an MIHF communicates with the remote MIHF that resides within the access network. MIH_Get_Information from the MN is carried over the appropriate transport (L2 or L3) and is delivered to the remote MIHF. The remote MIHF returns the necessary information to the MN via the appropriate response frame.

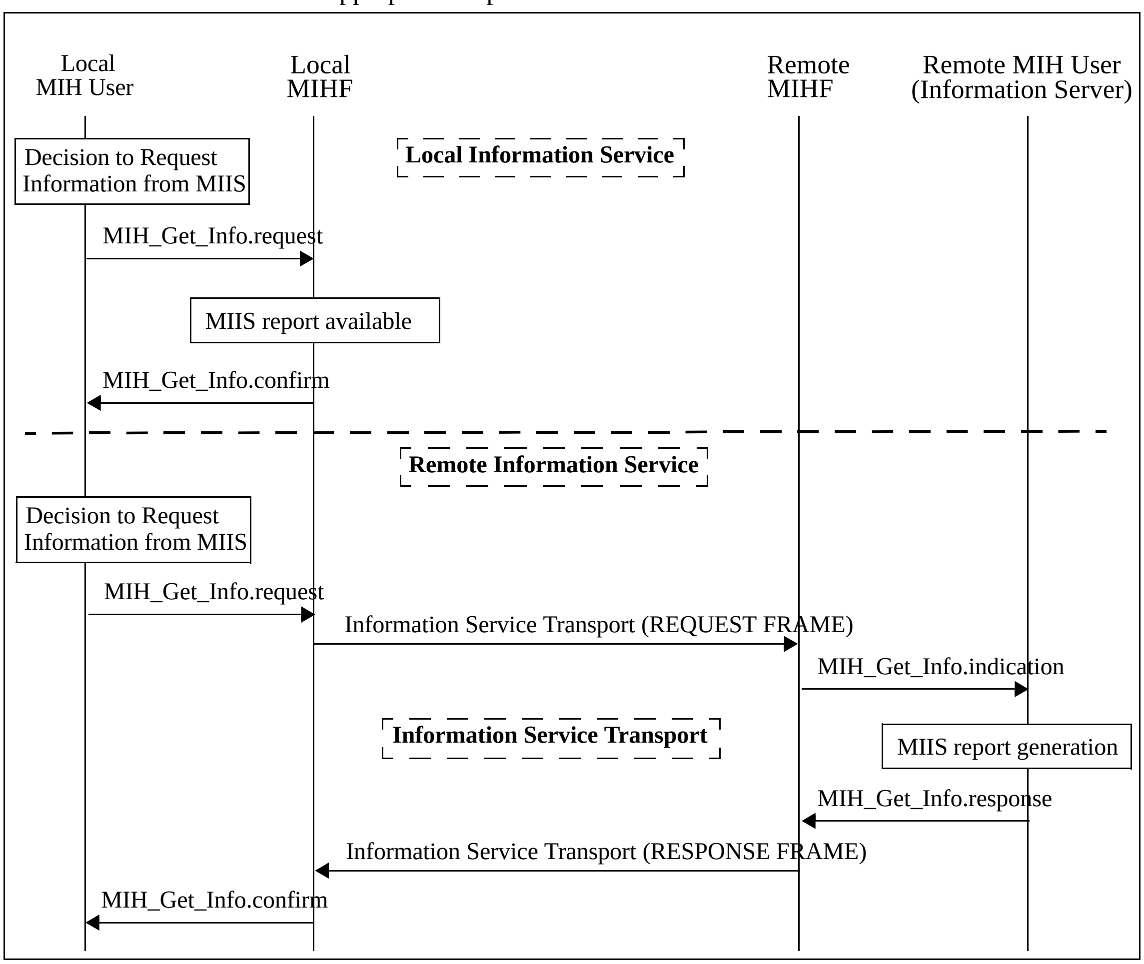


Figure 21 — MIIS information flow

7. Service access points (SAPs) and primitives

7.1 Introduction

The MIH Function uses the following SAPs for interfacing with other entities.

Media dependent SAPs:

1) MIH_LINK_SAP: Abstract media dependent interface of MIHF with the lower layers of the media-specific protocol stacks. The mappings between MIH_LINK_SAP and various media-specific SAPs are described in Annex M.
2) MIH_NET_SAP: Abstract media dependent interface of MIHF which provides transport services over the data plane on the local node, supporting the exchange of MIH information and messages with the remote MIHF.

Media independent SAPs:

1) MIH_SAP: This SAP defines the media independent interface between the MIHF and MIH Users.

7.2 SAPs

7.2.1 General

The SAPs are defined as a set of primitives. Taken together, the primitives define the services. Within the definition of each primitive there is a table of allowable parameters. Each parameter is defined using abstract data types. These types indicate the semantic value of that parameter. The parameters defined within the clause for a particular primitive are produced or consumed by that primitive. Several of the abstract data types are used in multiple primitive definitions. In each abstract data type definition, the various names applied to this type are listed.

Most of the primitives have corresponding MIH protocol messages. In those messages, there are TLV encoded parameters that implement the primitive parameter abstract data types within the protocol. The definition of the full binary encoding for each of these instantiations is in Annex B.

7.2.2 Media dependent SAPs

7.2.2.1 MIH_LINK_SAP

The primitives defined as part of the MIH_LINK_SAP are described in Table 10. Annex K contains their mapping to several specific link technologies.

Table 10—MIH_LINK_SAP primitives

Primitives	Service category	Description	Defined in
Link_Detected	Event	A new link is detected	7.3.1
Link_Up	Event	L2 connectivity is established	7.3.2
Link_Down	Event	L2 connectivity is lost	7.3.3
Link_Going_Down	Event	L2 connectivity loss is imminent	7.3.4
Link_Event_Rollback	Event	Predicted event has not occurred and hence event indication shall be rolled back.	7.3.5
Link_Parameters_Report	Event	Link parameters have crossed specified thresholds	7.3.6
Link_PDU_Transmit_Status	Event	Indicate transmission status of a PDU	7.3.7
Link_Handover_Imminent	Event	L2 handover is imminent	7.3.8
Link_Handover_Complete	Event	L2 handover has been completed	7.3.9
Link_Event_Subscribe	Command	Subscribe for event notifications	7.3.10
Link_Event_Unsubscribe	Command	Unsubscribe from event notifications	7.3.11
Link_Configure_Thresholds	Command	Configure link thresholds for Link events	7.3.12
Link_Capability_Discover	Command	Query and discover the list of supported link layer events and link layer commands	7.3.13
Link_Get_Parameters	Command	Request parameters of medium	7.3.14
Link_Action	Command	Request actions on a link layer connection	7.3.15

7.2.2.2 MIH_NET_SAP

The following primitive is defined for MIH_NET_SAP.

Table 11—MIH_NET_SAP primitive

Primitive	Service category	Description	Defined in
MIH_TP_Data	Network Communication	This primitive is used for transfer of data	7.5.1

7.2.3 Media independent SAPs

7.2.3.1 MIH_SAP

The primitives defined as part of MIH_SAP are described in Table 12.

Table 12—MIH_SAP primitives

Primitives	Service category	Description	Defined in
MIH_Capability_Discover	Service Management	Discover list of Events and Commands supported by MIHF.	7.4.1
MIH_Register	Service Management	Register with a remote MIHF.	7.4.2
MIH_DeRegister	Service Management	Deregister with a remote MIHF.	7.4.3
MIH_Event_Subscribe	Service Management	Subscribe for MIH event notifications	7.4.4
MIH_Event_Unsubscribe	Service Management	Unsubscribe from MIH event notifications	7.4.5
MIH_Link_Up	Event	L2 connection has been established	7.4.6
MIH_Link_Down	Event	L2 connectivity is lost	7.4.7
MIH_Link_Going_Down	Event	L2 connectivity is predicted to go down	7.4.8
MIH_Link_Detected	Event	A new link is detected	7.4.9
MIH_Link_Parameters_Report	Event	Link parameters have crossed specified threshold	7.4.10
MIH_Link_Event_Rollback	Event	Predicted event has not occurred and hence event indication shall be rolled back	7.4.11
MIH_Link_PDU_Transmit_Status	Event	Indicate transmission status of a PDU	7.4.12
MIH_Link_Handover_Imminent	Event	L2 handover is imminent	7.4.13
MIH_Link_Handover_Complete	Event	L2 handover has been completed	7.4.14
MIH_Get_Link_Parameters	Command	Get the status of link	7.4.15
MIH_Link_Configure_Thresholds	Command	Configure link parameter thresholds	7.4.16

Table 12—MIH_SAP primitives

Primitives	Service category	Description	Defined in
MIH_Link_Actions	Command	Control the behavior of a set of links	7.4.17
MIH_Net_HO_Candidate_Query	Command	Initiate handover	7.4.18
MIH_MN_HO_Candidate_Query	Command	Initiate MN query request for candidate network	7.4.19
MIH_N2N_HO_Query_Resources	Command	Prepare for handover and query available resources	7.4.20
MIH_Net_HO_Commit	Command	Network has committed to handover	7.4.21
MIH_MN_HO_Commit	Command	Mobile node has committed to handover	7.4.22
MIH_N2N_HO_Commit	Command	A serving network to inform a target network about a committed handover	7.4.23
MIH_MN_HO_Complete	Command	Initiate MN handover complete notification	7.4.24
MIH_N2N_HO_Complete	Command	Handover has been completed	7.4.25
MIH_Get_Information	Information	Request to get information from repository	7.4.26

MIH command primitives defined in MIH_SAP may indicate its destination as either the local MIHF or a remote peer MIHF. For the remote case, the local MIHF will first process the primitive to create an MIH message and then forward the message to the destination peer MIHF for execution.

7.3 MIH_LINK_SAP primitives

7.3.1 Link_Detected.indication

7.3.1.1 Function

Link_Detected indicates the presence of a new PoA. This may imply that the mobile node is in the coverage area and may listen to a beacon, or that the mobile node may have received a response to a probe. Link_Detected does not guarantee that the mobile node would be able to establish connectivity with the detected link, but just that the mobile node may attempt to gain connectivity. MIH Users and the MIHF itself may evaluate additional properties of the link before attempting to establish a L2 connection with the link. Moreover, Link_Detected is not generated when additional PoAs of the same link are discovered. In case of 802.11, this is generated by MSCGF.

7.3.1.2 Semantics of service primitive

```
Link_Detected.indication  (
                          LinkIdentifier,
                          AccessNetworkIdentifier,
                          SignalStrength,
                          MIHCapability
                          )
```

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.
Access Network Identifier	ACCESS_NETWORK_ID	Identifier to uniquely determine the access network.
Signal Strength	SIGNAL_STRENGTH	The signal strength on the detected link.
MIH Capability	LINK_MIH_CAPABILITY_FLAG	Used to indicate which MIH capabilities are supported on this link.

7.3.1.3 When generated

The Link Detected event is generated on the MN when the first PoA of an access network is detected. This event is not generated when subsequent PoAs of the same access network are discovered during the active connection on that link.

7.3.1.4 Effect on receipt

The MIHF receives this event from the link layer. The MIHF then passes this notification to the MIH User(s) which have subscribed for this notification. The MIH User(s), including the MIHF itself, may discover additional properties of the link before selecting it for establishing connectivity.

7.3.2 Link_Up.indication

7.3.2.1 Function

This notification is delivered when a layer 2 connection is established on the specified link interface. All layer 2 activities in establishing the link connectivity are expected to be completed at this point of time.

7.3.2.2 Semantics of service primitive

```
Link_Up.indication          (
                            LinkIdentifier,
                            OldAccessRouter,
                            NewAccessRouter,
                            IPRenewalFlag,
                            Mobility Management Support
                            )
```

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.
Old Access Router	LINK_ADDRESS	(Optional) Old Access Router link address.
New Access Router	LINK_ADDRESS	(Optional) New Access Router link address.
IP Renewal Flag	IP_RENEWAL_FLAG	(Optional) Indicates whether the MN shall change IP Address in the new PoA.
Mobility Management Support	IP_MOBILITY_MGMT	(Optional) Indicates the type of Mobility Management Protocol supported by the new PoA.

7.3.2.3 When generated

This notification is generated when a layer 2 connection is established for the specified link interface.

7.3.2.4 Effect on receipt

The MIHF passes this link notification to the MIH User(s) which has subscribed for this notification in an MIH_Link_Up event. The MIH User(s) may take different actions on this notification.

7.3.3 Link_Down.indication

7.3.3.1 Function

This notification is delivered when a layer 2 connection is no longer available for sending frames, that is, when the L2 connection with network is terminated and not during PoA to PoA transitions for the same network.

7.3.3.2 Semantics of service primitive

```
Link_Down.indication    (
                        LinkIdentifier,
                        OldAccessRouter,
                        ReasonCode
                        )
```

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.
Old Access Router	LINK_ADDRESS	(Optional) Old Access Router link address.
Reason Code	LINK_DOWN_REASON	Reason why the link went down.

7.3.3.3 When generated

This notification is generated when layer 2 connectivity is lost. Layer 2 connectivity may be lost explicitly in cases where the mobile node may initiate disassociate type procedures. In other cases the mobile node may infer loss of link connectivity due to successive time-outs for acknowledgements of retransmitted packets along with loss of reception of broadcast frames.

7.3.3.4 Effect on receipt

The MIHF passes this link notification to the MIH User(s) which has subscribed for this notification in an MIH_Link_Down event. The MIH User(s) may take different actions on this notification. The handover policy may eliminate this link from list of active links for routing connections and may consider handing over any potential active connections to other more suitable links.

7.3.4 Link_Going_Down.indication

7.3.4.1 Function

This notification is delivered when a Layer 2 connection is expected (predicted) to go down (Link_Down) within a certain time interval. Link_Going_Down event may be the indication to initiate handover procedures.

7.3.4.2 Semantics of service primitive

```
Link_Going_Down.indication      (
                                LinkIdentifier,
                                TimeInterval,
                                LinkGoingDownReason,
                                UniqueEventIdentifier
                                )
```

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.
Time Interval	UNSIGNED_INT(2)	Time Interval (in milliseconds) specifies the time interval at which the link is expected to go down. A non-zero value is specified only if there is a high confidence level in the time interval. A value of ‘0’ is specified if there is low-confidence in the computation of time interval.
Link Going Down Reason	LINK_GOING_DOWN_REASON	The reason why the link is going to be down.
Unique Event Identifier	UNSIGNED_INT(2)	Used to uniquely identify the event. To be used in case of event rollback

7.3.4.3 When generated

A Link_Going_Down event implies that a Link_Down is imminent within a certain time interval. If Link_Down or Link_Event_Rollback is NOT received within specified time interval then actions due to previous Link_Going_Down shall be ignored.

A 100% confidence would indicate certainty that the link is going down within the specified time interval. For example a BS that may have decided to shut down for administrative reasons may send down a Link_Going_Down trigger with 100% confidence level. Predictions based on changes in link parameters such as signal strength values would have a lower confidence level. Different links would use implementation specific methods to predict future loss in link connectivity.

In case of 802.11 networks, this notification is generated when the established 802.11 network connection is expected to go down within the specified time interval by the 802.11 MSGCF. The network may be expected to go down because of an event whose timing is well understood, such as an explicit disconnection event observed on the MLME_SAP. It may also be expected as the result of a predictive algorithm that monitors link quality. The details of such a predictive algorithm used are beyond the scope of this standard. This event is not generated when the 802.11 STA transitions from one AP to another in the same network.

7.3.4.4 Effect on receipt

The MIHF receives this event from the link layer. The MIHF then passes this notification to the MIH User(s) which have subscribed for this notification. MIH User(s) may take different actions on this notification. MIH Users may prepare to initiate handovers based on the confidence level reported as part of the event.

7.3.5 Link_Event_Rollback.indication

7.3.5.1 Function

Link_Event_Rollback is used in conjunction with Link_Going_Down. If the link is no longer expected to go down in the specified time interval, then a Link_Event_Rollback message is sent to the Event destination. The destination should disregard or rollback the changes associated with the event identifier in such cases.

7.3.5.2 Semantics of service primitive

```
Link_Event_Rollback.indication     (
                                   LinkIdentifier,
                                   UniqueEventIdentifier
                                   )
```

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event
Unique Event Identifier	UNSIGNED_INT (2)	Used to identify the event which needs to be rolled back

7.3.5.3 When generated

This notification is generated when a predictive event expected to occur within a certain time interval is no longer expected to occur before the expiry of the indicated time interval.

7.3.5.4 Effect on receipt

The MIHF receives this event from the link layer. The MIHF then passes this notification to the MIH User(s) which have subscribed for this notification. The MIH User(s) may reevaluate the link under consideration by retrieving other link quality related parameters. Any handover procedures that may have already been initiated may need to be rescinded.

7.3.6 Link_Parameters_Report.indication

7.3.6.1 Function

Link_Parameters_Report indicates changes in link parameters that have crossed specified threshold levels. Link_Parameters_Report may also be generated at specified intervals for various parameters.

In case of 802.11 network, this event is generated when higher protocol layers wish to monitor the performance parameters for a network. These higher layers may be on the network side (for network initiated handovers) and MIHF on the local MN can transfer these parameters. For local MN initiated handovers, the local SME and MSCGF would monitor link layer properties and the MIHF would normally be interested only in the Link_Going_Down.indication.

7.3.6.2 Semantics of service primitive

Link_Parameters_Report.indication(
LinkIdentifier,
LinkParametersReportList
)

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event
Link Parameters Report List	LIST(LINK_PARAM_ REPORT)	A list of Link Parameter Report.

The primitive to set parameter thresholds that could trigger this event is specified in 7.3.12.

7.3.6.3 When generated

For each specified parameter, this notification is generated either at a predefined regular interval determined by a user configurable timer or when it crosses a configured threshold.

In case of 802.11 network, this event is

7.3.6.4 Effect on receipt

The MIHF receives this event from the link layer. The MIHF then passes this notification to the MIH User(s) which have subscribed for this notification. The MIH User(s) may take different actions on this notification. If parameters related to link quality cross a certain threshold then that link may need to be evaluated for handing over current connections. The MIHF may collectively evaluate different parameters and give appropriate indications to higher layers regarding suitability of different links.

7.3.7 Link_PDU_Transmit_Status.indication

7.3.7.1 Function

Link_PDU_Transmit_Status indicates the transmission status of a higher layer PDU by the link layer. A success status indicates that the higher layer PDU has been successfully delivered from the link layer in the local node to the link layer in the peer node. A higher layer intermediate buffer management entity could use this indication to flush the delivered PDU from its buffer. A failure status indicates that the higher layer

PDU identified in the indication was not delivered successfully from the link layer in the local node to the link layer in the peer node. During a handover, if such a failure indication is received from the link connection with the source network, the higher layer intermediate buffer management entity could attempt to retransmit the failed PDU once a connection to the target network is established.

A Packet Identifier is expected to be passed alongside when each higher layer PDU is sent from the higher layer to the link for transmission. The Packet Identifier is defined in this standard as a container structure whose syntax and semantics will be decided by the upper layer (i.e., the MIH User which subscribes to this event). The MIHF and link layer just pass and return the Packet Identifier and do not need to understand its syntax and semantics.

To avoid receiving excessive amount of link PDU transmission status indications, an MIH User, for example, may choose to subscribe to this event only after it receives a Link_Handover_Imminent.indication or when it is about to invokes an MIH_Link_Actions.request to perform a handover, and to unsubscribe from the event once it receives indication that the handover is completed.

7.3.7.2 Semantics of service primitive

```
Link_PDU_Transmit_Status.indication      (
                                          LinkIdentifier,
                                          PacketIdentifier,
                                          TransmissionStatus
                                          )
```

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event
Packet Identifier	UNSIGNED_INT(2)	Identifier for higher layer PDU on which this notification is generated.
Transmission Status	BOOLEAN	Status of the transmitted packet. True: Success False: Failure

7.3.7.3 When generated

A success notification is generated when a higher layer PDU is successfully transmitted over the link. A failure notification is generated when a higher layer PDU was not transmitted successfully.

7.3.7.4 Effect on receipt

The MIHF receives this event from the link layer. The MIHF then passes this notification to the MIH User(s) which have subscribed for this notification. The MIH User(s) may take different actions on this notification. A higher layer intermediate buffer management entity in MIH could use the success indication to flush higher layer packets stored in any intermediate buffers and a failure indication to retransmit higher layer packets stored in any intermediate buffers, especially if there are changes in the access network during handovers.

7.3.8 Link_Handover_Imminent.indication

7.3.8.1 Function

Link_Handover_Imminent is generated when a native link layer handover or switch decision has been made and its execution is imminent (as opposed to Link_Going_Down which only indicates that a link is losing connectivity due to a change in a certain link condition such as signal strength, but does not guarantee that an autonomous link switch-over has been decided by the link layer). It contains information about the new point of attachment of the mobile node (the LinkIdentifier parameter contains information about the new PoA). Link_Handover_Imminent may be used by MIH Users to initiate handover specific adaptation thereby reducing packet loss due to link layer handovers. This is a link layer event that exists for intra-technology handovers defined in many media types.

7.3.8.2 Semantics of service primitive

```
Link_Handover_Imminent.indication       (
                                        LinkIdentifier,
                                        MacAccessRouter
                                        )
```

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event
Mac Access Router	MAC_ADDRESS	(Optional) MAC Address of new Access Router

7.3.8.3 When generated

Depending on whether it is the Mobile node or the Network, it is generated when a native link layer handover or switch decision has been made and its execution is imminent.

7.3.8.4 Effect on receipt

The MIHF receives this event from the link layer. The MIHF then passes this notification to the MIH User(s) which have subscribed for this notification. The MIH User(s) may take necessary actions to minimize the effect of the pending native link layer handover or switch on user data transfer. This event may also be used as an indication to start buffering packets.

7.3.9 Link_Handover_Complete.indication

7.3.9.1 Function

Link_Handover_Complete event is generated whenever a native link layer handover/switch has just been completed (as opposed to Link_Up which only indicates that a link has been brought up for L2 connectivity, but does not indicate that a native link handover/switch-over has just been completed by the link layer). Notifying the upper layer of this event may improve transport, session and application layer responsiveness to autonomous link changes. They may better adapt their data flows by resuming flows upon receiving this indication. The upper layer may also use this event to check whether its IP configuration needs to be updated. This is a link layer event that exists for intra-technology handovers defined in many media types. This event is applicable for the mobile node only and is valid only for intra-technology handovers.

7.3.9.2 Semantics of service primitive

Link_Handover_Complete.indication (
LinkIdentifier,
MacAccessRouter
)

Parameters:

Name	Type	Description
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event
Mac Access Router	MAC_ADDRESS	(Optional) MAC Address of new Access Router

7.3.9.3 When generated

This is generated whenever a L2 link layer handover or switch has just been completed.

7.3.9.4 Effect on receipt

The MIHF receives this event from the link layer. The MIHF then passes this notification to the MIH User(s) which have subscribed for this notification. Upon reception of this event, an upper layer may stop any handover adaptation that it has engaged to cope with the just completed native link layer handover/switch and resume normal data transfer. This event may also be used as an indication that a re-verification of the IP parameter should be considered.

7.3.10 Link_Event_Subscribe

7.3.10.1 Link_Event_Subscribe.request

7.3.10.1.1 Function

This primitive is used by MIHF (the subscriber) to subscribe an interest in one or more events from a specific link layer technology. The response indicates which of the requested events were successfully subscribed to. Events that were not successfully subscribed to will not be delivered to the subscriber.

7.3.10.1.2 Semantics of service primitive

Link_Event_Subscribe.request (
RequestedLinkEventList
)

Parameter:

Name	Type	Description
Requested Link Event List	LINK_EVENT_LIST	List of link layer events that the subscriber would like to receive indications for.

7.3.10.1.3 When generated

This primitive is generated by a subscriber such as the MIHF that is seeking to receive event indications from different link layer technologies.

7.3.10.1.4 Effect on receipt

The recipient responds immediately with Link_Event_Subscribe.confirm primitive.

7.3.10.2 Link_Event_Subscribe.confirm

7.3.10.2.1 Function

This primitive returns the result of the subscription request.

7.3.10.2.2 Semantics of service primitive

Link_Event_Subscribe.confirm (
ResponseLinkEventList,
Status
)

Parameters:

Name	Type	Description
Response Link Event List	LINK_EVENT_LIST	List of successfully subscribed link events. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation. Code 3 (Authorization Failure) is not applicable.

7.3.10.2.3 When generated

This primitive is generated in response to a Link_Event_Subscribe.request primitive.

7.3.10.2.4 Effect on receipt

The recipient may examine the ResponseLinkEventList and learn about the subscription status of different events. If Status does not indicate “Success”, the recipient perform appropriate error handling.

7.3.11 Link_Event_Unsubscribe

7.3.11.1 Link_Event_Unsubscribe.request

7.3.11.1.1 Function

This primitive is used by the MIHF (the subscriber) to unsubscribe from a set of previously subscribed link layer events.

7.3.11.1.2 Semantics of service primitive

Link_Event_Unsubscribe.request (
RequestedLinkEventList
)

Parameter:

Name	Type	Description
Requested Link Event List	LINK_EVENT_LIST	List of link layer events for which indications need to be unsubscribed from the Event Source

7.3.11.1.3 When generated

This primitive is generated by a subscriber such as the MIHF that is seeking to unsubscribe from an already subscribed set of events.

7.3.11.1.4 Effect on receipt

The recipient responds immediately with Link_Event_Unsubscribe.confirm primitive.

7.3.11.2 Link_Event_Unsubscribe.confirm

7.3.11.2.1 Function

This primitive returns the result of the request to unsubscribe from receiving link layer event notifications.

7.3.11.2.2 Semantics of service primitive

Link_Event_Unsubscribe.confirm (
ResponseLinkEventList,
Status
)

Parameters:

Name	Type	Description
Response Link Event List	LINK_EVENT_LIST	List of successfully unsubscribed link events. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation. Code 3 (Authorization Failure) is not applicable.

7.3.11.2.3 When generated

This primitive is generated in response to a Link_Event_Unsubscribe.request primitive.

7.3.11.2.4 Effect on receipt

The recipient may examine the ResponseLinkEventList and learn about the unsubscription status of different events. If Status does not indicate "Success", the recipient shall perform appropriate error handling.

7.3.12 Link_Configure_Thresholds

7.3.12.1 Link_Configure_Thresholds.request

7.3.12.1.1 Function

This primitive is used by the MIHF to configure thresholds and/or specify the time interval between periodic reports for the Link_Parameters_Report indication.

7.3.12.1.2 Semantics of service primitive

Link_Configure_Thresholds.request (
LinkConfigureParameterList
)

Parameter:

Name	Type	Description
Link Configure Parameter List	LIST(LINK_CONFIG_PARAM)	A list of Link Configure Parameter.

7.3.12.1.3 When generated

This primitive is generated by an MIHF that may need to set threshold values for different link parameters.

7.3.12.1.4 Effect on receipt

The recipient responds immediately with Link_Configure_Thresholds.confirm primitive.

7.3.12.2 Link_Configure_Thresholds.confirm

7.3.12.2.1 Function

This primitive is sent in response to the Link_Configure_Thresholds.request primitive. This primitive specifies the status of threshold configuration operation.

7.3.12.2.2 Semantics of service primitive

Link_Configure_Thresholds.confirm (
LinkConfigureStatusList,
Status
)

Parameters:

Name	Type	Description
Link Configure Status List	LIST(LINK_CONFIG_STATUS)	A list of Link Configure Status. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation. Code 3 (Authorization Failure) is not applicable.

7.3.12.2.3 When generated

This primitive is generated in response to the Link_Configure_Thresholds.request operation

7.3.12.2.4 Effect on receipt

The recipient prepares to receive Link_Parameters_Report indications on successful execution of this primitive. However, if Status does not indicate “Success”, the recipient shall perform appropriate error handling.

7.3.13 Link_Capability_Discover

7.3.13.1 Link_Capability_Discover.request

7.3.13.1.1 Function

This primitive is used by the MIHF to query and discover the list of supported link layer events and link layer commands.

7.3.13.1.2 Semantics of service primitive

No primitive parameters exist for this primitive.

```
Link_Capability_Discover.request   (
                                   )
```

7.3.13.1.3 When generated

This primitive is generated by the MIHF when it needs to receive link layer event notifications and learn about which link layer commands the lower layer can support.

7.3.13.1.4 Effect on receipt

The recipient responds immediately with Link_Capability_Discover.confirm primitive.

7.3.13.2 Link_Capability_Discover.confirm

7.3.13.2.1 Function

This primitive returns the result of the query to discover link layer capability.

7.3.13.2.2 Semantics of service primitive

```
Link_Capability_Discover.confirm  (
```

SupportedLinkEventList,
SupportedLinkCommandList,
Status
)

Parameters:

Name	Type	Description
Supported Link Event List	LINK_EVENT_LIST	List of link layer events supported by the link layer. Note, this parameter is not included if Status does not indicate "Success".
Supported Link Command List	LINK_COMMAND_LIST	List of link layer commands supported by the link layer. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation. Code 3 (Authorization Failure) is not applicable.

7.3.13.2.3 When generated

This primitive is generated in response to a Link_Capability_Discover.request primitive.

7.3.13.2.4 Effect on receipt

The recipient may examine the returned event and command list and learn about link layer capability. However, if Status does not indicate "Success", the recipient shall perform appropriate error handling.

7.3.14 Link_Get_Parameters

7.3.14.1 Link_Get_Parameters.request

7.3.14.1.1 Function

This primitive is used by the MIHF to obtain the current value of a set of link parameters of a specific link.

7.3.14.1.2 Semantics of service primitive

Link_Get_Parameters.request (
LinkParametersRequest,
LinkStatesRequest,
LinkDescriptorsRequest
)

Parameter:

Name	Type	Description
Link Parameters Request	LIST(LINK_PARAM_TYPE)	A list of link parameters for which status is requested.
Link States Request	LINK_STATES_REQ	The link status to be requested.
Link Descriptors Request	LINK_DESCRIPTORS_REQ	The link descriptors to be requested

7.3.14.1.3 When generated

This primitive is generated by the MIHF to obtain the current value of a set of link parameters from a link.

7.3.14.1.4 Effect on receipt

The recipient link shall respond with Link_Get_Parameters.confirm primitive.

7.3.14.2 Link_Get_Parameters.confirm

7.3.14.2.1 Function

This primitive is sent in response to the Link_Get_Parameters.request primitive. This primitive provides current value of the requested link parameters. (Note, how the value is measured or calculated by the link is not specified by this standard).

7.3.14.2.2 Semantics of service primitive

```
Link_Get_Parameters.confirm     (
                                LinkParametersStatusList,
                                LinkStatesResponse,
                                LinkDescriptorsResponse,
                                Status
                                )
```

Parameters:

Name	Type	Description
Link Parameters Status List	LIST(LINK_PARAM)	A list of measurable link parameters and their current values. Note, this parameter is not included if Status does not indicate “Success”.
Link States Response	LINK_STATES_RSP	The current link state information. Note, this parameter is not included if Status does not indicate “Success”.
Link Descriptors Response	LINK_DESCRIPTORS_RSP	The descriptors of a link. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation. Code 3 (Authorization Failure) is not applicable.

7.3.14.2.3 When generated

This primitive is generated in response to the Link_Get_Parameters.request operation.

7.3.14.2.4 Effect on receipt

The recipient may pass the link parameter values received to the MIH Users. However, if Status does not indicate “Success”, the recipient shall perform appropriate error handling.

7.3.15 Link_Action

7.3.15.1 Link_Action.request

7.3.15.1.1 Function

This primitive is used by the MIHF to request an action on a link layer connection to enable optimal handling of link layer resources for the purpose of handovers. The link layer connection can be ordered, e.g. to shut down, to remain active, to perform a scan, or to come up active and remain in stand-by mode. The command execution delay time can also be specified for cases where the link layer technology under consideration supports the action.

7.3.15.1.2 Semantics of Service Primitive

```
Link_Action.request     (
                         PoALinkAddress,
                         LinkAction,
                         ExecutionDelay
                        )
```

Parameters:

Name	Type	Description
PoA Link Address	LINK_ADDRESS	(Optional) The PoA link address is included when the DATA_FORWARDING_REQUEST action is requested.
Link Action	LINK_ACTION	Specifies the suggested action
Execution Delay	UNSIGNED_INT(2)	Time (in ms) to elapse before the action needs to be taken. A value of 0 indicates that the action shall be taken immediately. Time elapsed shall be calculated from the instance the request arrives until the time when the execution of the action is carried out.

7.3.15.1.3 When generated

The MIHF generates this primitive upon request from the MIH User to perform an action on a pre-defined link layer connection.

7.3.15.1.4 Effect on receipt

Upon receipt of this primitive, the link layer technology supporting the current link layer connections performs the action specified by the Link Action parameter in accordance with the procedures specified by the relevant standards organization and at the time specified by the Execution Delay parameter.

7.3.15.2 Link_Action.confirm

7.3.15.2.1 Function

This primitive is used by link layer technologies to provide an indication of the result of the action executed on the current link layer connection.

7.3.15.2.2 Semantics of Service Primitive

```
Link_Action.confirm       (
                          ScanResponseSet,
                          LinkActionResult,
                          Status
                          )
```

Parameters:

Name	Type	Description
Scan Response Sets	LIST(LINK_SCAN_RSP)	(Optional) A list of discovered links and related information. Note, this parameter is not included if Status does not indicate “Success”.
Link Action Result	LINK_AC_RESULT_CODE	Specifies whether the link action was successful. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of the operation. Note: Code 3 (Authorization Failure) is not applicable.

7.3.15.2.3 When generated

The link layer technology generates this primitive to communicate the result of the action executed on the link layer connection.

7.3.15.2.4 Effect on receipt

Upon receipt of this primitive, the MIHF determines the relevant MIH command that needs to be used to provide an indication or confirmation to the MIH User of the actions performed on the current link layer connection. If a Scan action was issued by the associated Link_Action.request, the optional ScanResponseSet field shall be included in the Link_Action.confirm response.

7.4 MIH_SAP primitives

The primitives defined as part of MIH_SAP are described in the following subclauses.

7.4.1 MIH_Capability_Discover

7.4.1.1 MIH_Capability_Discover.request

7.4.1.1.1 Function

This primitive is used by an MIH User to discover the capabilities of the local MIHF or a remote MIHF. When invoking this primitive to discover the capabilities of a remote MIHF, the MIH User can optionally piggyback the capability information of its local MIHF so that the two MIHFs can mutually discover each other's capabilities with a single invocation of this primitive.

7.4.1.1.2 Semantics of service primitive

MIH_Capability_Discover.request (
DestinationIdentifier,
LinkMACs,
SupportedMihEventList,
SupportedMihCommandList,
SupportedIsQueryTypeList,
SupportedTransportList,
MBBHandoverSupport
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies the local MIHF or a remote MIHF which will be the destination of this request.
Link MACs	LIST(NETWORK_TYPE_ANDMAC)	(Optional) A list of network type and MAC address pair.
Supported MIH Event List	MIH_EVENT_LIST	(Optional) List of supported events on the local MIHF.
Supported MIH Command List	MIH_COMMAND_LIST	(Optional) List of supported commands on the local MIHF.
Supported IS Query Type List	MIH_IS_QUERY_TYPE_LIST	(Optional) List of supported MIIS query types on the local MIHF.
Supported Transport List	MIH_TRANSPORT_LIST	(Optional) List of supported transport types on the local MIHF.
MBB Handover Support	LIST(MBB_HANDOVER_SUPPORT)	(Optional) This is used to indicate if a make before break handover is supported. It is assumed that a break before make handover is supported by default.

7.4.1.1.3 When generated

This primitive is generated by an MIH User to discover the capabilities of the local MIHF or a remote MIHF. In the case of remote discovery, this primitive may contain the SupportedMihEventList, SupportedMihCommandList, SupportedIsQueryTypeList, and SupportedTransportList parameters of the local MIHF to enable mutual discovery of each other's capabilities.

7.4.1.1.4 Effect on receipt

If the destination of the request is the local MIHF itself, the local MIHF shall respond with MIH_Capability_Discover.confirm. If the destination of the request is a remote MIHF, the local MIHF shall generate and send a corresponding MIH_Capability_Discover request message to the remote MIHF if it does not have the capability information of the remote MIHF.

7.4.1.2 MIH_Capability_Discover.indication

7.4.1.2.1 Function

This primitive is used by an MIHF to notify an MIH User on the receipt of an MIH_Capability_Discover request message from a peer MIHF.

7.4.1.2.2 Semantics of Service primitive

MIH_Capability_Discover.indication (
SourceIdentifier,
LinkMACs,
SupportedMihEventList,
SupportedMihCommandList,
SupportedIsQueryTypeList,
SupportedTransportList,
MBBHandoverSupport
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.
Link MACs	LIST(NETWORK_TYPE_AND MAC)	(Optional) A list of network type and MAC address pair.
Supported MIH Event List	MIH_EVENT_LIST	(Optional) List of supported events on MIHF.
Supported MIH Command List	MIH_COMMAND_LIST	(Optional) List of supported commands on MIHF.
Supported IS Query Type List	MIH_IS_QUERY_TYPE_LIST	(Optional) List of supported MIIS query types.
Supported Transport List	MIH_TRANSPORT_LIST	(Optional) List of supported transport types.
MBB Handover Support	LIST(MBB_HANDOVER_SUP PORT)	(Optional) This is used to indicate if a make before break handover is supported. It is assumed that a break before make handover is supported by default.

7.4.1.2.3 When generated

This primitive is used by an MIHF to notify an MIH User when an MIH_Capability_Discover request message is received. This primitive is optional since the MIHF may also immediately return an MIH_Capability_Discover response message without generating this primitive to the MIH User.

7.4.1.2.4 Effect on receipt

The MIH User shall respond with an MIH_Capability_Discover.response primitive when an indication is received.

7.4.1.3 MIH_Capability_Discover.response

This primitive is used by an MIH User to convey the supported MIH capabilities to the MIH User which invoked the MIH_Capability_Discover request.

7.4.1.3.1 Semantics of Service primitive

MIH_Capability_Discover.response(
DestinationIdentifier,
LinkMACs,
SupportedMihEventList,
SupportedMihCommandList,
SupportedIsQueryTypeList,
SupportedTransportList,
MBBHandoverSupport,
Status
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies the remote MIHF which will be the destination of this response.
Link MACs	LIST(NETWORK_TYPE_AND MAC)	(Optional) A list of network type and MAC address pair.
Supported MIH Event List	MIH_EVENT_LIST	(Optional) List of supported events on MIHF.
Supported MIH Command List	MIH_COMMAND_LIST	(Optional) List of supported commands on MIHF.
Supported IS Query Type List	MIH_IS_QUERY_TYPE_LIST	(Optional) List of supported MIIS query types.
Supported Transport List	MIH_TRANSPORT_LIST	(Optional) List of supported transport types.
MBB Handover Support	LIST(MBB_HANDOVER_SUPPORT)	(Optional) This is used to indicate if a make before break handover is supported. It is assumed that a break before make handover is supported by default.
Status	STATUS	Status of operation.

7.4.1.3.2 When generated

This primitive is generated by an MIH User as a response to a received MIH_Capability_Discover.indication primitive. This primitive is optional.

7.4.1.3.3 Effect on receipt

Upon receiving this primitive, the MIHF generates and sends the corresponding MIH_Capability_Discover response message to the destination MIHF.

7.4.1.4 MIH_Capability_Discover.confirm

7.4.1.4.1 Function

This primitive is used by the MIHF to convey the supported MIH capabilities about Event Service, Command Service, and Information Service to the MIH User which invoked the MIH_Capability_Discover.request.

7.4.1.4.2 Semantics of service primitive

```
MIH_Capability_Discover.confirm (
                                SourceIdentifier,
                                LinkMACs,
                                SupportedMihEventList,
                                SupportedMihCommandList,
                                SupportedIsQueryTypeList,
                                SupportedTransportList,
                                MBBHandoverSupport,
                                Status
                                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link MACs	LIST(NETWORK_TYPE_AND MAC)	(Optional) A list of network type and MAC address pair.
Supported MIH Event List	MIH_EVENT_LIST	(Optional) List of supported events on MIHF.
Supported MIH Command List	MIH_COMMAND_LIST	(Optional) List of supported commands on MIHF.
Supported IS Query Type List	MIH_IS_QUERY_TYPE_LIST	(Optional) List of supported MIIS query types.
Supported Transport List	MIH_TRANSPORT_LIST	(Optional) List of supported transport types.
MBB Handover Support	LIST(MBB_HANDOVER_SUPPORT)	(Optional) This is used to indicate if a make before break handover is supported. It is assumed that a break before make handover is supported by default.
Status	STATUS	Status of operation.

7.4.1.4.3 When generated

This primitive is invoked by a local MIHF to convey the results of a previous MIH_Capability_Discover.request primitive from an MIH User.

7.4.1.4.4 Effect on receipt

Upon reception of this primitive the receiving entity becomes aware of the supported MIH capabilities. However, if Status does not indicate “Success”, the recipient shall ignore any other returned values and, instead, shall perform appropriate error handling.

7.4.2 MIH_Register

7.4.2.1 MIH_Register.request

7.4.2.1.1 Function

This primitive is used by an MIH User to register the local MIHF with remote MIHF.

7.4.2.1.2 Semantics of service primitive

MIH_Register.request (
DestinationIdentifier,
LinkIdentifierList,
RequestCode
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this request.
Link Identifier List	LIST(LINK_ID)	List of local link identifiers.
Request Code	REG_REQUEST_CODE	Registration request code. Depending on the request code, the MIH User can choose to either register or re-register with the remote MIHF.

7.4.2.1.3 When generated

This primitive is invoked by the MIH User when it needs to register the local MIHF with a remote MIHF.

7.4.2.1.4 Effect on receipt

On receipt, the local MIHF shall send an MIH_Register request message to the destination MIHF.

7.4.2.2 MIH_Register.indication

7.4.2.2.1 Function

This primitive is used by an MIHF to notify an MIH User that an MIH_Register request message has been received.

7.4.2.2.2 Semantics of service primitive

MIH_Register.indication (
SourceIdentifier,
LinkIdentifierList,

RequestCode
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.
Link Identifier List	LIST(LINK_ID)	List of link identifiers of the remote MIHF.
Request Code	REG_REQUEST_CODE	Registration request code. Depending on the request code, the MIH User can choose to either register or re-register with the remote MIHF.

7.4.2.2.3 When generated

This primitive is generated by the remote MIHF when an MIH_Register request message is received.

7.4.2.2.4 Effect on receipt

The remote MIH User will perform necessary actions to process the registration request and respond with an MIH_Register.response.

7.4.2.3 MIH_Register.response

7.4.2.3.1 Function

This primitive is used by an MIH User to send the processing status of a received registration request.

7.4.2.3.2 Semantics of service primitive

MIH_Register.response (
Destination Identifier,
ValidTimeInterval,
Status
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this response.
Valid Time Interval	UNSIGNED_INT(4)	Time interval in seconds during which the registration is valid. Parameter applicable only when the status parameter indicates a successful operation. A value of 0 indicates an infinite validity period. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.2.3.3 When generated

This primitive is invoked by the MIH User to report back the result after completing the processing of a registration request.

7.4.2.3.4 Effect on receipt

Upon receipt, the local MIHF shall send an MIH_Register response message to the destination MIHF.

7.4.2.4 MIH_Register.confirm

7.4.2.4.1 Function

This primitive is used by the local MIHF to convey the result of a registration request to an MIH User.

7.4.2.4.2 Semantics of service primitive

```
MIH_Register.confirm      (
                          SourceIdentifier,
                          ValidTimeInterval,
                          Status
                          )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.
Valid Time Interval	UNSIGNED_INT(4)	Time interval in seconds during which the registration is valid. Parameter applicable only when the status parameter indicates a successful operation. A value of 0 indicates an infinite validity period. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.2.4.3 When generated

This primitive is used by an MIHF to notify an MIH User the result of an MIH registration request.

7.4.2.4.4 Effect on receipt

Upon receipt, the MIH User can determine the result of the registration request.

7.4.3 MIH_DeRegister

7.4.3.1 MIH_DeRegister.request

7.4.3.1.1 Function

This primitive is used by an MIH User to deregister the local MIHF with peer MIHF.

7.4.3.1.2 Semantics of service primitive

MIH_DeRegister.request (
DestinationIdentifier
)

Parameter:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this request.

7.4.3.1.3 When generated

This primitive is invoked by the MIH User when it needs to terminate an existing MIH registration with a remote MIHF.

7.4.3.1.4 Effect on receipt

Upon receipt, the local MIHF shall generate and send an MIH_DeRegister request message to the destination MIHF.

7.4.3.2 MIH_DeRegister.indication

7.4.3.2.1 Function

This primitive is used by an MIHF to notify an MIH User that an MIH_DeRegister request message has been received.

7.4.3.2.2 Semantics of service primitive

MIH_DeRegister.indication(
SourceIdentifier
)

Parameter:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.

7.4.3.2.3 When generated

This primitive is generated by an MIHF when an MIH_DeRegister request message is received.

7.4.3.2.4 Effect on receipt

The MIH User will perform necessary actions to process the deregistration request and respond with an MIH_DeRegister.response.

7.4.3.3 MIH_DeRegister.response

7.4.3.3.1 Function

This primitive is invoked by a remote MIH User to respond with the processing status of a received deregistration request.

7.4.3.3.2 Semantics of service primitive

```
MIH_DeRegister.response (
                         DestinationIdentifier,
                         Status
                         )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this response.
Status	STATUS	Status of operation. Note: Code 2 (Reject) shall not be used.

7.4.3.3.3 When generated

This primitive is invoked by the MIH User to report back the result after completing the processing of a deregistration request from a remote MIH User.

7.4.3.3.4 Effect on receipt

Upon receipt, the local MIHF shall send an MIH_DeRegister response message to the destination MIHF.

7.4.3.4 MIH_DeRegister.confirm

7.4.3.4.1 Function

This primitive is used by the local MIHF to convey the result of a deregistration request to the local MIH User.

7.4.3.4.2 Semantics of service primitive

```
MIH_DeRegister.confirm  (
                         SourceIdentifier,
                         Status
                         )
```

Parameters:

Name	Type	Description

Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.
Status	STATUS	Status of operation. Code 2 (Rejected) shall not be used.

7.4.3.4.3 When generated

This primitive is used by an MIHF to notify the local MIH User the status of MIH deregistration request.

7.4.3.4.4 Effect on receipt

Upon receipt, the MIH User can determine the status of the deregistration request.

7.4.4 MIH_Event_Subscribe

7.4.4.1 MIH_Event_Subscribe.request

7.4.4.1.1 Function

This primitive is used by an MIH User (the subscriber) to subscribe an interest in one or more MIH event types from the local or a remote MIHF. Optionally, the subscriber may indicate a list of specific configuration information applicable for various events being subscribed. If configured, the event must be triggered only when all the criteria set in the parameters are met.

7.4.4.1.2 Semantics of service primitive

```
MIH_Event_Subscribe.request      (
                                 DestinationIdentifier,
                                 LinkIdentifier,
                                 RequestedMihEventList,
                                 EventConfigurationInfoList,
                                 )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies the local MIHF or a remote MIHF which will be the destination of this request.
Link Identifier	LINK_TUPLE_ID	Identifier of the link for event subscription. For local event subscription, PoA link address need not be present if the link type lacks such a value.
Requested MIH Event List	MIH_EVENT_LIST	List of MIH events that the endpoint would like to receive indications for, from the Event Source.
Event Configuration Info List	LIST(EVENT_CONFIGURATION_INFO)	List of additional configuration information for event subscription.

7.4.4.1.3 When generated

This primitive is invoked by an MIH User when it wants to receive indications on a set of specific MIH events from the local MIHF or a remote MIHF.

7.4.4.1.4 Effect on receipt

If the destination of the request is the local MIHF itself, the local MIHF shall respond immediately with an MIH_Event_Subscribe.confirm primitive. If the destination of the request is a remote MIHF, the local MIHF shall generate and send an MIH_Event_Subscribe request message to the remote MIHF.

7.4.4.2 MIH_Event_Subscribe.confirm

7.4.4.2.1 Function

This primitive returns the result of an MIH event subscription request.

7.4.4.2.2 Semantics of service primitive

```
MIH_Event_Subscribe.confirm     (
                                SourceIdentifier
                                LinkIdentifier,
                                ResponseMihEventList,
                                Status
                                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link for event subscription.
Response MIH Event List	MIH_EVENT_LIST	List of successfully subscribed MIH events. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.4.2.3 When generated

This primitive is generated by the local MIHF at the completion of processing an MIH_Event_Subscribe.request primitive from a local MIH User or in response to the receiving of an MIH_Event_Subscribe response message from a peer MIHF.

7.4.4.2.4 Effect on receipt

The recipient MIH User may examine the returned event list and learn about the subscription status of different events. However, if Status does not indicate “Success”, the recipient shall perform appropriate error handling.

7.4.5 MIH_Event_Unsubscribe

7.4.5.1 MIH_Event_Unsubscribe.request

7.4.5.1.1 Function

This primitive is used by an MIH User (the subscriber) to unsubscribe from a set of previous subscribed MIH events.

7.4.5.1.2 Semantics of service primitive

```
MIH_Event_Unsubscribe.request    (
                                 DestinationIdentifier,
                                 LinkIdentifier,
                                 RequestedMihEventList
                                 )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies the local MIHF or a remote MIHF which will be the destination of this request.
Link Identifier	LINK_TUPLE_ID	Identifier of the link for event unsubscription. For local event unsubscription, PoA MAC Address in the Link Identifier need not be present if the link type lacks such a value.
Requested MIH Event List	MIH_EVENT_LIST	List of MIH events for which indications need to be unsubscribed from the Event Source.

7.4.5.1.3 When generated

This primitive is invoked by an MIH User (subscriber) that is seeking to unsubscribe from an already subscribed set of events from the local MIHF or a remote MIHF.

7.4.5.1.4 Effect on receipt

If the destination of the request is the local MIHF itself, the local MIHF shall respond immediately with MIH_Event_Unsubscribe.confirm primitive. If the destination of the request is a remote MIHF, the local MIHF shall generate and send an MIH_Event_Unsubscribe request message to the remote MIHF.

7.4.5.2 MIH_Event_Unsubscribe.confirm

7.4.5.2.1 Function

This primitive returns the result of an MIH event unsubscription request.

7.4.5.2.2 Semantics of service primitive

```
MIH_Event_Unsubscribe.confirm    (
                                 SourceIdentifier,
                                 LinkIdentifier,
                                 ResponseMihEventList,
```

Status
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link for event unsubscription.
Response MIH Event List	MIH_EVENT_LIST	List of successfully unsubscribed link events. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.5.2.3 When generated

This primitive is generated by the local MIHF at the completion of processing an MIH_Event_Unsubscribe.request primitive from a local MIH User or in response to the receiving of an MIH_Event_Unsubscribe response message from a peer MIHF.

7.4.5.2.4 Effect on receipt

The recipient MIH User may examine the returned event list and learn about the unsubscription status of different events. However, if Status does not indicate “Success”, the recipient shall perform appropriate error handling.

7.4.6 MIH_Link_Up

7.4.6.1 MIH_Link_Up.indication

7.4.6.1.1 Function

The MIH_Link_Up.indication is sent to local MIHF users to notify them of a local event, or is the result of the receipt of an MIH_Link_Up indication message to indicate to the remote MIHF users, who have subscribed to this remote event.

7.4.6.1.2 Semantics of the service primitive

MIH_Link_Up.indication (
SourceIdentifier,
LinkIdentifier,
OldAccessRouter,
NewAccessRouter,
IPRenewalFlag,
Mobility Management Support
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.
Old Access Router	LINK_ADDRESS	(Optional) Link address of old Access Router.
New Access Router	LINK_ADDRESS	(Optional) Link address of new Access Router.
IP Renewal Flag	IP_RENEWAL_FLAG	(Optional) Indicates whether the MN shall change IP Address in the new PoA.
Mobility Management Support	IP_MOBILITY_MGMT	(Optional) Indicates the type of Mobility Management Protocol supported by the new PoA.

7.4.6.1.3 When generated

The MIH_Link_Up.indication is sent to local MIHF users to notify them of a local event (i.e., Link_Up.indication), or is the result of the receipt of an MIH_Link_Up indication message to indicate to the remote MIHF users, who have subscribed to this remote event, that a remote link up event occurred.

7.4.6.1.4 Effect on receipt

MIH User dependant.

7.4.7 MIH_Link_Down

7.4.7.1 MIH_Link_Down.indication

7.4.7.1.1 Function

The MIH_Link_Down.indication is sent to local MIHF users to notify them of a local event, or is the result of the receipt of an MIH_Link_Down indication message to indicate to the remote MIHF users, who have subscribed to this remote event.

7.4.7.1.2 Semantics of the service primitive

```
MIH_Link_Down.indication (
                    SourceIdentifier,
                    LinkIdentifier,
                    OldAccessRouter,
                    ReasonCode
                    )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.

Name	Type	Description
Old Access Router	LINK_ADDRESS	(Optional) Link address of old Access Router.
Reason Code	LINK_DOWN_RE ASON	Reason why the link went down.

7.4.7.1.3 When generated

The MIH_Link_Down.indication is sent to local MIHF users to notify them of a local event (i.e., Link_Down.indication), or is the result of the receipt of an MIH_Link_Down indication message to indicate to the remote MIHF users, who have subscribed to this remote event, that a remote link_down event occurred.

7.4.7.1.4 Effect on receipt

MIH User dependant.

7.4.8 MIH_Link_Going_Down

7.4.8.1 MIH_Link_Going_Down.indication

7.4.8.1.1 Function

The MIH_Link_Going_Down.indication is sent to local MIHF users to notify them of a local event, or is the result of the receipt of an MIH_Link_Going_Down indication message to indicate to the remote MIHF users, who have subscribed to this remote event.

7.4.8.1.2 Semantics of the service primitive

MIH_Link_Going_Down.indication (
SourceIdentifier,
LinkIdentifier,
TimeInterval,
LinkGoingDownReason,
UniqueEventIdentifier
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.
Time Interval	UNSIGNED_INT(2)	Time Interval (in milliseconds) specifies the time interval at which the link is expected to go down. A non-zero value is specified only if there is a high confidence level in the time interval. A value of ‘0’ is specified if there is low-confidence in the computation of time interval.

Name	Type	Description
Link Going Down Reason	LINK_GOING_DOWN_REASON	The reason why the link is going down.
Unique Event Identifier	UNSIGNED_INT(2)	Used to uniquely identify the event. To be used in case of event rollback

7.4.8.1.3 When generated

The MIH_Link_Going_Down.indication is sent to local MIHF users to notify them of a local event (i.e., Link_Going_Down.indication), or is the result of the receipt of an MIH_Link_Going_Down indication message to indicate to the remote MIHF users, who have subscribed to this remote event, that a remote link_going_down event occurred.

7.4.8.1.4 Effect on receipt

MIH User dependant.

7.4.9 MIH_Link_Detected

7.4.9.1 MIH_Link_Detected.indication

7.4.9.1.1 Function

The MIH_Link_Detected.indication is sent to local MIHF users to notify them of a local event, or is the result of the receipt of an MIH_Link_Detected indication message to indicate to the remote MIHF users, who have subscribed to this remote event.

7.4.9.1.2 Semantics of the service primitive

```
MIH_Link_Detected.indication (
                    SourceIdentifier,
                    LinkDetectedInfoList
                    )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Detected Info List	LIST(LINK_DETECTED_INFO)	List of link detection information.

7.4.9.1.3 When generated

The MIH_Link_Detected.indication is sent to local MIHF users to notify them of a local event (i.e., Link_Detected.indication), or is the result of the receipt of an MIH_Link_Detected indication message to indicate to the remote MIHF users, who have subscribed to this remote event, that a remote link_detected event occurred.

7.4.9.1.4 Effect on receipt

MIH User dependant.

7.4.10 MIH_Link_Parameters_Report

7.4.10.1 MIH_Link_Parameters_Report.indication

7.4.10.1.1 Function

MIH_Link_Parameters_Report indication is sent by the local MIHF to a local MIH User to report the status of a set of parameters of a local or remote link.

The event may be local or remote.

7.4.10.1.2 Semantics of service primitive

```
MIH_Link_Parameters_Report.indication    (
                                          SourceIdentifier,
                                          LinkIdentifier,
                                          LinkParameterReportList
                                          )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event
Link Parameter Report List	LIST(LINK_PARAM_REPORT)	A list of Link Parameter Reports.

7.4.10.1.3 When generated

This notification is generated by the local MIHF either:

- at a predefined regular interval determined by a user configurable timer;
- when a specified parameter of a currently active local interface crosses a configured threshold. In such a case, the local MIHF most likely will first receive a Link_Parameters_Report.indication from the local link layer; or
- when an MIH_Link_Parameters_Report indication message is received from a remote MIHF.

7.4.10.1.4 Effect on receipt

Upper layer entities may take different actions upon receipt of this indication.

7.4.11 MIH_Link_Event_Rollback

7.4.11.1 MIH_Link_Event_Rollback.indication

7.4.11.1.1 Function

The MIH_Link_Event_Rollback.indication is sent to local MIHF users to notify them of a local event.

7.4.11.1.2 Semantics of the service primitive

```
MIH_Link_Event_Rollback.indication (
                SourceIdentifier,
                LinkIdentifier,
                UniqueEventIdentifier
                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event.
Unique Event Identifier	UNSIGNED_INT(2)	Used to identify the event which needs to be rolled back.

7.4.11.1.3 When generated

The MIH_Link_Event_Rollback.indication is sent to local MIHF users to notify them of a local event (i.e., Link_Event_Rollback.indication), or is the result of the receipt of an MIH_Link_Event_Rollback indication message to indicate to the remote MIHF users, who have subscribed to this remote event, that a remote link_event_rollback event occurred.

7.4.11.1.4 Effect on receipt

MIH User dependant.

7.4.12 MIH_Link_PDU_Transmit_Status

7.4.12.1 MIH_Link_PDU_Transmit_Status.indication

7.4.12.1.1 Function

The MIH_Link_PDU_Transmit_Status.indication is sent to local MIHF users to notify them of a local event.

7.4.12.1.2 Semantics of the service primitive

```
MIH_Link_PDU_Transmit_Status.indication(
                SourceIdentifier,
                LinkIdentifier,
                PacketIdentifier,
                TransmissionStatus
                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the local MIHF where this event occurred.
Link Identifier	LINK_TUPLE_ID	Identifier of the link associated with the event
Packet Identifier	UNSIGNED_INT(2)	Identifier for higher layer PDU on which this notification is generated.
Transmission Status	Boolean	Status of the transmitted packet. True: Success False: Failure

7.4.12.1.3 When generated

The MIH_Link_PDU_Transmit_Status.indication is sent to local MIHF users to notify them of a local event (i.e., Link_PDU_Transmit_Status.indication).

7.4.12.1.4 Effect on receipt

MIH User dependant.

7.4.13 MIH_Link_Handover_Imminent

7.4.13.1 MIH_Link_Handover_Imminent.indication

MIH_Link_Handover_Imminent is issued by the MIHF to report that a link switch is about to occur. The event notified may be either an intra-technology or inter-technology link switch. In the event of an intra-technology link switch, this indication directly corresponds to the link layer event Link_Handover_Imminent.indication defined in Subclause 7.3.8.

7.4.13.1.1 Function

This primitive is issued by the MIHF to report the imminent occurrence of either an intra-technology or inter-technology link handover to MIH Users. This MIH event may be either local or remote.

7.4.13.1.2 Semantics of service primitive

```
MIH_Link_Handover_Imminent.indication  (
                                        SourceIdentifier,
                                        Old Link Identifier,
                                        New Link Identifier,
                                        HandoverType,
                                        MacOldAccessRouter,
                                        MacNewAccessRouter
                                        )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Old Link Identifier	LINK_TUPLE_ID	Identifier of the old link.
New Link Identifier	LINK_TUPLE_ID	Identifier of the new link.
Handover Type	HANDOVER_TYPE	Indicates whether intra-technology or inter-technology handover:
Mac Old Access Router	LINK_ADDRESS	(Optional) Link address of old Access Router.
Mac New Access Router	LINK_ADDRESS	(Optional) Link address of new Access Router.

7.4.13.1.3 When generated

This notification is generated by the MIHF when a link layer inter- or intra-technology handover is about to occur. In the case of an intra-technology handover, the event could be triggered by the reception of a Link_Handover_Imminent.indication from a link.

7.4.13.1.4 Effect on receipt

Upper layer entities may take different actions upon notification.

7.4.14 MIH_Link_Handover_Complete

7.4.14.1 MIH_Link_Handover_Complete.indication

MIH_Link_Handover_Complete is issued by the MIHF to report the successful completion of a link handover to MIH Users. The event notified may be either an intra-technology or inter-technology link handover. The link handover may be a result of either the execution of an MIH_MN_HO_Commit or MIH_Net_HO_Commit command previously issued by an MIH User or by a Link_Handover_Complete indication from the link layer.

7.4.14.1.1 Function

This primitive is issued by the MIHF to report the completion of a link handover to MIH Users. This event may notify MIH Users of a successful intra-technology or inter-technology handover. This MIH event may be local or remote.

7.4.14.1.2 Semantics of service primitive

```
MIH_Link_Handover_Complete.indication  (
                                        SourceIdentifier,
                                        OldLinkIdentifier,
                                        NewLinkIdentifier,
                                        MacOldAccessRouter,
                                        MacNewAccessRouter,
                                        LinkHandoverStatus
                                        )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Old Link Identifier	LINK_TUPLE_ID	Identifier of the old link
New Link Identifier	LINK_TUPLE_ID	Identifier of the new link
Mac Old Access Router	LINK_ADDRESS	(Optional) Link address of old Access Router
Mac New Access Router	LINK_ADDRESS	(Optional) Link address of new Access Router
Link Handover Status	STATUS	Status of the link handover.

7.4.14.1.3 When generated

This notification is generated by the MIHF when a link layer inter-technology or intra-technology handover is completed. In the case of an intra-technology handover, the event could be triggered by the reception of a Link_Handover_Complete.indication from a link. In the case of an inter-technology handover, the event could be triggered by the successful execution of an MIH_MN_HO_Commit or MIH_Net_HO_Commit primitive issued earlier by an MIH User.

7.4.14.1.4 Effect on receipt

Upper layer entities may take different actions on this notification. An MIH User may make use of this notification to configure other layers (IP, Mobile IP) for various upper layer handovers that are needed. Transport layers (e.g., TCP) may also make use of this primitive to fine tune their flow control and flow congestion mechanisms.

7.4.15 MIH_Get_Link_Parameters

An MIH_Get_Link_Parameters command is issued by upper layer entities to discover and monitor the status of the currently connected and potentially available links. The destination of an MIH_Get_Link_Parameters command may be local or remote. For example, an MIH_Get_Link_Parameters request issued by a local upper layer may help the policy function that resides out of the MIH to make optimal handover decisions for different applications when multiple links are available in a mobile node. However, a remotely initiated MIH_Get_Link_Parameters request from the network side may enable the network to collect the status information on multiple links in a mobile node through the currently connected link.

7.4.15.1 MIH_Get_Link_Parameters.request

7.4.15.1.1 Function

This primitive is invoked by an MIH User to discover the status of the currently connected and potentially available links.

7.4.15.1.2 Semantics of the service primitive

MIH_Get_Link_Parameters.request (
    DestinationIdentifier,
    DeviceStatesRequest,
    LinkIdentifierList,

GetStatusRequestSet
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies the local MIHF or a remote MIHF which will be the destination of this request.
Device States Request	DEVICE_STATES_REQ	(Optional) List of device states being requested.
Link Identifier List	LIST(LINK_ID)	List of link identifiers for which status is requested. If the list is empty, return the status of all available links.
Get Status Request Set	LINK_STATUS_REQ	Indicate which link status(es) is being requested.

7.4.15.1.3 When generated

This primitive is invoked by an MIH User when it wants to request the status information of a set of local or remote links.

7.4.15.1.4 Effect of receipt

If the destination of the request is the local MIHF itself, the local MIHF shall get the requested information on the status of the specified local links and respond with an MIH_Get_Link_Parameters.confirm. If the destination of the request is a remote MIHF, the local MIHF shall generate and send an MIH_Get_Link_Parameters request message to the remote MIHF.

7.4.15.2 MIH_Get_Link_Parameters.confirm

7.4.15.2.1 Function

This primitive is issued by an MIHF to report the requested status of a set of specific local or remote links in response to an MIH_Get_Link_Parameters request from a local or remote MIH User.

7.4.15.2.2 Semantics of the service primitive

MIH_Get_Link_Parameters.confirm (
SourceIdentifier,
DeviceStatesResponseList,
GetStatusResponseList,
Status
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.

Device States Response List	LIST(DEVICE_STATES_RSP)	(Optional) List of device states responses. Note, this parameter is not included if Status does not indicate "Success".
Get Status Response List	LIST(LINK_STATUS_RSP)	List of Status Responses. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.15.2.3 When generated

This primitive returns the results of an MIH_Get_Link_Parameters request to the requesting MIH User.

7.4.15.2.4 Effect of receipt

Upon receipt of the link status information, the MIH User makes appropriate decisions and takes suitable actions. However, if Status does not indicate "Success", the recipient shall perform appropriate error handling.

7.4.16 MIH_Link_Configure_Thresholds

The MIH_Link_Configure_Thresholds may be issued by an upper layer entity to configure parameter report thresholds of a lower layer. When the MIHF receives MIH_Link_Configure_Thresholds, it issues the Link_Configure_Thresholds commands to the corresponding link.

7.4.16.1 MIH_Link_Configure_Thresholds.request

7.4.16.1.1 Function

This primitive is issued by an MIH User to configure thresholds of a lower layer link.

7.4.16.1.2 Semantics of the service primitive

```
MIH_Link_Configure_Thresholds.request (
                              DestinationIdentifier,
                              LinkIdentifier,
                              ConfigureRequestList
                              )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies the local MIHF or a remote MIHF which will be the destination of this request.
Link Identifier	LINK_TUPLE_ID	Identifier of the link to be configured.
Configure Request List	LIST(LINK_CONFIG_PARAM)	List of Configuration Requests.

7.4.16.1.3 .When generated

This primitive is invoked by an MIH User when it attempts to configure thresholds of a local or remote lower layer link.

7.4.16.1.4 Effect of receipt

If the destination of the request is the local MIHF itself, the local MIHF shall issue a Link_Configure_Thresholds request to the lower layer link to set the thresholds for the link according to the specified configuration parameters.

If the destination of the request is a remote MIHF, the local MIHF shall generate and send an MIH_Link_Configure_Thresholds request message to the remote MIHF. Upon the receipt of the message, the remote MIHF shall then issue a Link_Configure_Thresholds request to the lower layer link to set the thresholds for the link according to the specified configuration parameters.

7.4.16.2 MIH_Link_Configure_Thresholds.confirm

7.4.16.2.1 Function

This primitive is issued by an MIHF to report the result of an MIH_Link_Configure_Thresholds request.

7.4.16.2.2 Semantics of the service primitive

```
MIH_Link_Configure_Thresholds.confirm (
                                SourceIdentifier,
                                LinkIdentifier,
                                ConfigureResponseList,
                                Status
                                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF.
Link Identifier	LINK_TUPLE_ID	Identifier of the link configured.
Configure Response List	LIST(LINK_CONFIG_STATUS)	List of the configuration status for each requested link. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.16.2.3 When generated

This primitive returns the result of an MIH_Link_Configure_Thresholds request to the requesting MIH User.

7.4.16.2.4 Effect of receipt

Upon receipt of the result, the MIH User makes appropriate evaluations and takes any suitable actions. However, if Status does not indicate “Success”, the recipient shall perform appropriate error handling.

7.4.17 MIH_Link_Actions

7.4.17.1 MIH_Link_Actions.request

7.4.17.1.1 Function

This primitive is used by an MIH User to control the behavior of a set of local or remote lower layer links.

7.4.17.1.2 Semantics of Service Primitive

The parameters of the service primitive are as follows:

```
MIH_Link_Actions.request (
                          Destination Identifier,
                          LinkActionsList
                          )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies the local MIHF or a remote MIHF which will be destination of this request.
Link Actions List	LIST(LINK_ACTION_REQ)	Specifies the suggested actions.

7.4.17.1.3 When generated

This primitive is invoked by an MIH User when it attempts to control the behavior of a set of local or remote lower layer links.

7.4.17.1.4 Effect on receipt

If the destination of the request is the local MIHF itself, the local MIHF shall issue Link_Action request(s) to the specified lower layer link(s).

If the destination of the request is a remote MIHF, the local MIHF shall generate and send an MIH_Link_Actions request message to the remote MIHF. Upon the receipt of the message, the remote MIHF shall then issue Link_Action request(s) to the specified lower layer link(s).

7.4.17.2 MIH_Link_Actions.confirm

7.4.17.2.1 Function

This primitive is issued by an MIHF to report the result of an MIH_Link_Actions request.

7.4.17.2.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MIH_Link_Actions.confirm (
SourceIdentifier,
LinkActionsResponse,
Status
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be either the local MIHF or a remote MIHF
Link Actions Response	LIST(LINK_ACTION_RSP)	Contain the result of the request link actions. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.17.2.3 When generated

This primitive returns the result of an MIH_Link_Actions.request to the requesting MIH User.

7.4.17.2.4 Effect on receipt

Upon receipt of the result, the MIH User makes appropriate evaluations and takes any suitable actions. However, if Status does not indicate "Success", the recipient shall perform appropriate error handling.

7.4.18 MIH_Net_HO_Candidate_Query

7.4.18.1 General

For network initiated handovers, the network controller provides a list of candidate network choices to the MN (via MIH_Net_HO_Candidate_Query request message). The MN indicates resources required on each of these candidate networks in the MIH_Net_HO_Candidate_Query response message. The network controller may then query each of the candidate networks for available resources (using MIH_N2N_HO_Query_Resources primitive). Once the target network has been selected, the network controller may send an MIH_Net_HO_Commit message. An example of this operation is illustrated in H.2 and H.3.

7.4.18.2 MIH_Net_HO_Candidate_Query.request

7.4.18.2.1 Function

The primitive is invoked by an MIH User on a network node to communicate to a peer MIH User about its intent of handover initiation.

7.4.18.2.2 Semantics of service primitive

MIH_Net_HO_Candidate_Query.request (
DestinationIdentifier,
SuggestedNewLinkList,

QueryResourceReportFlag
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this request.
Suggested New Link List	LIST(LINK_POA_LIST)	A list of PoAs for each link, suggesting the new access networks to which handover initiation should be considered. The access networks towards the top of the list are more preferable than those towards the bottom of the list.
Query Resource Report Flag	BOOLEAN	Flag to specify if resources need to be reported by MN: TRUE - Required to report resource list FALSE - Not required to report resource list.

7.4.18.2.3 When generated

This primitive is invoked by an MIH User to communicate with a remote MIH User about its intent of handover initiation. Serving PoS requests MN to provide information about resources required to initiate a handover by setting the QueryResourceReportFlag parameter.

7.4.18.2.4 Effect on receipt

Upon receipt of this primitive by the local MIHF, it shall generate and send an MIH_Net_HO_Candidate_Query request message to the remote MIHF identified by the Destination Identifier. On the remote MIHF, the MIHF shall forward the request as an indication to the MIH User.

7.4.18.3 MIH_Net_HO_Candidate_Query.indication

7.4.18.3.1 Function

This primitive is generated by an MIHF to indicate to an MIH User that an MIH_Net_HO_Candidate_Query request message was received from a remote MIHF.

7.4.18.3.2 Semantics of service primitive

MIH_Net_HO_Candidate_Query.indication (
SourceIdentifier,
SuggestedNewLinkList,
QueryResourceReportFlag
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.

Suggested New Link List	LIST(LINK_POA_LIST)	A list of PoAs for each link, suggesting the new access networks to which handover initiation should be considered. The access networks towards the top of the list are more preferable than those towards the bottom of the list.
Query Resource Report Flag	BOOLEAN	Flag to specify if resources need to be reported by MN: TRUE - Required to report resource list FALSE - Not required to report resource list.

7.4.18.3.3 When generated

This primitive is generated by an MIHF on receiving an MIH_Net_HO_Candidate_Query request message from a peer MIHF.

7.4.18.3.4 Effect on receipt

An MIH User receiving this indication should invoke an MIH_Net_HO_Candidate_Query.response primitive towards the remote MIHF indicated by the Source Identifier in the request message.

7.4.18.4 MIH_Net_HO_Candidate_Query.response

7.4.18.4.1 Function

This primitive is used by the MIHF on an MN to respond to an MIH_Net_HO_Candidate_Query request message from a remote MIHF in the network.

7.4.18.4.2 Semantics of service primitive

```
MIH_Net_HO_Candidate_Query.response (
                DestinationIdentifier,
                SourceLinkIdentifier,
                PreferredLinkList,
                AvailableResourceSet,
                HandoverStatus,
                IPConfigurationMethods,
                DHCPServerAddress,
                FAAddress,
                AccessRouterAddress,
                Status
                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this response.
Source Link Identifier	LINK_TUPLE_ID	This identifies the current link.

Preferred Link List	LIST(LINK_POA_LIST)	A list of PoAs for each link, suggesting the new access networks and PoAs to which handover initiation should be considered. This may be different than the networks that were suggested in the handover request. The list is sorted from most preferred first to least preferred last. Note, this parameter is not included if Status does not indicate "Success".
Available Resource Set	LIST(QOS_LIST)	A list of available resources for each link. Note, this parameter is not included if Status does not indicate "Success".
Handover Status	HANDOVER_STATUS	Lists the acceptance status (permit/decline) of the handover request. Note, this parameter is not included if Status does not indicate "Success".
IP Configuration Methods	IP_CONFIG_METHODS	(Optional) IP configuration methods applicable for all PreferredCandidateLinkList. Note, this parameter is not included if Status does not indicate "Success".
DHCP Server Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate DHCP Server. It is only included when dynamic address configuration is supported. Note, this parameter is not included if Status does not indicate "Success".
FA Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate Foreign Agent. It is only included Mobile IPv4 is supported. Note, this parameter is not included if Status does not indicate "Success".
Access Router Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate Access Router. It is only included IPv6 Stateless configuration is supported. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.18.4.3 When generated

The remote MIH User invokes this primitive in response to an MIH_Net_HO_Candidate_Query.indication from its MIHF.

7.4.18.4.4 Effect on receipt

The MIHF shall send an MIH_Net_HO_Candidate_Query response message to the peer MIHF as indicated in the Destination Identifier.

7.4.18.5 MIH_Net_HO_Candidate_Query.confirm

7.4.18.5.1 Function

This primitive is generated by the MIHF to confirm that an MIH_Net_HO_Candidate_Query response message was received from a peer MIHF.

7.4.18.5.2 Semantics of service primitive

MIH_Net_HO_Candidate_Query.confirm (
SourceIdentifier,
SourceLinkIdentifier,
PreferredLinkList,
AvailableResourceSet,
HandoverStatus,
IPConfigurationMethods,
DHCPServerAddress,
FAAddress,
AccessRouterAddress,
Status
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the mobile node that sent the MIH_Net_HO_Candidate_Query response message.
Source Link Identifier	LINK_TUPLE_ID	This identifies the current link.
Preferred Link List	LIST(LINK_POA_LIST)	A list of PoAs, suggesting the new access networks to which handover initiation should be considered. This may be different than the networks that were suggested in the handover request. The list is sorted from most preferred first to least preferred last. Note, this parameter is not included if Status does not indicate “Success”.
Available Resource Set	LIST(QOS_LIST)	A list of available resources for each link. Note, this parameter is not included if Status does not indicate “Success”.
Handover Status	HANDOVER_STATUS	Indicates the MN accepted or declined the handover request. Note, this parameter is not included if Status does not indicate “Success”.
IP Configuration Methods	IP_CONFIG_METHODS	(Optional) IP configuration methods applicable for all PreferredCandidateLinkList. Note, this parameter is not included if Status does not indicate “Success”.
DHCP Server Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate DHCP Server. It is only included when dynamic address configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.
FA Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate Foreign Agent. It is only included Mobile IPv4 is supported. Note, this parameter is not included if Status does not indicate “Success”.

Access Router Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate Access Router. It is only included IPv6 Stateless configuration is supported. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.18.5.3 When generated

This primitive is generated by the MIHF on receiving an MIH_Net_HO_Candidate_Query response message from a peer MIHF.

7.4.18.5.4 Effect on receipt

On receiving the primitive the entity which originally initiated the handover request may decide to carry out the handover or abort it based on the primitive. However, if Status does not indicate "Success", the recipient shall ignore any other returned values and, instead, shall perform appropriate error handling.

7.4.19 MIH_MN_HO_Candidate_Query

7.4.19.1 MIH_MN_HO_Candidate_Query.request

7.4.19.1.1 Function

This primitive is used by MIH Users on a mobile node to inform MIHF to query candidates for possible handover initiation. The request may include queries on QoS resources and/or whether IP address configuration method of the ongoing data sessions can be supported in the candidate network. This primitive also includes the current IP configuration server address (e.g., DHCP server, FA IP address, AR IP address) when the current IP configuration method is included.

7.4.19.1.2 Semantics of service primitive

```
MIH_MN_HO_Candidate_Query.request (
                    DestinationIdentifier,
                    SourceLinkIdentifier,
                    CandidateLinkList,
                    QueryResourceList,
                    IPConfigurationMethods,
                    DHCPServerAddress,
                    FAAddress,
                    AccessRouterAddress
                    )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this request.
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link for handover.

Candidate Link List	LIST(LINK_POA_LIST)	A list of PoAs, identifying candidate networks to which handover needs to be initiated. The list is sorted from most preferred first to least preferred last.
Query Resource List	QOS_LIST	List of resources to be queried at candidate networks.
IP Configuration Methods	IP_CONFIG_METHODS	(Optional) Current IP configuration methods.
DHCP Server Address	IP_ADDRESS	(Optional) IP address of current DHCP Server. It is only included when mobile node is using dynamic address configuration
FA Address	IP_ADDRESS	(Optional) IP address of current Foreign Agent. It is only included when mobile node is using Mobile IPv4
Access Router Address	IP_ADDRESS	(Optional) IP address of current Access Router. It is only included when mobile node is using IPv6

7.4.19.1.3 When generated

This primitive is generated by an MIH User in the mobile node that may want to query other candidate networks for a possible handover. MN uses the QueryResourceList parameter to notify the serving PoS of the minimal resource requirement at the candidate networks in order for the handover to be successful. An MIH User on MN may generate this primitive when it wants to query IP address related information from the candidate networks before handover.

7.4.19.1.4 Effect on receipt

Upon receipt of this primitive by the local MIHF, it shall generate and send an MIH_MN_HO_Candidate_Query request message to the remote MIHF identified by the Destination Identifier.

7.4.19.2 MIH_MN_HO_Candidate_Query.indication

7.4.19.2.1 Function

This primitive is used by MIHF to indicate the receipt of MIH_MN_HO_Candidate_Query request message from a mobile node.

7.4.19.2.2 Semantics of service primitive

```
MIH_MN_HO_Candidate_Query.indication (
                    SourceIdentifier,
                    SourceLinkIdentifier,
                    CandidateLinkList,
                    QueryResourceList,
                    IPConfigurationMethods,
                    DHCPServerAddress,
                    FAAddress,
                    AccessRouterAddress
                    )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link for handover.
Candidate Link List	LIST(LINK_POA_LIST)	A list of PoAs, identifying candidate networks to which handover needs to be initiated. The list is sorted from most preferred first to least preferred last.
Query Resource List	QOS_LIST	List of resources to be queried at candidate networks.
IP Configuration Methods	IP_CONFIG_METHODS	(Optional) Current IP configuration methods.
DHCP Server Address	IP_ADDRESS	(Optional) IP address of current DHCP Server. It is only included when mobile node is using dynamic address configuration
FA Address	IP_ADDRESS	(Optional) IP address of current Foreign Agent. It is only included when mobile node is using Mobile IPv4
Access Router Address	IP_ADDRESS	(Optional) IP address of current Access Router. It is only included when mobile node is using IPv6

7.4.19.2.3 When generated

This primitive is generated by MIHF on receiving MIH_MN_HO_Candidate_Query request message from a peer MIHF in a mobile node.

7.4.19.2.4 Effect on receipt

The MIH User may invoke MIH_N2N_HO_Query_Resources.request primitive to exchange MIH_N2N_HO_Query_Resource messages with MIHF in one or more candidate networks under consideration before invoking the MIH_MN_HO_Candidate_Query.response primitive.

7.4.19.3 MIH_MN_HO_Candidate_Query.response

7.4.19.3.1 Function

This primitive is used by MIH Users to inform MIHF of the result of the candidate query request.

7.4.19.3.2 Semantics of service primitive

```
MIH_MN_HO_Candidate_Query.response (
                DestinationIdentifier,
                SourceLinkIdentifier,
                PreferredCandidateLinkList,
                AvailableResourceSet,
                IPConfigurationMethods,
                DHCPServerAddress,
                FAAddress,
                AccessRouterAddress,
                Status
                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this response.
Source Link Identifier	LINK_TUPLE_ID.	This identifies the source link.
Preferred Candidate Link List	LIST(LINK_POA_LIST)	A list of PoAs, identifying candidate networks to which handover needs to be initiated. The list is sorted from most preferred first to least preferred last. Note, this parameter is not included if Status does not indicate “Success”.
Available Resource Set	LIST(QOS_LIST)	A list of available resources for each PoA in PreferredCandidateLinkList. The order of this list is the same as the PreferredCandidateLinkList. Note, this parameter is not included if Status does not indicate “Success”.
IP Configuration Methods	IP_CONFIG_METHODS	IP configuration methods applicable for all PreferredCandidateLinkList. Note, this parameter is not included if Status does not indicate “Success”.
DHCP Server Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate DHCP Server. It is only included when dynamic address configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.
FA Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate Foreign Agent. It is only included Mobile IPv4 is supported. Note, this parameter is not included if Status does not indicate “Success”.
Access Router Address List	LIST(IP_ADDRESS)	(Optional) IP address of candidate Access Router. It is only included IPv6 Stateless configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.19.3.3 When generated

The MIH User invokes this primitive in response to an MIH_MN_HO_Candidate_Query request message from a peer MIHF entity in mobile node and possibly after the exchange of MIH_N2N_HO_Query_Resources messages with the MIHF in the candidate networks.

7.4.19.3.4 Effect on receipt

Upon receipt of this primitive MIHF shall send a response message to the destination.

7.4.19.4 MIH_MN_HO_Candidate_Query.confirm

7.4.19.4.1 Function

This primitive is used by MIHF to inform MIH Users of the receipt of candidate query and IP address related information response.

7.4.19.4.2 Semantics of service primitive

MIH_MN_HO_Candidate_Query.confirm (
SourceIdentifier,
SourceLinkIdentifier,
PreferredCandidateLinkList,
AvailableResourceSet,
IPConfigurationMethods,
DHCPServerAddress,
FAAddress,
AccessRouterAddress,
Status
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which is a remote MIHF.
Source Link Identifier	LINK_TUPLE_ID.	This identifies the source link for handover.
Preferred Candidate Link List	LIST(LINK_POA_LIST)	A list of PoAs, identifying candidate networks to which handover needs to be initiated. The list is sorted from most preferred first to least preferred last. Note, this parameter is not included if Status does not indicate “Success”.
Available Resource Set	LIST(QOS_LIST)	A list of available resources for each PoA in Preferred-CandidateLinkList. The order of this list is the same as the PreferredCandidateLinkList. Note, this parameter is not included if Status does not indicate “Success”.
IP Configuration Methods	IP_CONFIG_METHODS	IP configuration methods on PreferredCandidateLinkList. Note, this parameter is not included if Status does not indicate “Success”.
DHCP Server Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate DHCP Server. It is only included when dynamic address configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.

FA Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate Foreign Agent. It is only included Mobile IPv4 is supported. Note, this parameter is not included if Status does not indicate “Success”.
Access Router Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate Access Router. It is only included IPv6 Stateless configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.19.4.3 When generated

This primitive is generated by MIHF on receiving MIH_MN_HO_Candidate_Query response message from a peer MIHF in the network.

7.4.19.4.4 Effect on receipt

On receiving the primitive the MIH User entity which originally initiated the candidate query request may decide to choose the candidate network for handover or abort it based on the list of PoA, available resources, and the IP address related information. However, if Status does not indicate “Success”, the recipient shall ignore any other returned values and, instead, shall perform appropriate error handling.

7.4.20 MIH_N2N_HO_Query_Resources

7.4.20.1 MIH_N2N_HO_Query_Resources.request

7.4.20.1.1 Function

This primitive is used by an MIHF on the serving network to communicate with its peer MIHF on the candidate network. This is used to query the available link resource and IP address related information of the candidate network.

7.4.20.1.2 Semantics of service primitive

```
MIH_N2N_HO_Query_Resources.request (
                                    DestinationIdentifier,
                                    QueryResourceList,
                                    IPConfigurationMethods,
                                    DHCPServerAddress,
                                    FAAddress,
                                    AccessRouterAddress,
                                    CandidateLinkList
                                    )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this request.
Query Resource List	QOS_LIST	List of resources to be queried at the candidate network.

IP Configuration Methods	IP_CONFIG_METHODS	(Optional) Current IP configuration methods.
DHCP Server Address	IP_ADDRESS	(Optional) IP address of current DHCP Server. It is only included when mobile node is using dynamic address configuration
FA Address	IP_ADDRESS	(Optional) IP address of current Foreign Agent. It is only included when mobile node is using Mobile IPv4
Access Router Address	IP_ADDRESS	(Optional) IP address of current Access Router. It is only included when mobile node is using IPv6
Candidate Link List	LIST(LINK_ID)	(Optional) A list of candidate links (i.e., APs or BSs) on a specific candidate network. In this list, each link is indicated by its link type and a PoA MAC address.

7.4.20.1.3 When generated

This primitive is generated after receiving the MIH_MN_HO_Candidate_Query request message from the MIHF on the mobile node in the case of mobile-initiated handover. In the case of network-initiated handover, this primitive is generated after receiving the MIH_Net_HO_Candidate_Query response message from the mobile node.

7.4.20.1.4 Effect on receipt

Upon receipt of this primitive MIHF shall send a request message to the destination.

7.4.20.2 MIH_N2N_HO_Query_Resources.indication

7.4.20.2.1 Function

The MIHF on the candidate network indicates that an MIH_N2N_HO_Query_Resources request message is received from a remote MIHF on the serving network so that the upper layer entity can identify the link resource usage and provide IP address related information for the impending handover.

7.4.20.2.2 Semantics of service primitive

MIH_N2N_HO_Query_Resources.indication (
SourceIdentifier,
QueryResourceList,
IPConfigurationMethods,
DHCPServerAddress,
FAAddress,
AccessRouterAddress,
CandidateLinkList
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_N2N_HO_Candidate_Query request message.
Query Resource List	QOS_LIST	List of resources to be queried at the new candidate network.

IP Configuration Methods	IP_CONFIG_METHODS	(Optional) Current IP configuration methods.
DHCP Server Address	IP_ADDRESS	(Optional) IP address of current DHCP Server. It is only included when mobile node is using dynamic address configuration
FA Address	IP_ADDRESS	(Optional) IP address of current Foreign Agent. It is only included when mobile node is using Mobile IPv4
Access Router Address	IP_ADDRESS	(Optional) IP address of current Access Router. It is only included when mobile node is using IPv6
Candidate Link List	LIST(LINK_ID)	(Optional) A list of candidate links (i.e., APs or BSs) on a specific candidate network. In this list, each link is indicated by its link type and a PoA MAC address.

7.4.20.2.3 When generated

This primitive is generated by MIHF when the MIHF on the candidate network receives MIH_N2N_HO_Query_Resources request message from a peer MIHF on the serving network.

7.4.20.2.4 Effect on receipt

The MIH User on the candidate network identifies the link resource usage for the impending handover. It also replies with MIH_N2N_HO_Query_Resources.response primitive.

7.4.20.3 MIH_N2N_HO_Query_Resources.response

7.4.20.3.1 Function

This primitive is used by an MIHF on the candidate network to communicate with its peer MIHF on the serving network which sent out an MIH_N2N_HO_Query_Resources request. This is used to respond with the result of any resource preparation for the impending handover and to notify the MIHF on the serving network of the link resource status of the candidate network. It is also used to provide IP address related information of the candidate networks.

7.4.20.3.2 Semantics of service primitive

```
MIH_N2N_HO_Query_Resources.response (
                DestinationIdentifier,
                ResourceStatus,
                AvailableResourceSet,
                IPConfigurationMethods,
                DHCPServerAddress,
                FAAddress,
                AccessRouterAddress,
                CandidateLinkList,
                IPAddressInformationStatus,
                Status
                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_N2N_HO_Query_Resources request message.
Resource Status	LINK_RESOURCE_STATUS	Specifies whether requested resources are available or not at the new PoA. Note, this parameter is not included if Status does not indicate “Success”.
Available Resource Set	LIST(QOS_LIST)	A list of available resources for each link. Note, this parameter is not included if Status does not indicate “Success”.
IP Configuration Methods	IP_CONFIG_METHODS	(Optional) IP configuration methods. Note, this parameter is not included if Status does not indicate “Success”.
DHCP Server Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate DHCP Server. It is only included when dynamic address configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.
FA Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate Foreign Agent. It is only included Mobile IPv4 is supported. Note, this parameter is not included if Status does not indicate “Success”.
Access Router Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate Access Router. It is only included IPv6 Stateless configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.
Candidate Link List	LIST(LINK_ID)	(Optional) A list of candidate links (i.e., APs or BSs) on a specific candidate network. In this list, each link is indicated by its link type and a PoA MAC address. Note, this parameter is not included if Status does not indicate “Success”.
IP Address Information Status	IP_CONFIG_STATUS	The result of IP configuration. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.20.3.3 When generated

The MIHF on the candidate network invokes this primitive in response to an MIH_N2N_HO_Query_Resources request message from a peer MIHF entity on the serving network.

7.4.20.3.4 Effect on receipt

Upon receipt of this primitive MIHF shall send a response message to the destination.

7.4.20.4 MIH_N2N_HO_Query_Resources.confirm

7.4.20.4.1 Function

This primitive is generated by the MIHF on the serving network to respond with the result of any resource preparation for the impending handover and to notify the link resource status of the candidate network. It also carries IP address related information on the candidate networks to MIH Users on the serving network.

7.4.20.4.2 Semantics of service primitive

```
MIH_N2N_HO_Query_Resources.confirm (
                                SourceIdentifier,
                                ResourceStatus,
                                AvailableResourceSet,
                                IPConfigurationMethods,
                                DHCPServerAddress,
                                FAAddress,
                                AccessRouterAddress,
                                CandidateLinkList,
                                IPAddressInformationStatus,
                                Status
                                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_N2N_HO_Candidate_Query response message.
ResourceStatus	LINK_RESOURCE_STATUS	Specifies whether requested resources are available or not at the new PoA. Note, this parameter is not included if Status does not indicate “Success”.
Available Resource Set	LIST(QOS_LIST)	A list of available resource for each link. Note, this parameter is not included if Status does not indicate “Success”.
IP Configuration Methods	IP_CONFIG_METHODS	(Optional) IP configuration methods. Note, this parameter is not included if Status does not indicate “Success”.
DHCP Server Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate DHCP Server. It is only included when dynamic address configuration is supported. Note, this parameter is not included if Status does not indicate “Success”.
FA Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate Foreign Agent. It is only included Mobile IPv4 is supported. Note, this parameter is not included if Status does not indicate “Success”.

Access Router Address	LIST(IP_ADDRESS)	(Optional) IP address of candidate Access Router. It is only included IPv6 Stateless configuration is supported. Note, this parameter is not included if Status does not indicate "Success".
Candidate Link List	LIST(LINK_ID)	(Optional) A list of candidate links (i.e., APs or BSs) on a specific candidate network. In this list, each link is indicated by its link type and a PoA MAC address. Note, this parameter is not included if Status does not indicate "Success".
IP Address Information Status	IP_CONFIG_STATUS	The result of IP configuration. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.20.4.3 When generated

This primitive is generated by the MIHF when the MIHF on the serving network receives an MIH_N2N_HO_Query_Resources response message from a peer MIHF on the candidate network.

7.4.20.4.4 Effect on receipt

After receiving this primitive, the MIH User on the serving network may send an MIH_MN_HO_Candidate_Query.response primitive to the MIHF in case it was indicated with MIH_MN_HO_Candidate_Query.indication primitive before. However, if Status does not indicate "Success", the recipient shall ignore any other returned values and, instead, shall perform appropriate error handling.

7.4.21 MIH_Net_HO_Commit

7.4.21.1 MIH_Net_HO_Commit.request

7.4.21.1.1 Function

This primitive is used by an MIH User on the network to communicate with the remote MIH User on the mobile node. The primitive is used to request the peer MIH User the commitment to perform link handover based on selected choices for candidate networks and PoA.

7.4.21.1.2 Semantics of service primitive

```
MIH_Net_HO_Commit.request       (
                                DestinationIdentifier,
                                LinkType,
                                TargetNetworkInfoList,
                                LinkActionSetList,
                                AssignedResourceSet
                                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	Shall contain the MIHF ID of the MN MIHF that is to be committed.
Link Type	LINK_TYPE	Contains target link type.
Target Network Info List	LIST(TARGET_NETWORK_INFO)	Contains target network information for handover.
Link Action Set List	LIST(LINK_ACTION_REQ)	(Optional) A list of handover actions for the links.
Assigned Resource Set	ASSIGNED_RESOURCE_SET	This includes the set of resource parameters assigned to the MN for performing the handover

7.4.21.1.3 When generated

The MIH User generates this primitive to order specific handover actions on one or more links.

7.4.21.1.4 Effect on receipt

Upon receipt of this primitive MIHF shall send a request message to the destination.

7.4.21.2 MIH_Net_HO_Commit.indication

7.4.21.2.1 Function

This primitive is generated by an MIHF to indicate that an MIH_Net_HO_Commit request message has been received from a peer MIHF.

7.4.21.2.2 Semantics of service primitive

```
MIH_Net_HO_Commit.indication (
                                SourceIdentifier,
                                LinkType,
                                TargetNetworkInfoList,
                                LinkActionSetList,
                                AssignedResourceSet
                                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_Net_HO_Commit request message.
Link Type	LINK_TYPE	Contains target link type.
Target Network Info List	LIST(TARGET_NETWORK_INFO)	Contains target network information for handover.

Link Action Set List	LIST(LINK_ACTION_REQ)	(Optional) A list of handover actions for the links.
Assigned Resource Set	ASSIGNED_RESOURCE_SET	This includes the set of resource parameters assigned to the MN for performing the handover

7.4.21.2.3 When generated

This primitive is generated by an MIHF on receiving an MIH_Net_HO_Commit request message from a peer MIHF.

7.4.21.2.4 Effect on receipt

The MIH User receiving this primitive replies with an MIH_Net_HO_Commit.response primitive.

7.4.21.3 MIH_Net_HO_Commit.response

7.4.21.3.1 Function

This primitive is used by an MIHF to communicate with a peer MIHF from which an MIH_Net_HO_Commit request message is received. The primitive is used to communicate the response of a handover commit request.

7.4.21.3.2 Semantics of service primitive

MIH_Net_HO_Commit.response (
DestinationIdentifier,
LinkType,
TargetNetworkInfo,
LinkActionResultList,
Status
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_Net_HO_Commit request message.
LinkType	LINK_TYPE	Contains target link type. Note, this parameter is not included if Status does not indicate "Success".
TargetNetworkInfo	TARGET_NETWORK_INFO	Contains target network information for handover. Note, this parameter is not included if Status does not indicate "Success".
Link Action Result List	LIST(LINK_ACTION_RSP)	(Optional) A list of link actions result. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.21.3.3 When generated

This primitive is generated in response to an MIH_Net_HO_Commit.indication primitive.

7.4.21.3.4 Effect on receipt

Upon receipt, the old Serving PoS is informed about the status of the previously issued command request.

Since MIH_Net_HO_Commit contains actions to effect the handover, and the response has the status of those actions, the link to the old PoS may not be accessible (e.g. break before make) to receive the response before L3 connectivity has been established on the new link and only if the MN knows the old PoS L3 address thus, the old PoS may not receive this response.

7.4.21.4 MIH_Net_HO_Commit.confirm

7.4.21.4.1 Function

This primitive is generated by the MIHF to confirm that an MIH_Net_HO_Commit response message is received from a peer MIHF.

7.4.21.4.2 Semantics of service primitive

```
MIH_Net_HO_Commit.confirm (
                        SourceIdentifier,
                        LinkType,
                        TargetNetworkInfo,
                        LinkActionResultList,
                        Status
                        )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_Net_HO_Commit response message.
LinkType	LINK_TYPE	Contains target link type. Note, this parameter is not included if Status does not indicate “Success”.
TargetNetworkInfo	TARGET_NETWORK_INFO	Contains target network information for handover. Note, this parameter is not included if Status does not indicate “Success”.
Link Action Result List	LIST(LINK_ACTION_RSP)	(Optional) A list of link actions result. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.21.4.3 When generated

This primitive is generated by the MIHF on receiving an MIH_Net_HO_Commit response message from a peer MIHF.

7.4.21.4.4 Effect on receipt

Upon receipt, the old serving PoS is informed about the status of the previously issued command request.

7.4.22 MIH_MN_HO_Commit

7.4.22.1 MIH_MN_HO_Commit.request

7.4.22.1.1 Function

This primitive is used by MIH Users on a mobile node to commit locally a handover to a specific target network.

7.4.22.1.2 Semantics of service primitive

```
MIH_MN_HO_Commit.request (
                          SourceLinkIdentifier,
                          TargetLinkIdentifier,
                          SourceLinkAction.,
                          TargetLinkAction
                          )
```

Parameters:

Name	Type	Description
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link.
Target Link Identifier	LINK_TUPLE_ID	This is the identifier target network to which handover needs to be initiated.
Source Link Action	LINK_ACTION	Specifies the action on source link during handover.
Target Link Action	LINK_ACTION	Specifies the action on target link during handover.

7.4.22.1.3 When generated

This primitive may be generated when a mobile node decides to locally perform a handover.

7.4.22.1.4 Effect on receipt

Upon receipt of this primitive by the local MIHF, it shall generate corresponding link actions to perform the specified handover.

7.4.22.2 MIH_MN_HO_Commit.confirm

7.4.22.2.1 Function

This primitive is used by MIHF to inform MIH Users the results of an earlier MIH_MN_HO_Commit.request.

7.4.22.2.2 Semantics of service primitive

MIH_MN_HO_Commit.confirm (
SourceLinkIdentifier,
TargetLinkIdentifier,
SourceLinkActionResult,
TargetLinkActionResult,
Status
)

Parameters:

Name	Type	Description
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link.
Target Link Identifier	LINK_TUPLE_ID	This identifies the target link.
Source Link Action Result	LINK_AC_RESULT_CODE	The result status of the source link action. Note, this parameter is not included if Status does not indicate "Success".
Target Link Action Result	LINK_AC_RESULT_CODE	The result status of the target link action. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.22.2.3 When generated

This primitive may be generated by an MIHF when it completes the execution of an MIH_MN_HO_Commit.request.

7.4.22.2.4 Effect on receipt

MN may now proceed with the next step in handover.

7.4.23 MIH_N2N_HO_Commit

7.4.23.1 MIH_N2N_HO_Commit.request

7.4.23.1.1 Function

This primitive is used by an MIH User on the serving network to inform a selected target network that a mobile node is about to move to the target network.

7.4.23.1.2 Semantics of service primitive

The parameters of the primitive are as follows:

MIH_N2N_HO_Commit.request (
DestinationIdentifier,
MobileNodeIdentifier,
TargetMobileNodeLinkIdentifier,
TargetPoA,
PreferredLinkList,
RequestedResourceSet
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this request.
Mobile Node Identifier	MIHF_ID	This identifies an MIHF of the mobile node which commits to perform handover action.
Target Mobile Node Link Identifier	LINK_ID	This is the identifier of the mobile node's target link for which resources are requested.
Target PoA	MAC_ADDRESS	This is the MAC address of the target Point of Attachment (AP/BS).
Preferred Link List	LIST(LINK_POA_LIST)	A list of PoAs for each link, suggesting the new access networks and PoAs to which handover initiation should be considered. This may be different than the networks that were suggested in the handover query. The list is sorted from most preferred first to least preferred last.
Requested Resource Set	REQUESTED_RESOURCE_SET	This includes the set of parameters required for performing MN admission control and resource reservation at the target network

7.4.23.1.3 When generated

The MIH User on the serving network invokes this primitive when a single target network has been decided.

7.4.23.1.4 Effect on receipt

Upon receipt of this primitive by the local MIHF, it shall generate and send an MIH_N2N_HO_Commit request message to the remote MIHF on the selected target network identified by the Destination Identifier.

7.4.23.2 MIH_N2N_HO_Commit.indication

7.4.23.2.1 Function

This primitive is generated by an MIHF to indicate that an MIH_N2N_HO_Commit request message has been received from a peer MIHF on serving network.

7.4.23.2.2 Semantics of service primitive

The parameters of the primitive are as follows:

MIH_N2N_HO_Commit.indication(
SourceIdentifier,
MobileNodeIdentifier,
TargetMobileNodeLinkIdentifier,
TargetPoA,
PreferredLinkList,
RequestedResourceSet
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	This identifies the invoker of this primitive which can be a remote MIHF.
Mobile Node Identifier	MIHF_ID	This identifies an MIHF of the mobile node which commits to perform handover action.
Target Mobile Node Link Identifier	LINK_ID	This is the identifier of the mobile node's target link for which resources are requested.
Target PoA	MAC_ADDRESS	This is the MAC address of the target Point of Attachment (AP/BS).
Preferred Link List	LIST(LINK_POA_LIST)	A list of PoAs for each link, suggesting the new access networks and PoAs to which handover initiation should be considered. This may be different than the networks that were suggested in the handover query. The list is sorted from most preferred first to least preferred last.
Requested Resource Set	REQUESTED_RESOURCE_SET	This includes the set of parameters required for performing MN admission control and resource reservation at the target network

7.4.23.2.3 When generated

This primitive is generated by an MIHF on receiving an MIH_N2N_HO_Commit request message from a peer MIHF on the serving network.

7.4.23.2.4 Effect on receipt

Upon receipt of this primitive, MIH User shall generate an MIH_N2N_HO_Commit response primitive.

7.4.23.3 MIH_N2N_HO_Commit.response

7.4.23.3.1 Function

This primitive is used by an MIH User to respond to an MIH_N2N_HO_Commit indication primitive.

7.4.23.3.2 Semantics of service primitive

The parameters of the primitive are as follows:

MIH_N2N_HO_Commit.response (
DestinationIdentifier,
MobileNodeIdentifier,
TargetLinkIdentifier,

AssignedResourceSet,
Status
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ ID	This identifies a remote MIHF which will be the destination of this request.
Mobile Node Identifier	MIHF_ID	This identifies an MIHF of the mobile node which commits to perform handover action.
Target Link Identifier	LINK_TUPLE_ID	This contains the identifier of the target Point of Attachment (AP/BS) for the MN. Note, this parameter is not included if Status does not indicate "Success".
Assigned Resource Set	ASSIGNED_ RESOURCE_SET	This includes the set of resource parameters assigned by the target network to the MN. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.23.3.3 When generated

This primitive is generated by an MIHF User in response to a received MIH_N2N_HO_Commit indication primitive.

7.4.23.3.4 Effect on receipt

Upon receipt, the MIHF shall generate and send an MIH_N2N_HO_Commit response message to the peer MIHF on the serving network which sent an MIH_N2N_HO_Commit request message.

7.4.23.4 MIH_N2N_HO_Commit.confirm

7.4.23.4.1 Function

This primitive is generated by the MIHF to confirm that an MIH_N2N_HO_Commit response message is received from a peer MIHF on the selected target network.

7.4.23.4.2 Semantics of service primitive

The parameters of the primitive are as follows:

MIH_N2N_HO_Commit.confirm (
SourceIdentifier,
MobileNodeIdentifier,
TargetLinkIdentifier,
AssignedResourceSet,
Status
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_N2N_HO_Commit response message.
Mobile Node Identifier	MIHF_ID	This identifies an MIHF of the mobile node which commits to perform handover action.
Target Link Identifier	LINK_TUPLE_ID	This contains the identifier of the target Point of Attachment (AP/BS) for the MN. Note, this parameter is not included if Status does not indicate “Success”.
Assigned Resource Set	ASSIGNED_ RESOURCE_SET	This includes the set of resource parameters assigned by the target network to the MN. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.23.4.3 When generated

This primitive is generated by the MIHF on receiving an MIH_N2N_HO_Commit response message from a peer MIHF on the selected target network.

7.4.23.4.4 Effect on receipt

Upon receipt, the serving network is informed about the status of the previously issued command request so that it can react accordingly. For instance, the serving network may determine that the handover procedure is acknowledged by the target network and it can notify the MN to perform handover. However, if Status does not indicate “Success”, the recipient shall ignore any other returned values and, instead, shall perform appropriate error handling.

7.4.24 MIH_MN_HO_Complete

7.4.24.1 MIH_MN_HO_Complete.request

7.4.24.1.1 Function

This primitive may be optionally used by MIH Users to indicate the completion of MIH level handover aiding procedure.

7.4.24.1.2 Semantics of service primitive

```
MIH_MN_HO_Complete.request (
                DestinationIdentifier,
                SourceLinkIdentifier,
                TargetLinkIdentifier,
                HandoverResult
                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	This identifies a remote MIHF which will be the destination of this request.
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	This identifies the target link.
Handover Result	HANDOVER_RESULT	Handover result.

7.4.24.1.3 When generated

This primitive may be generated when MIH level handover procedure is complete.

7.4.24.1.4 Effect on receipt

Upon receipt of this primitive by the local MIHF, it shall generate and send an MIH_MN_HO_Complete request message to the remote MIHF identified by the Destination Identifier.

7.4.24.2 MIH_MN_HO_Complete.indication

7.4.24.2.1 Function

This primitive is used by MIHF to inform MIH Users locally that an MIH_MN_HO_Complete request message is received.

7.4.24.2.2 Semantics of service primitive

```
MIH_MN_HO_Complete.indication (
                                SourceIdentifier,
                                SourceLinkIdentifier,
                                TargetLinkIdentifier,
                                HandoverResult
                                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_MN_HO_Complete request message.
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	This identifies the target link.
Handover Result	HANDOVER_RESULT	Handover result.

7.4.24.2.3 When generated

This primitive shall be generated when an MIH_MN_HO_Complete request message is received.

7.4.24.2.4 Effect on receipt

This indicates the completion of the handover. A corresponding response is generated.

7.4.24.3 MIH_MN_HO_Complete.response

7.4.24.3.1 Function

This primitive is used by MIH Users to send a response to MIH_MN_HO_Complete request.

7.4.24.3.2 Semantics of service primitive

```
MIH_MN_HO_Complete.response (
                DestinationIdentifier,
                SourceLinkIdentifier,
                TargetLinkIdentifier,
                Status
                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_MN_HO_Complete request message.
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	This identifies the target link.
Status	STATUS	Status of operation.

7.4.24.3.3 When generated

This primitive shall be generated when MIH_MN_HO_Complete request message is received.

7.4.24.3.4 Effect on receipt

This indicates the completion of the MIH level handover aiding procedure.

7.4.24.4 MIH_MN_HO_Complete.confirm

7.4.24.4.1 Function

This primitive is used by MIHF to inform MIH Users locally that an MIH_MN_HO_Complete response message is received.

7.4.24.4.2 Semantics of service primitive

```
MIH_MN_HO_Complete.confirm (
                SourceIdentifier,
                SourceLinkIdentifier,
                TargetLinkIdentifier,
                Status
                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_MN_HO_Complete response message.
Source Link Identifier	LINK_TUPLE_ID	This identifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	This identifies the target link.
Status	STATUS	Status of operation.

7.4.24.4.3 When generated

MIHF shall generate this primitive when an MIH_MN_HO_Commit response message is received.

7.4.24.4.4 Effect on receipt

This indicates the completion of the MIH level handover aiding procedure.

7.4.25 MIH_N2N_HO_Complete

7.4.25.1 MIH_N2N_HO_Complete.request

7.4.25.1.1 Function

This primitive is used by an MIH User in the network to communicate with a peer network MIH entity about the completion of handover operation.

7.4.25.1.2 Semantics of service primitive

```
MIH_N2N_HO_Complete.request (
                                DestinationIdentifier,
                                SourceLinkIdentifier,
                                TargetLinkIdentifier,
                                HandoverResult
                                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	Identify the MIHF ID of the destination node.
Source Link Identifier	LINK_TUPLE_ID	Specifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	Specifies the target link of the handover.
Handover Result	HANDOVER_RESULT	Handover result.

7.4.25.1.3 When generated

The MIH User invokes this primitive when handover operations have been completed.

7.4.25.1.4 Effect on receipt

Upon receipt of this primitive by the local MIHF, it shall generate and send an MIH_N2N_HO_Complete request message to the remote MIHF identified by the Destination Identifier.

7.4.25.2 MIH_N2N_HO_Complete.indication

7.4.25.2.1 Function

This primitive is generated by the MIHF to indicate the status of the handover operation.

7.4.25.2.2 Semantics of service primitive

MIH_N2N_HO_Complete.indication(
SourceIdentifier,
SourceLinkIdentifier,
TargetLinkIdentifier,
HandoverResult
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_N2N_HO_Complete request message.
Source Link Identifier	LINK_TUPLE_ID	Specifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	Specifies the target link of the handover.
Handover Result	HANDOVER_RESULT	Handover result.

7.4.25.2.3 When generated

This primitive is generated by the MIHF on receiving an MIH_N2N_HO_Complete request message from a peer MIHF.

7.4.25.2.4 Effect on receipt

The MIH User receiving this primitive replies with an MIH_N2N_HO_Complete.response primitive.

7.4.25.3 MIH_N2N_HO_Complete.response

7.4.25.3.1 Function

This primitive is used to send a response to a handover complete request.

7.4.25.3.2 Semantics of service primitive

MIH_N2N_HO_Complete.response (
DestinationIdentifier,
SourceLinkIdentifier,
TargetLinkIdentifier,

ResourceStatus,
Status
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_N2N_HO_Complete request message.
Source Link Identifier	LINK_TUPLE_ID	Specifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	Specifies the target link of the handover.
Resource Status	LINK_RESOURCE_RETENTION_STATUS	Status of resource. Note, this parameter is not included if Status does not indicate "Success".
Status	STATUS	Status of operation.

7.4.25.3.3 When generated

The MIH User responds with this primitive after processing the handover complete request.

7.4.25.3.4 Effect on receipt

Upon receipt, the MIHF in the new network may forward this message to the MN.

7.4.25.4 MIH_N2N_HO_Complete.confirm

7.4.25.4.1 Function

This primitive is generated by the MIHF to confirm that an MIH_N2N_HO_Complete response message is received from a peer MIHF.

7.4.25.4.2 Semantics of service primitive

MIH_N2N_HO_Complete.confirm (
SourceIdentifier,
SourceLinkIdentifier,
TargetLinkIdentifier,
ResourceStatus,
Status
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_N2N_HO_Complete response message.

Source Link Identifier	LINK_TUPLE_ID	Specifies the source link of the handover.
Target Link Identifier	LINK_TUPLE_ID	Specifies the target link of the handover.
Resource Status	LINK_RESOURCE_RETENTION_STATUS	Status of resource. Note, this parameter is not included if Status does not indicate “Success”.
Status	STATUS	Status of operation.

7.4.25.4.3 When generated

This primitive is generated by MIHF on receiving MIH_N2N_HO_Complete response message from a peer MIHF.

7.4.25.4.4 Effect on receipt

Upon receipt, the MIH User may determine that the handover complete request was processed successfully. However, if Status does not indicate “Success”, the recipient shall ignore any other returned values and, instead, shall perform appropriate error handling.

7.4.26 MIH_Get_Information

7.4.26.1 MIH_Get_Information.request

7.4.26.1.1 Function

This primitive is used by an MIH User to request information from an MIH information server. The information query may be related to a specific interface, attributes to the network interface, as well as the entire network capability. The service primitive has the flexibility to query either a specific data within a network interface or extended schema of a given network. It is assumed that the available information could be broadcast in access technology specific manner such as in 802.11 and 802.16.

7.4.26.1.2 Semantics of service primitive

```
MIH_Get_Information.request     (
                                DestinationIdentifier,
                                InfoQueryBinaryDataList,
                                InfoQueryRDFDataList,
                                InfoQueryRDFSchemaURL,
                                InfoQueryRDFSchemaList,
                                MaxResponseSize,
                                UnauthenticatedInformationRequest
                                )
```

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	The local MIHF or a remote MIHF which will be the destination of this request.

Info Query Binary Data List	LIST(INFO_QUERY_BINARY_DATA)	(Optional) A list of TLV queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS. See Table B-15 for detailed definition.
Info Query RDF Data List	LIST(INFO_QUERY_RDF_DATA)	(Optional) A list of RDF queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS. See Table B-16 for detailed definition.
Info Query RDF Schema URL	NULL	(Optional) An RDF Schema URL query.
Info Query RDF Schema List	LIST(INFO_QUERY_RDF_SCHEMA)	(Optional) A list of RDF schema queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS.
Max Response Size	UNSIGNED_INT(2)	(Optional) This field specifies the maximum size of Info Response parameters in MIH_Get_Information response primitive in octets. If this field is not specified, the maximum size is set to 65,535. The actual maximum size forced by the IS server may be smaller than that specified by the IS client.
Unauthenticated Information Request	BOOLEAN	The value of UIR bit to be set in the MIH_Get_Information request message sent to the remote MIHF.

One and only one of the following parameters shall be specified:

— Info Query Binary Data List

— Info Query RDF Data List

— Info Query RDF Schema URL

— Info Query RDF Schema List

7.4.26.1.3 When generated

This primitive is generated by an MIH User that is seeking to retrieve information.

The order of the queries in each of Info Query Binary Data List, Info Query RDF Data List and Info Query RDF Schema List parameters identifies the priority of the query. The first query has the highest priority to be processed by MIIS.

7.4.26.1.4 Effect on receipt

The recipient will forward the query in an MIH message to the designated MIIS server. The MIIS server tries to interpret the query request and retrieve the specified information.

7.4.26.2 MIH_Get_Information.indication

7.4.26.2.1 Function

This primitive is generated by the MIHF to indicate that an MIH_Get_Information Request message is received from a peer MIHF.

7.4.26.2.2 Semantics of service primitive

MIH_Get_Information.indication (

SourceIdentifier,
InfoQueryBinaryDataList,
InfoQueryRDFDataList,
InfoQueryRDFSchemaURL,
InfoQueryRDFSchemaList,
MaxResponseSize,
UnauthenticatedInformationRequest
)

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that sent the MIH_GET_Information request message.
Info Query Binary Data List	LIST(INFO_QUERY_ BINARY_DATA)	(Optional) A list of TLV queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS. See Table B-15 for detailed definition.
Info Query RDF Data List	LIST(INFO_QUERY_ RDF_DATA)	(Optional) A list of RDF queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS.
Info Query RDF Schema URL	NULL	(Optional) An RDF Schema URL query.
Info Query RDF Schema List	LIST(INFO_QUERY_ RDF_SCHEMA)	(Optional) A list of RDF schema queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS.
Max Response Size	UNSIGNED_INT(2)	(Optional) This field specifies the maximum size of Info Response parameters in MIH_Get_Information response primitive in octets. If this field is not specified, the maximum size is set to 65,535. The actual maximum size forced by the IS server may be smaller than that specified by the IS client.
Unauthenticated Information Request	BOOLEAN	The value of UIR bit contained in the MIH_Get_Information request message received from the remote MIHF.

7.4.26.2.3 When generated

This primitive is generated by the MIHF on receiving an MIH_Get_Information request message from a peer MIHF. The order of the queries in each of Info Query Binary Data List, Info Query RDF Data List and Info Query RDF Schema List parameters identifies the priority of the query. The first query has the highest priority to be processed by MIIS. Thus the order of the queries shall be maintained as indicated by the request message.

7.4.26.2.4 Effect on receipt

The recipient tries to interpret the query request and retrieve the specified information. Once the information is retrieved, the recipient replies with the MIH_Get_Information.response primitive.

7.4.26.3 MIH_Get_Information.response

7.4.26.3.1 Function

This primitive is used by an MIH User, i.e., MIIS Server, to respond to an MIH_GET_Information.indication primitive.

7.4.26.3.2 Semantics of service primitive

MIH_Get_Information.response (
DestinationIdentifier,
InfoResponseBinaryDataList,
InfoResponseRDFDataList,
InfoResponseRDFSchemaURLList,
InfoResponseRDFSchemaList,
Status
)

Parameters:

Name	Type	Description
Destination Identifier	MIHF_ID	The local MIHF or a remote MIHF which will be the destination of this response.
Info Response Binary Data List	LIST(INFO_RSP_BINARY_DATA)	(Optional) A list of TLV query responses. The list may be sorted from most preferred first to least preferred last.
Info Response RDF Data List	LIST(INFO_RSP_RDF_DATA)	(Optional) A list of RDF query responses. The list may be sorted from most preferred first to least preferred last.
Info Response RDF Schema URL List	LIST(INFO_RSP_RDF_URL)	(Optional) A list of RDF Schema URL. The list may be sorted from most preferred first to least preferred last.
Info Response RDF Schema List	LIST(INFO_RSP_RDF_SCHEMA)	(Optional) A list of RDF schema query responses. The list may be sorted from most preferred first to least preferred last.
Status	STATUS	Status of operation. The response lists contains meaningful data if and only if the status is ‘0’

7.4.26.3.3 When generated

This primitive is generated by an MIH User in response to a received MIH_Get_Information indication message. When the size of the Info Response parameters exceeds the maximum size specified in the Max Response Size parameter from MIH_Get_Information.indication primitive, one or more of the lower order list elements in Info Response parameters may be omitted.

7.4.26.3.4 Effect on receipt

The recipient will return an MIH_Get_Information response message to the designated MIIS client.

7.4.26.4 MIH_Get_Information.confirm

7.4.26.4.1 Function

This primitive is generated by the MIHF to respond to an MIH_GET_Information.request primitive.

7.4.26.4.2 Semantics of service primitive

```
MIH_Get_Information.confirm     (
                                SourceIdentifier,
                                InfoResponseBinaryDataList,
                                InfoResponseRDFDataList,
                                InfoResponseRDFSchemaURL,
                                InfoResponseRDFSchemaList,
                                Status
                                )
```

Parameters:

Name	Type	Description
Source Identifier	MIHF_ID	Shall contain the MIHF ID of the node that invoked MIH_GET_Information.response.
Info Response Binary Data List	LIST(INFO_RSP_BINARY_DATA)	(Optional) A list of TLV query responses. The list may be sorted from most preferred first to least preferred last.
Info Response RDF Data List	LIST(INFO_RSP_RDF_DATA)	(Optional) A list of RDF query responses. The list may be sorted from most preferred first to least preferred last.
Info Response RDF Schema URL	LIST(INFO_RSP_RDF_URL)	(Optional) A list of RDF Schema URL. The list may be sorted from most preferred first to least preferred last.
Info Response RDF Schema List	LIST(INFO_RSP_RDF_SCHEMA)	(Optional) A list of RDF schema query responses. The list may be sorted from most preferred first to least preferred last.
Status	STATUS	Status of operation. The response lists contains meaningful data if and only if the status is ‘0’

7.4.26.4.3 When generated

This primitive is generated by the MIHF on receiving an MIH_Get_Information Response message from a peer MIHF.

7.4.26.4.4 Effect on receipt

The MIH User that requested the information tries to interpret the Info Response parameters and takes suitable action. However, if Status does not indicate “Success”, the recipient shall ignore any other returned values and, instead, shall perform appropriate error handling.

When the size of the Info Response parameters exceeds the maximum size specified in the Max Response Size parameter from MIH_Get_Information.request primitive, one or more of the lower order list elements in Info Response parameters may be omitted.

7.5 MIH_NET_SAP primitives

7.5.1 MIH_TP_Data

The primitives associated with data transfers are as follows:

- MIH_TP_Data.request

- MIH_TP_Data.indication

- MIH_TP_Data.confirm

The MIHF uses the MIH_TP_Data.request primitive to request that an MIH PDU be transported. The transport service provider uses the MIH_TP_Data.indication primitive to indicate the arrival of an MIH PDU. MIH_TP_Data.confirm primitive is used to acknowledge the successful transfer of the MIH PDU.

7.5.1.1 MIH_TP_Data.request

7.5.1.1.1 Function

This primitive is the request for transfer of an MIH PDU.

7.5.1.1.2 Semantics

```
MIH_TP_Data.request      (
                         TransportType,
                         SourceAddress,
                         DestinationAddress,
                         ReliableDeliveryFlag,
                         MIHProtocolPDU
                         )
```

Parameters:

Name	Type	Description
Transport Type	TRANSPORT_TYPE	Identifies the protocol layer specific transport option
Source Address	TRANSPORT_ADDRESS	Protocol layer specific Transport Address of entity which has the Source MIHF
Destination Address	TRANSPORT_ADDRESS	Protocol layer specific Transport Address of entity which has the Destination MIHF
Reliable Delivery Flag	BOOLEAN	Indicate that the data shall be sent reliably and an error is generated if delivery fails. True: Reliable delivery is required. False: Reliable delivery is not required.
MIH Protocol PDU	OCTET_STRING	MIH Protocol PDU to be transferred

7.5.1.1.3 When generated

This primitive is used to request that an MIH PDU be transported to a remote MIHF.

7.5.1.1.4 Effect on receipt

The receipt of this primitive causes the selected transport service provider to attempt to transport the MIH PDU.

7.5.1.2 MIH_TP_Data.indication

7.5.1.2.1 Function

This primitive is the indication of a received MIH PDU.

7.5.1.2.2 Semantics

```
MIH_TP_Data.indication  (
                        TransportType,
                        SourceAddress,
                        DestinationAddress,
                        ReliableDeliveryFlag,
                        MIHProtocolPDU
                        )
```

Parameters:

Name	Type	Description
Transport Type	TRANSPORT_TYPE	Identifies the protocol layer specific transport option
Source Address	TRANSPORT_ADDRESS	Protocol layer specific Transport Address of entity which has the Source MIHF
Destination Address	TRANSPORT_ADDRESS	Protocol layer specific Transport Address of entity which has the Destination MIHF
Reliable Delivery Flag	BOOLEAN	Indicate that the data shall be sent reliably and an error generated if delivery fails. True: Reliable delivery is required. False: Reliable delivery is not required.
MIH Protocol PDU	OCTET_STRING	MIH Protocol PDU received.

7.5.1.2.3 When generated

This primitive is used by the transport service provider to indicate that an MIH PDU has been received from a remote MIHF.

7.5.1.2.4 Effect on receipt

The receipt of this primitive causes the MIHF to receive the MIH PDU that was transported.

7.5.1.3 MIH_TP_Data.confirm

7.5.1.3.1 Function

This primitive is used to confirm an acknowledged transfer.

7.5.1.3.2 Semantics

MIH_TP_Data.confirm (
TransportType,
SourceAddress,
DestinationAddress,
Status
)

Parameters:

Name	Type	Description
Transport Type	TRANSPORT_TYPE	Identifies the protocol layer specific transport option.
Source Address	TRANSPORT_ ADDRESS	Protocol layer specific Transport Address of entity which has the Source MIHF
Destination Address	TRANSPORT_ADDRESS	Protocol layer specific Transport Address of entity which has the Destination MIHF
Status	STATUS	Status of operation.

7.5.1.3.3 When generated

This primitive is passed from the transport service provider to the MIHF to confirm that a request to transfer an MIH PDU succeeded.

7.5.1.3.4 Effect on receipt

Upon receipt of this primitive, the receiving MIHF may stop its retransmission timer for the corresponding request. When the MIHF does not receive this primitive for a pre-defined time after transmitting an MIH_TP_Data.request with ReliableDeliveryFlag set to TRUE, the MIHF may try to retransmit the MIH_TP_Data.request.

8. Media Independent Handover Protocol

8.1 Introduction

This standard specifies an MIH Function that could be used in mobile nodes and also in entities located in the network. The MIH Functions in MN and network entities communicate with each other using the MIH protocol messages specified in this subclause. The MIH protocol allows any peer MIH Function entities to communicate with each other. The MIH protocol defines message formats for exchanging these messages between peer MIH Function entities. These messages are based on the primitives which are part of the MIH Services

The peer MIH Function entities shall complete service management procedures described in clause 6 before initiating MIH protocol procedures. The service management procedures include MIH Function discovery, MIH capability discovery and MIH registration.

8.2 MIH protocol description

The Media Independent Handover protocol provides the following services:

1) MIH Function discovery: The peer MIHF in a mobile node discovers which entity in the access network provides MIHF services.
2) MIH capability discovery: The MIHF entity in MN discovers the capabilities of a peer MIHF entity in the network. This includes a list of supported events, a list of supported commands and supported query types for the Information Service.
3) MIH registration: Peer MIHFs may register with each other to establish a new MIHF pairing.
4) MIH event subscription: MIH Function may subscribe to a particular set of remote events.
5) MIH message exchanges: These include the following:
    a) Information query: The MIHF in MN or serving PoS queries the Information Server for neighboring network information.
    b) Resource availability check: The serving PoS queries resources on the candidate target networks to check whether there are sufficient resources available for the mobile node to handover to these networks.
    c) Resource preparation: After the target network has been selected, the serving PoS requests the target PoS to prepare the necessary resources for the mobile node to handover to the selected target network.
    d) Target decision and handover execution: The MIHF entity in MN commits to do a handover to the selected target network. The handover can be commanded by the serving PoS through remote MIHF command messages. The MIHF handover commitment establishes a new layer 2 link and also any necessary higher layers with the target network.
    e) Resource release: After the handover is executed, the resources at the source network need to be released. The target PoS which is now serving the mobile node, will request the PoS on the source network to release its network resources.

8.2.1 MIH protocol acknowledgement operation

MIH protocol messages require reliability to ensure the receipt of data by the intended destination. Reliability may be provisioned with the optional acknowledgement mechanism as part of the MIH protocol. The acknowledgement mechanism is particularly useful when the underlying transport used for remote communication does not provide reliable services. When the MIH transport is reliable, the acknowledgement mechanism is optional based on carrier requirements.

The source MIHF may optionally request for an acknowledgement message (MIH ACK) to ensure successful receipt of an MIH protocol message. The MIH ACK message is used to acknowledge the successful receipt of an MIH protocol message at the destination MIHF. The source MIHF, when requesting for acknowledgement service, receives the MIH ACK message when the MIH protocol message is reliably delivered to the destination MIHF. If the MIH protocol message or the MIH ACK message is lost, the ACK timer on the source MIHF shall expire and the source MIHF shall subsequently retransmit the same MIH protocol message.

The MIH acknowledgement capability is supported by the use of two bits of information that are defined exclusively for ACK usage in the MIH header. The ACK-Req bit is set by the source MIH node and the ACK-Rsp bit is set by the destination MIH node to utilize the acknowledgement service. It is expected that the underlying transport layer would take care of ensuring the integrity of the MIH protocol message during delivery.

When seeking acknowledgement service, the source MIH node shall start a retransmission timer after sending an MIH protocol message with the ACK-Req bit set and shall save a copy of the MIH protocol message while the timer is active. The algorithm defined in IETF RFC 2988 may be used to calculate the value of the retransmission timer. If the acknowledgement message is not received before the expiration of the timer, the source MIH node may immediately retransmit the saved message with the same Message-ID and with the same Transaction-ID (with ACK-Req bit set). If the source MIH node receives the acknowledgement before the expiration of the timer on the first or any subsequent retransmitted attempt, then the source MIH node has ensured the receipt of the MIH packet and therefore, may reset the timer and release the saved copy of the MIH protocol message. During retransmission, if the source MIH node receives the ACK for any of the previous transmission attempts then the source MIH node may determine successful delivery of the message and may not have to wait for any further acknowledgements for the current message. The source MIH node shall retransmit an MIH protocol message with ACK-Req bit set until it receives an acknowledgment or the number of retransmissions reaches its maximum value. The maximum number of retransmissions can be reconfigured depending on implementation. The default value of maximum number of retransmissions is two (2). The source MIH node shall not attempt to retransmit a message with same Message-ID and Transaction-ID when the ACK-Req bit was not set in the first MIH message

When a destination MIH node receives an MIH protocol message with the ACK-Req bit set, then the destination MIH node returns an MIH ACK message with the ACK-Rsp bit set and copying the Message-ID and Transaction-ID from the received MIH protocol message. The MIH ACK message may have only the MIH header and no other payload. In instances where the destination MIH node may immediately process the received MIH protocol message and a response is immediately available, then the ACK-Rsp bit may be set in the corresponding MIH protocol response message.

The destination MIH node shall respond with an acknowledgement message for duplicate MIH messages (messages with same transaction-ID) that have the ACK-Req bit set. However, the destination MIH node shall not process these duplicate messages if it has already done so. If a destination MIH node receives an MIH protocol message with no ACK-Req bit set then no action is taken with respect to the acknowledgement functionality.

In all cases, the MIH protocol message in a transaction shall be processed only once at the destination MIH node, irrespective of the number of ACK messages received. The destination MIH node may set the ACK-Rsp bit in an MIH protocol response message and may additionally request acknowledgement by setting the ACK-Req bit for the same MIH protocol response message.

8.2.2 MIH protocol transaction state diagram

This clause describes the state diagram for an MIH transaction. A MIH transaction is identified by a sequence of messages with the same Transaction-ID submitted to, or received from, one specific remote MIHF ID.

8.2.2.1 State machines

A node that has a new available message related to a new transaction to send is called transaction source and shall start the transaction source state machine. In the same manner, a node that receives a message related to a new transaction is called transaction destination node and shall start the destination transaction state machine.

If the ACK feature is being used by the source and/or destination transaction node, the ACK-Requestor and/or ACK-Responder state machine shall be started (specific conditions specified below). The ACK related state machine shall be run in parallel to the transaction source/destination state machines. The conditions and transitions in the state machines running in one MIHF can be affected by the same global variables. There are no cases where two or more state machines running at the same time can write the value of the same variables.

Figure 22 illustrates the interaction of transaction source/destination state machines with the ACK related state machines.

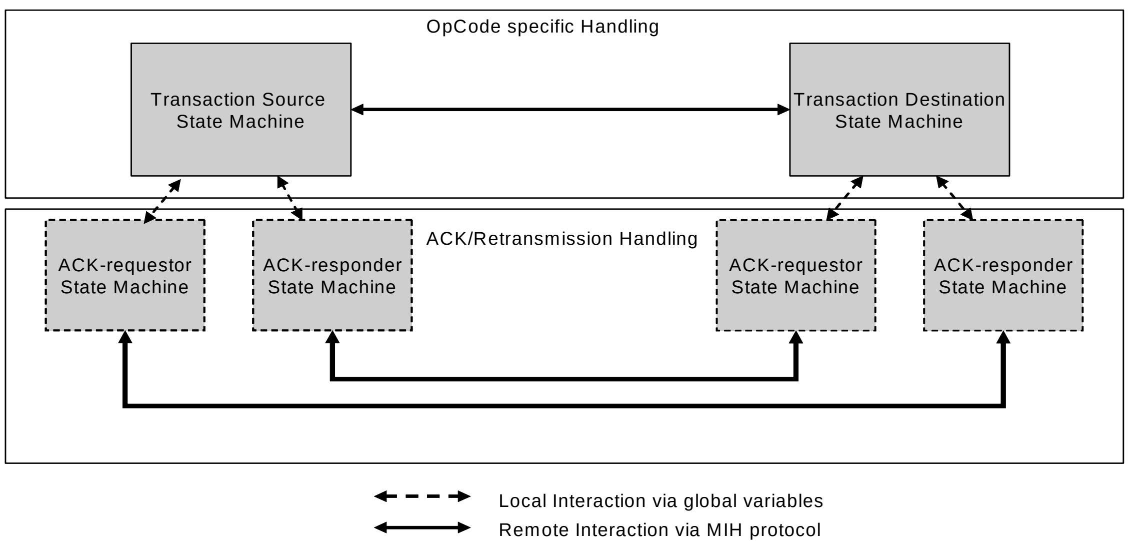


Figure 22—State machines interactions

8.2.2.2 Notational conventions used in state diagrams

State diagrams are used to represent the operation of an MIH transaction as a group of connected, mutually exclusive states. Only one state of each state machine can be active at any given time.

Each state is represented in the state diagram as a rectangular box, divided into two parts by a horizontal line. The upper part contains the state identifier, written in uppercase letters. The lower part contains any procedures that are executed on entry to the state.

All permissible transitions between states are represented by arrows, the arrowhead denoting the direction of the possible transition. Labels attached to arrows denote the condition(s) that shall be met in order for the transition to take place.

A transition that is global in nature (i.e., a transition that occurs from any of the possible states if the condition attached to the arrow is met) is denoted by an open arrow; i.e., no specific state is identified as the origin of the transition.

On entry to a state, the procedures defined for the state (if any) are executed exactly once, in the order that they appear on the page. Each action is deemed to be atomic; i.e., execution of a procedure completes before the next sequential procedure starts to execute. No procedures execute outside of a state block. On completion of all of the procedures within a state, all exit conditions for the state (including all conditions associated with global transitions) are evaluated continuously until such a time as one of the conditions is met. All exit conditions are regarded as Boolean expressions that evaluate to TRUE or FALSE; if a condition evaluates to True, then the condition is met.

The label UCT denotes an unconditional transition (i.e., UCT always evaluates to TRUE).

A variable that is set to a particular value in a state block retains this value until a subsequent state block executes a procedure that modifies the value.

Should a conflict exist between the interpretation of a state diagram and either the corresponding global transition tables or the textual description associated with the state machine, the state diagram takes precedence.

The interpretation of the special symbols and operators used in the state diagrams is defined in Table 13; these symbols and operators are derived from the notation of the “C” programming language, ANSI X3.159.

Table 13—State machine symbols

Symbol	Interpretation
( )	Used to force the precedence of operators in Boolean expressions and to delimit the argument(s) of actions within state boxes.
;	Used as a terminating delimiter for actions within state boxes. Where a state box contains multiple actions, the order of execution follows the normal English language conventions for reading text.
=	Assignment action. The value of the expression to the right of the operator is assigned to the variable to the left of the operator. Where this operator is used to define multiple assignments, (e.g. a = b = X) the action causes the value of the expression following the right-most assignment operator to be assigned to all of the variables that appear to the left of the right-most assignment operator.
!	Logical NOT operator.
&&	Logical AND operator.
||	Logical OR operator.
if...then...	Conditional action. If the Boolean expression following the **if** evaluates to TRUE, then the action following the **then** is executed.
{statement 1, ... statement N}	Compound statement. Braces are used to group statements that are executed together as if they were a single statement.
!=	Inequality. Evaluates to TRUE if the expression to the left of the operator is not equal in value to the expression to the right.

Table 13—State machine symbols

Symbol	Interpretation
==	Equality. Evaluates to TRUE if the expression to the left of the operator is equal in value to the expression to the right.
<	Less than. Evaluates to TRUE if the value of the expression to the left of the operator is less than the value of the expression to the right.
>	Greater than. Evaluates to TRUE if the value of the expression to the left of the operator is greater than the value of the expression to the right.
>=	Greater than or equal to. Evaluates to TRUE if the value of the expression to the left of the operator is either greater than or equal to the value of the expression to the right.
+	Arithmetic addition operator.
-	Arithmetic subtraction operator.

8.2.2.3 Global variables

Global variables are available for use by more than one state machine and are used to perform interstate-machine communication and initialization functions.

Exported variables are global variables that are also readable and writable from entities external to the state machines. The global and external variables are specified in Tables 14 and 15, respectively.

Table 14—Global state machine variables

Name	Type	Description
Opcode	OPCODE	An Opcode.
TID	TID	A transaction identifier.
MID	MID	A message identifier.
StartACKRequestor	BOOLEAN	When its value turns to TRUE the ACK-requestor state machines is started. While its value is TRUE it indicates that the ACK requestor state machine is running.
StartACKResponder	BOOLEAN	When its value turns to TRUE the ACK-responder state machines is started. While its value is TRUE it indicates that the ACK responder state machine is running.
AckFailure	BOOLEAN	Indicates if ACK operation failed.
TransactionStopWhen	UNSIGNED_INT(1)	A timer to stop the transaction.
RetransmissionWhen	UNSIGNED_INT(1)	A timer to retransmit a message.
DelayedAckWhen	UNSIGNED_INT(1)	A timer to send a delayed acknowledgment.

8.2.2.4 Global procedures

a) **void Process(MIH_MESSAGE)** - This procedure processes the incoming message passed as an input variable.

b) **void Transmit(MIH_MESSAGE)** - This procedure transmits the message passed as the input variable.

c) **BOOLEAN IsBroadcastMsg(MIH_MESSAGE)**- This procedure outputs TRUE if the input message has a broadcast destination MIHF ID. Otherwise outputs FALSE.

d) **MIHF_ID SrcMIHF_ID(MIH_MESSAGE)** - This procedure obtains a Source Identifier TLV from the message passed as the input and returns the value of the TLV.

e) **MIHF_ID DstMIHF_ID(MIH_MESSAGE)** - This procedure obtains a Destination Identifier TLV from the message passed as the input and returns the value of the TLV.

8.2.2.5 Global constants

a) **TransactionLifetime** - The maximum time from the initiation of a transaction until its termination.

b) **Request** - An OPCODE value of 0x1.

c) **Response** - An OPCODE value of 0x2.

d) **Indication** - An OPCODE value of 0x3.

8.2.2.6 Timers

The timers defined for these state machines are decremented, if their value is non-zero, by the operation of Transaction Timers state machine. All timers have a resolution of one second, i.e., the initial values used to start the timers are integer values, and they represent the timer period as an integral number of seconds.

8.2.2.6.1 Local variables and constants

Tick - This variable is set in response to a regular one-second tick generated by an external system clock function. Whenever the system clock generates a one-second tick, the tick variable is set to TRUE. The variable is set to FALSE by the operation of the state machine. The operation of the system clock functions is not otherwise specified by the standard.

void dec(Timer) - This procedure decrements the timer only if its value is greater than 0.

8.2.2.6.2 Transaction Timers state machine

The Transaction Timers state machine for a given transaction is responsible for decrementing the timer variables for this transaction each second, in response to an external system clock function. The timer variables are used, and set to their initial values, by the operation of the individual state machines for the transaction.

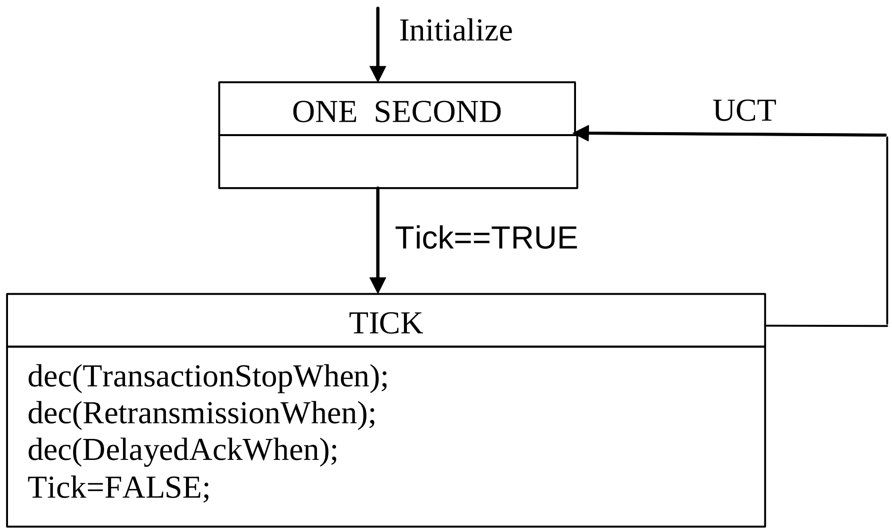


Figure 23—Transaction Timers state machine

8.2.2.7 Transaction source and destination state machines

8.2.2.7.1 Local variables

IsBroadcast - This variable's type is Boolean. When its value is TRUE it indicates that a message has a broadcast destination MIHF ID. Otherwise, its value is FALSE.

8.2.2.7.2 Local procedures

TID NewTID(void) - This procedure generates a new transaction ID for the transaction generated by the new available message.

UNSIGNED_INT(1) GetAckDelay(MID) - This procedure computes the time duration for delaying an ACK using MID of a message. Especially the output value is always zero (0) if MID.OPCODE==Indication or MID.OPCODE==Response. If MID.OPCODE==Request, the time duration for delaying the ACK may depend on MID.SID and MID.AID.

8.2.2.7.3 Transaction source state machine

The transaction source state machine shall be started, and related transaction initiated, when a message related to a new transaction is available to be sent (MsgOutAvail is TRUE). The transaction terminates when it transits to the SUCCESS state and any ACK related state machines if started were terminated; or if it transits to the FAILURE state.

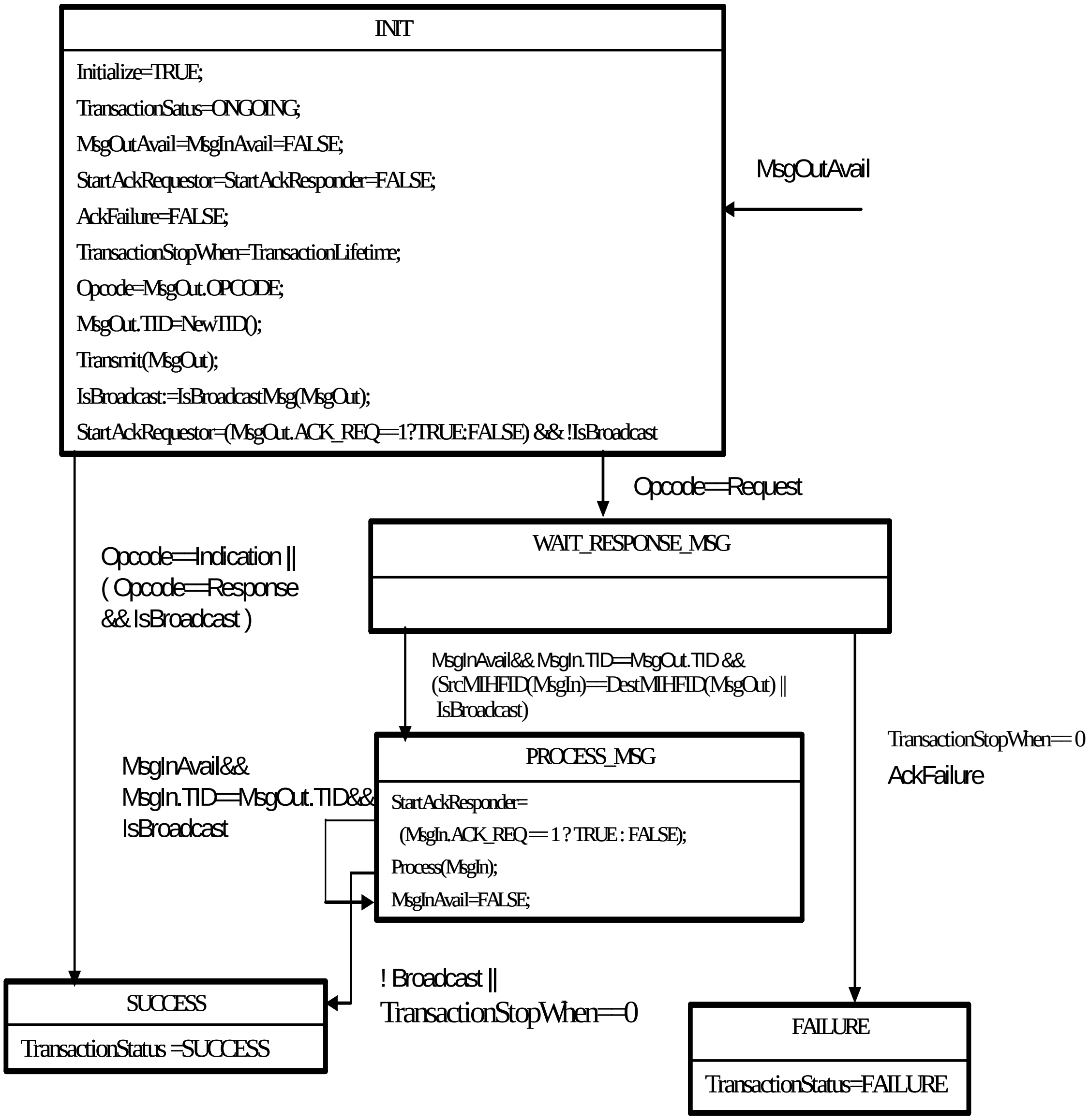


Figure 24—Transaction source state machine

8.2.2.7.4 Transaction destination state machine

The transaction destination state machine shall be started, and related transaction initiated, when a message related to a new transaction is received (MsgInAvail is TRUE).

The transaction terminates when it transits to the FAILURE state or SUCCESS state and any ACK related state machines, if started, were terminated.

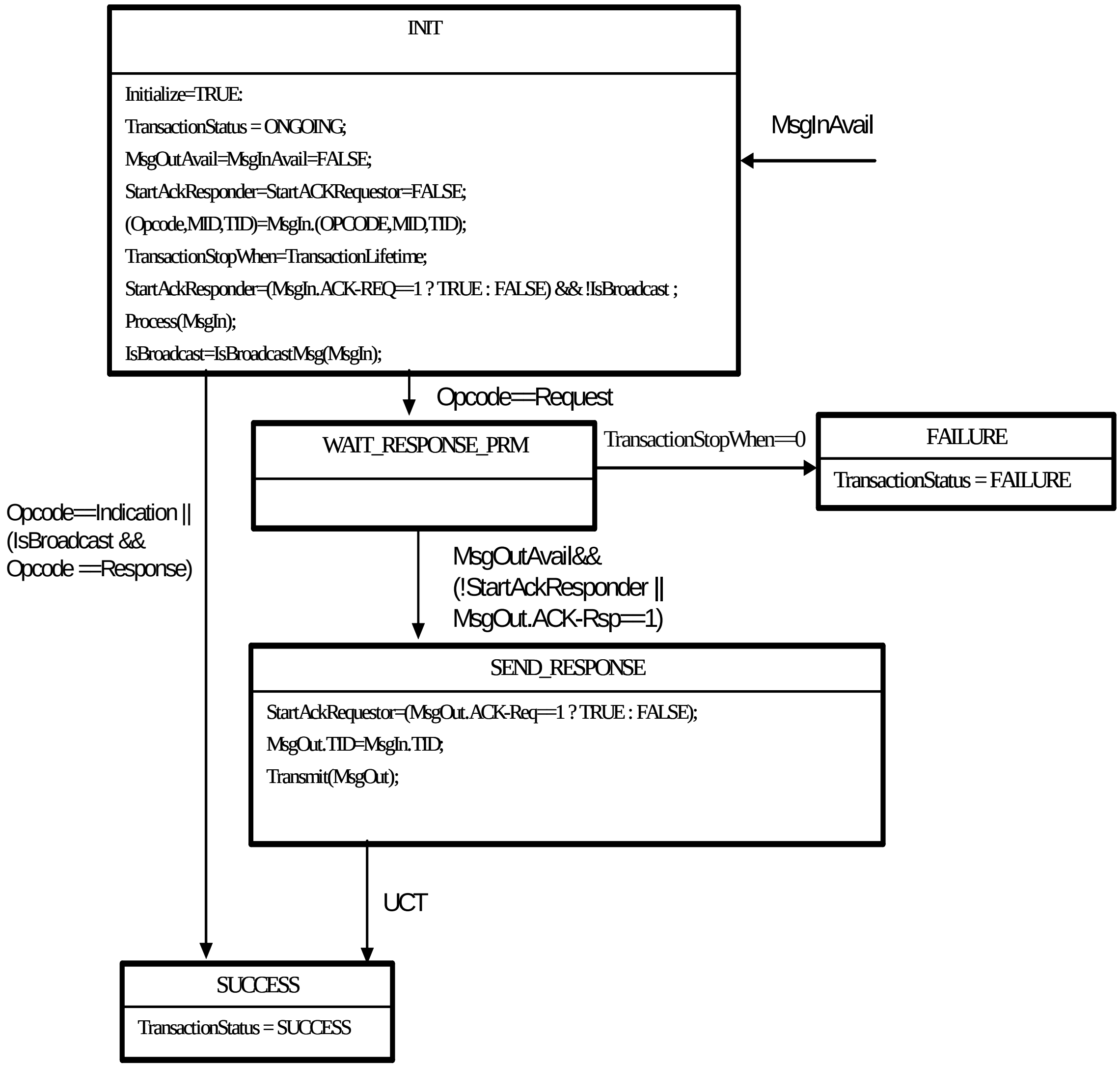


Figure 25—Transaction destination state machine

8.2.2.8 ACK related state machines

The ACK-requestor state machine is started when the StartAckRequest variable turns TRUE and ACK-responder state machine is started when StartAckResponder variable turns TRUE.

8.2.2.8.1 Local variables

DUP - This variable is of type MIH_MESSAGE and represents an MIH message which has already been sent. This variable is used within ACK Responder state machine.

ACK - This variable is of type MIH_MESSAGE and represents an MIH message with the ACK-Rsp bit set and the same message ID and transaction ID as the MIH message it acknowledges. This variable is used within ACK Responder state machine.

RtxCtr - This variable is of type UNSIGNED_INT(1) and represents a number of retransmissions of a specific message. This variable is used within ACK Requestor state machine.

AckDelayTime - This variable is of type UNSIGNED_INT(1). It represents the maximum time that an MIH request with Ack-Req set receiver should wait for the generation of an MIH response before sending a ACK message. If the response is available to be sent before AckDelayTime, the Acknowledgement will be piggy-back in the response message and there is no need to send ACK message.

8.2.2.8.2 Local constants

RetransmissionInterval - The time interval between two subsequent transmissions of a specific message.

RtxCtrMax - The maximum number of times that a message will be retransmitted, if retransmission conditions occur.

The maximum number of retransmissions and the retransmission interval may depend on the characteristics of the underlying transport. These shall be defined as configuration parameters of an implementation, possibly in a future defined MIB.

Note that the maximum number of retransmission is bounded by the transaction lifetime.

8.2.2.8.3 ACK requestor state machine

The ACK requestor state machine shall be started, when the StartAckRequestor variable turns to TRUE in a source or destination transaction state machine. This state machine uses the global variables set by the originating state machine. This state machine terminates when it transits to the FAILURE state or SUCCESS state.

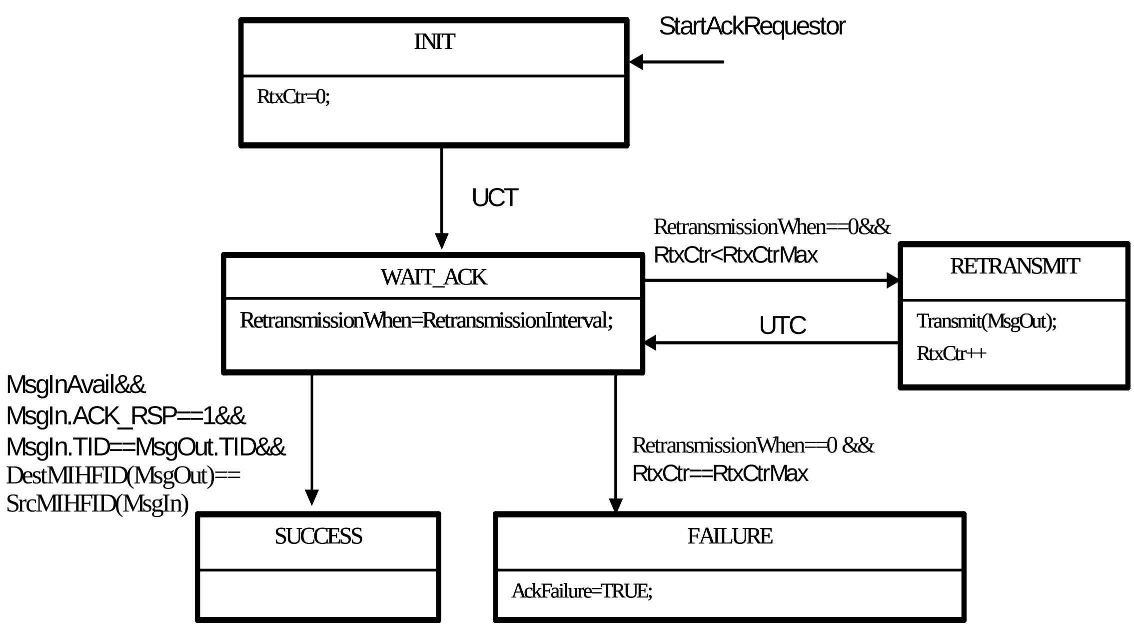


Figure 26—ACK requestor state machine

8.2.2.8.4 ACK responder state machine

The ACK responder state machine shall be started, when the StartAckResponder variable turns to TRUE in a source or destination transaction state machine. This state machine uses the global variables set by the originating state machine. This state machine terminates when it transits to SUCCESS state.

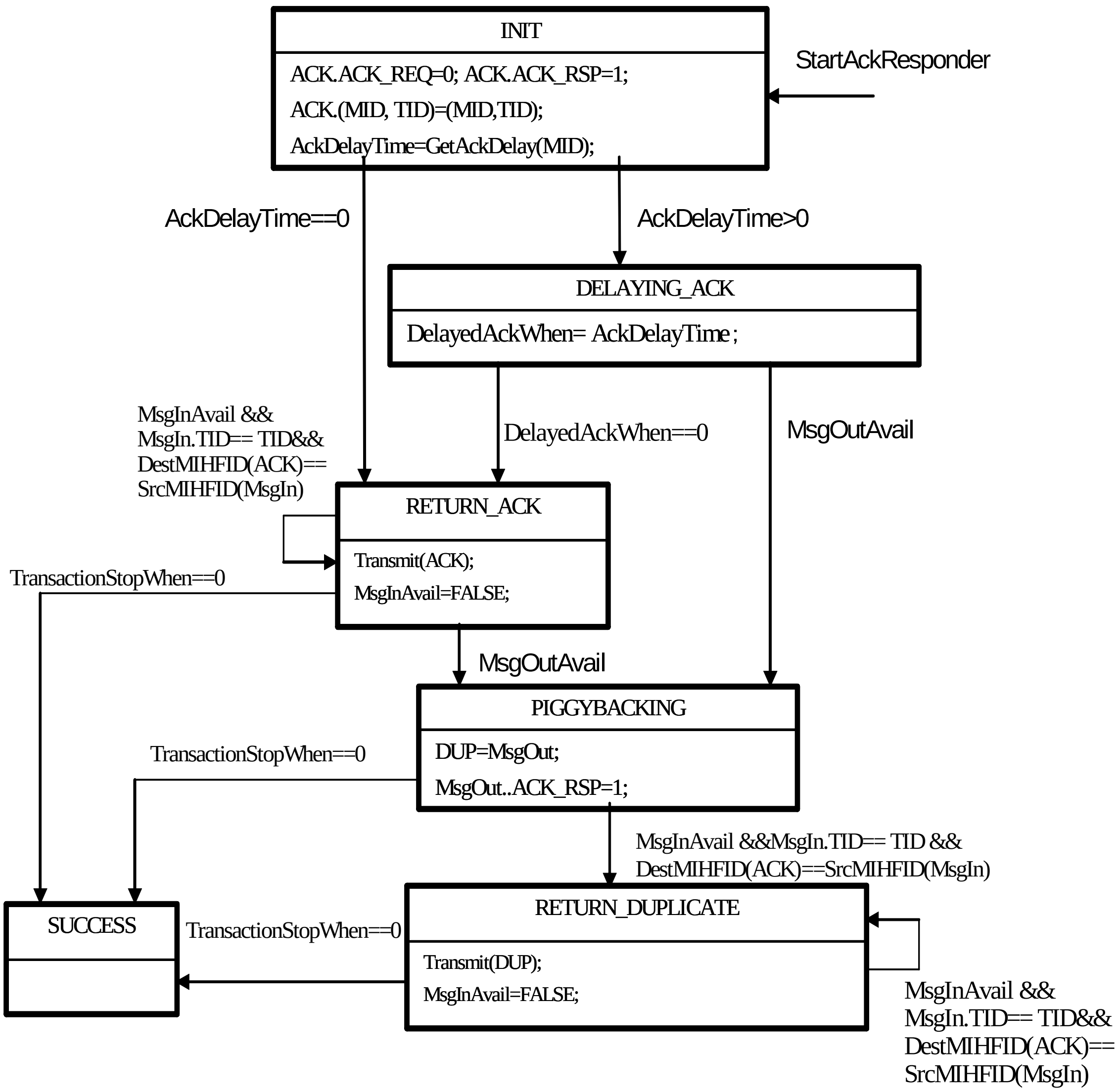


Figure 27—ACK responder state machine

8.2.3 Other considerations

8.2.3.1 Congestion control and load management

The MIH protocol does not provide direct support for congestion control. Therefore, it is recommended to run the MIH protocol over congestion aware transport layers.

In order to help prevent congestion, flow control mechanisms are implemented at the MIHF. A single rate limiter applies to all traffic (for all interfaces and message types). It applies to retransmissions, as well as new messages, although an implementation may choose to prioritize one over the other. When the rate limiter is in effect, MIH messages are queued until transmission is re-enabled, or an error condition may be indicated back to local upper layer applications. The rate limiting mechanism is implementation specific, but it is recommended that a token bucket limiter as described in IETF RFC 4443 be used.

When an MIHF suffers from overload, it may drop requests from MIH requestors, and may differentiate between requestors to implement selective dropping. For example, messages could be dropped from a particular requestor if that requestor could be established as the origin of a denial of service attack. An MIHF may not drop a message if it was delivered reliably by the transport (L2 or L3). Any reliable delivery function may be able to indicate a flow control back to the requestor, and an MIHF may invoke flow control towards a specific requestor when overloaded with reliably delivered messages.

8.2.3.2 Reliability

MIH protocol messages are delivered via media dependent transport. The MIH Function determines the transport reliability provided to the MIH user. However, the MIH user may provide its preference to the MIH Function. How the preference is communicated to the MIHF depends on the implementation and is outside the scope of this standard. If the MIHF uses unreliable transport and a reliable message delivery is required, the MIH protocol ACK operation shall be applied for transmitting the message. If the MIHF uses reliable transport, then the MIH protocol ACK operation should not be required for transmitting the message.

8.2.3.3 Fragmentation

MIH Protocol does not provide fragmentation and reassembly services.

8.2.3.4 MIHF discovery

8.2.3.4.1 General

The MIHF discovery refers to the procedure that allows one MIHF to discover its peer MIHFs (e.g., an MN discovers available peer MIHFs in an access network). MIHF discovery can be done either at layer 2 or layer 3. However, MIHF discovery at layer 3 and MIHF discovery performed over control plane using media specific broadcast control message are outside the scope of this standard.

8.2.3.4.2 Combined MIH function discovery and capability discovery over data plane

Combined MIH function discovery and capability discovery is performed to discover MIHF ID, the peer MIHF transport address, and MIHF capabilities at the same time. As stated in 6.2.3, MIHF Discovery can be implicitly performed using the MIH Capability Discovery when both MIH nodes are residing in the same broadcast domain. If MIHF ID and transport address are known, e.g., pre-configured, MIHF may use MIH_Capability_Discover messages to discover MIHF capabilities only. The following subclauses refer to the MIH Capability Discovery both as a means to discover the MIHF and its capabilities.

<u>Unsolicited MIH capability discovery</u>

An MIHF may discover its peer MIHF entities and their capabilities either by listening to media-specific broadcast messages or media independent MIH capability broadcast message.

For example, by listening to a media-specific broadcast message such as a beacon in IEEE 802.11 or a DCD in IEEE 802.16, link layers on an MN forward the received message to its MIHF. An MIHF may receive MIH_Capability_Discover response message broadcasted over data plane periodically from an MIH entity.

<u>Solicited MIH capability discovery</u>

An MIHF (the requestor) may choose to discover its peer MIH functions and their capability by broadcasting or unicasting an MIH_Capability_Discover request message to either its broadcast domain or a known MIHF ID and address, respectively. Only MIH network entities shall respond to a broadcasted MIH_Capability_Discover request.

When a peer MIH function (the responder) receives the MIH_Capability_Discover request message, it shall unicast an MIH_Capability_Discover response message back to the requestor. The response shall be sent using the same transport type over which the request message was received. When the requestor receives the unicast MIH_Capability_Discover response message, it learns the responder's MIHF ID by checking the source ID of MIH_Capability_Discover response.

For complete operation, the requestor may set a timer at the time of sending an MIH_Capability_Discover request during which time the requestor is in waiting state for a response from the responder. When the response message is received while the timer is running, the requestor stops the timer and finishes the MIH function and capability discovery procedure. When the timer expires without receiving a response message, the requestor may try the combined MIH function discovery and capability discovery procedure using a different transport than previously or terminate the MIH function and capability discovery procedure.

8.3 MIH protocol identifiers

Following identifiers are used in MIH protocol messages:

- MIHF ID
- Transaction ID

8.3.1 MIHF ID

MIHF Identifier (MIHF ID) is an identifier that is required to uniquely identify an MIHF entity for delivering the MIH services. MIHF ID is used in all MIH protocol messages. This enables the MIH protocol to be transport agnostic.

MIHF ID may be assigned to the MIHF during its configuration process. The configuration process is outside the scope of the standard. For example, MIHF ID may be an FQDN or NAI of the sender of the registration request.

Broadcast MIHF ID is defined as an MIHF ID of zero length. A broadcast MIHF ID can be used when destination MIHF ID is not known to a sending MIHF. When MIH protocol message with broadcast MIHF ID is transmitted over data plane, the MIH protocol message shall be broadcasted over either L2 or L3 data plane.

The MIHF ID is of type MIHF_ID (See Annex B.3.11).

8.3.2 Transaction ID

Transaction Identifier (Transaction ID) is an identifier that is used to match a request message with its corresponding response message. This identifier is also required to match each request, response or indication message and its corresponding acknowledgment. This identifier shall be created at the node initiating the transaction and it is carried over within the fixed header part of the MIH protocol frame. In messages where Transaction ID is not used it is set to 0.

Transaction ID is defined as a 16 bit long unsigned integer whose value is unique among all the pending transactions between a given pair of the sender and receiver. For example, this could be an integer that starts from a random initial value and incremented by one (modulo 2^16) every time a new Transaction ID is generated.

8.4 MIH protocol frame format

8.4.1 General frame format

In MIH protocol messages, all TLV definitions are always aligned on an octet boundary and hence no padding is required.

Figure 28 shows the components of the MIH protocol frame.

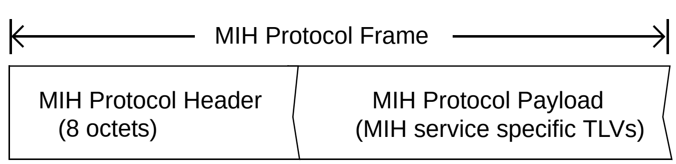


Figure 28 — MIH protocol general frame format

8.4.1.1 MIH protocol header fields

The MIH protocol header carries the essential information which is present in every frame and is important for parsing and analyzing the MIH protocol frame.

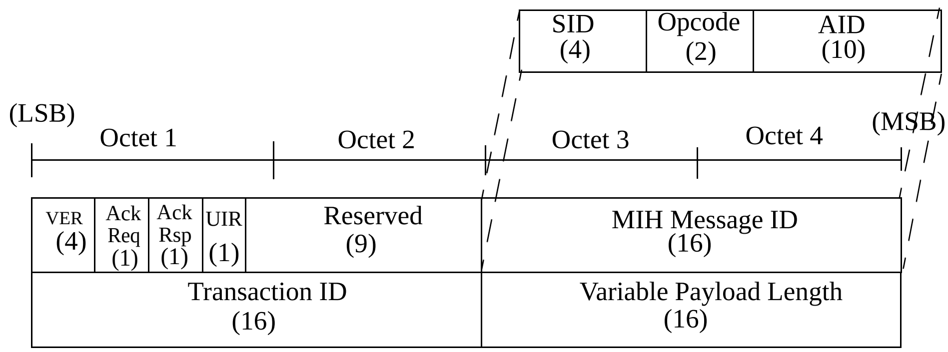


Figure 29 — MIH protocol header format

Table 15 shows the description of the header fields.

Table 15—Description of MIH protocol header fields

Field name	Size (bits)	Description
Version	4	This field is used to specify the version of MIH protocol used. 0: Not to be used 1: First version 2 - 15: (*Reserved*) The version number will be incremented only when a fundamental incompatibility exists between a new revision and the prior edition of the standard. An MIH node that receives an MIH message with a higher version number than it supports will discard the frame without indication to the sending MIH node.
ACK-Req	1	This field is used for requesting an acknowledgement for the message.
ACK-Rsp	1	This field is used for responding to the request for an acknowledgement for the message.
Unauthenticated Information Request (UIR)	1	This field is used by the MIH Information Service to indicate if the protocol message is sent in pre-authentication/pre-association state so that the length of the response message can be limited. The UIR bit should be set by the originator when making an MIH information service request over a certain link in the un-associated/unauthenticated or unregistered state. Network entities along the communication path to the information server may verify the setting of UIR bit and modify if necessary. However, the MIH Information Server may not have means to verify the authenticity of the UIR bit.
Reserved	9	This field is intentionally kept reserved. When not used, all the bits of this field are to be set to '0'.

Table 15—Description of MIH protocol header fields

Field name	Size (bits)	Description
MIH Message ID (MID)	16	Combination of the following 3 fields.
-- Service Identifier (SID)	4	Identifies the different MIH services, possible values are: 1: Service Management 2: Event Service 3: Command Service 4: Information Service
-- Operation Code (Opcode)	2	Type of operation to be performed with respect to the SID, possible values are: 1: Request 2: Response 3: Indication
-- Action Identifier (AID)	10	This indicates the action to be taken with regard to the SID (see Table C-1 for AID assignments).
Transaction ID	16	This field is used for matching Request and Response, as well as matching Request, Response and Indication to an ACK.
Variable Load Length	16	Indicates the total length of the variable load embedded in this MIH protocol frame. The length of the MIH protocol header is NOT included.

8.5 Message parameter TLV encoding

The following TLV encoding shall be used for all parameters in an MIH protocol message.

Type (1 octet)	Length (variable octets)	Value (variable octets)
Type of this parameter.	Length of the *Value* field of this parameter.	Value of this parameter.

Specifically, the *Type* field shall be one octet[2], and the *Length* shall be encoded with the rules described in subclause 6.5.6.2.

Moreover, TLV *Type* values shall be unique within the MIH protocol. The TLV encoding starts at 1 and any subsequent values are assigned in ascending order.

8.6 MIH protocol messages

This subclause specifies different MIH protocol messages in TLV form. The shaded areas represent the MIH protocol header, while the unshaded areas represent the MIH protocol payload. The payload consists of a set of identifiers in TLV form.

The TLV Type assignment for each TLV can be found in Annex C.

[2]Note that the TLV *Type* field length is different than the Information Element *Type* length, which is 4 octets.

TLV type values ranging from 110 to 127 are reserved for experimental TLVs. These values may be used by different implementations to evaluate the option of using TLVs not defined by the specification.

When a TLV type value is in the range of experimental TLVs and the data type of the TLV value is unknown or the TLV value is not in the range of valid values, the TLV should be ignored and the rest of the message should be processed. Also, experimental TLVs can be ignored, based on the MIHF information that is communicating with another MIHF with different experimental TLVs implementation.

All MIH messages carry a source MIHF ID followed by a destination MIHF ID as the first two TLVs of the MIH protocol payload part of the message. Broadcast MIHF ID is allowed in an MIH_Capability_Discover request and response message as its destination MIHF ID.

8.6.1 MIH messages for service management

8.6.1.1 MIH_Capability_Discover request

The corresponding MIH primitive of this message is defined in 7.4.1.1.

If a requesting MIHF entity does not know the destination MIHF entity’s MIHF ID, the requesting MIHF entity fills its destination MIHF ID with broadcast MIHF ID and broadcasts this message over data plane, either L2 or L3. If a requesting MIHF entity knows the destination MIHF entity’s MIHF ID, the requesting MIHF entity fills its destination MIHF ID and unicast this message over data plane, either L2 or L3.

If the generation of this message is invoked upon receiving MIH capability advertisement in unauthenticated state through media specific broadcast message, such as Beacon and DCD, destination MIHF ID is filled with broadcast MIHF ID and this message is transmitted over control plane using a L2 management frame, such as a 802.11 management action frame or a 802.16 MIH MAC management message.

This message may contain the SupportedMihEventList, SupportedMihCommandList, SupportedISQueryTypeList and SupportedTransportList TLVs to enable the receiving MIHF to discover the sending MIHF's capability. Therefore, peer MIHF entities can discover each other's MIH capability by one MIH protocol message transaction.

MIH Header Fields (SID=1, Opcode=1, AID=1)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SupportedMihEventList (optional) (MIH Event List TLV)
SupportedMihCommandList (optional) (MIH Command List TLV)
SupportedISQueryTypeList (optional) (MIIS Query Type List TLV)
SupportedTransportList (optional) (Transport Option List TLV)

8.6.1.2 MIH_Capability_Discover response

The corresponding MIH primitive of this message is defined in 7.4.1.3.

Only an MIHF capable network entity shall respond with an MIH_Capability_Discover response to the received MIH_Capability_Discover request with a broadcast MIHF ID, or send unsolicited MIH_Capability_Discover responses periodically. When an MIH network entity broadcasts an unsolicited MIH_Capability_Discover response to advertise its MIHF ID and capabilities, Destination ID shall be a broadcast MIHF ID.

MIH Header Fields (SID=1, Opcode=2, AID=1)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkMACs (optional) (Link MACs TLV)
SupportedMihEventList (optional) (MIH Event List TLV)
SupportedMihCommandList (optional) (MIH Command List TLV)
SupportedISQueryTypeList (optional) (MIIS Query Type List TLV)
SupportedTransportList (optional) (Transport Option List TLV)
MBBHandoverSupport (optional) (MBB Handover Support TLV)
Status (Status TLV)

8.6.1.3 MIH_Register request

The corresponding MIH primitive of this message is defined in 7.4.2.1.

This message is transmitted to the remote MIHF to perform a registration or re-registration.

MIH Header Fields (SID=1, Opcode=1, AID=2)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifierList (Link Identifier List TLV)
RequestCode (Register Request Code TLV)

8.6.1.4 MIH_Register response

The corresponding MIH primitive of this message is defined in 7.4.2.3.

This message is sent in response to a registration or re-registration request.

MIH Header Fields (SID=1, Opcode=2, AID=2)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
ValidTimeInterval (not included if Status does not indicate “Success”) (Valid Time Interval TLV)
Status (Status TLV)

8.6.1.5 MIH_DeRegister request

The corresponding MIH primitive of this message is defined in 7.4.3.1.

This message is transmitted to the remote MIHF to request a de-registration. There is no parameter for this message.

MIH Header Fields (SID=1, Opcode=1, AID=3)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)

8.6.1.6 MIH_DeRegister response

The corresponding MIH primitive of this message is defined in 7.4.3.3.

This message is sent in response to a de-registration request.

MIH Header Fields (SID=1, Opcode=2, AID=3)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
Status (Status TLV)

8.6.1.7 MIH_Event_Subscribe request

The corresponding MIH primitive of this message is defined in 7.4.4.1.

This message is sent by a remote MIHF (the subscriber) to subscribe to one or more event types from a particular event origination point.

MIH Header Fields (SID=1, Opcode=1, AID=4)

Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
RequestedMihEventList (MIH Event List TLV)
EventConfigurationInfoList (Event Configuration Info List TLV)

8.6.1.8 MIH_Event_Subscribe response

The corresponding MIH primitive of this message is defined in 7.4.4.2.

The response indicates which of the event types were successfully subscribed.

MIH Header Fields (SID=1, Opcode=2, AID=4)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
ResponseMihEventList (not included if Status does not indicate “Success”) (MIH Event List TLV)
Status (Status TLV)

8.6.1.9 MIH_Event_Unsubscribe request

The corresponding MIH primitive of this message is defined in 7.4.5.1.

This message is sent by a remote MIHF (the subscriber) to unsubscribe from a set of link layer events.

MIH Header Fields (SID=1, Opcode=1, AID=5)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
RequestedMihEventList (MIH Event List TLV)

8.6.1.10 MIH_Event_Unsubscribe response

The corresponding MIH primitive of this message is defined in 7.4.5.2.

The response indicates which of the event types were successfully unsubscribed.

MIH Header Fields (SID=1, Opcode=2, AID=5)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
ResponseMihEventList (not included if Status does not indicate “Success”) (MIH Event List TLV)
Status (Status TLV)

8.6.2 MIH messages for event service

8.6.2.1 MIH_Link_Up indication

The corresponding MIH primitive of this message is defined in 7.4.6.1.

This notification is delivered from an MIHF, when present in the PoA, to an MIHF in the network when a layer 2 connection is successfully established with a mobile node.

MIH Header Fields (SID=2, Opcode=3, AID=1)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
OldAccessRouter (optional) (Old Access Router TLV)
NewAccessRouter (optional) (New Access Router TLV)
IPRenewalFlag (optional) (IP Renewal Flag TLV)
MobilityManagementSupport (optional) (Mobility Management Support TLV)

8.6.2.2 MIH_Link_Down indication

The corresponding MIH primitive of this message is defined in 7.4.7.1.

This notification is delivered from an MIHF, when present in the PoA, to an MIHF in the network when a layer 2 connection with a mobile node is disconnected due to a certain reason.

MIH Header Fields (SID=2, Opcode=3, AID=2)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
OldAccessRouter (optional) (Old Access Router TLV)
ReasonCode (Link Down Reason Code TLV)

8.6.2.3 MIH_Link_Going_Down indication

The corresponding MIH primitive of this message is defined in 7.4.8.1.

This message is transmitted to the remote MIHF when a layer 2 connectivity is expected (predicted) to go down within a certain time interval.

MIH Header Fields (SID=2, Opcode=3, AID=3)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
TimeInterval (Time Interval TLV)
LinkGoingDownReason (Link Going Down Reason TLV)
UniqueEventIdentifier (Unique Event Identifier TLV)

8.6.2.4 MIH_Link_Detected indication

The corresponding MIH primitive of this message is defined in 7.4.9.1.

This message is transmitted to the remote MIHF when a new link has been detected.

MIH Header Fields (SID=2, Opcode=3, AID=5)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)

LinkIdentifier (Link Identifier TLV)
MIHCapability (MIH Capability Flag TLV)

8.6.2.5 MIH_Link_Parameters_Report indication

The corresponding MIH primitive of this message is defined in 7.4.10.1.

This message indicates changes in link parameters that have crossed pre-configured threshold levels.

MIH Header Fields (SID=2, Opcode=3, AID=6)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
LinkParameterReportList (Link Parameter Report List TLV)

8.6.2.6 MIH_Link_Event_Rollback indication

The corresponding MIH primitive of this message is defined in 7.4.11.1.

This message is used in conjunction with Link_Going_Down. If the link is no longer expected to go down in the specified time interval, a Link_Event_Rollback message is sent to the remote Event destination.

MIH Header Fields (SID=2, Opcode=3, AID=4)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
UniqueEventIdentifier (Unique Event Identifier TLV)

8.6.2.7 MIH_Link_Handover_Imminent indication

The corresponding MIH primitive of this message is defined in 7.4.13.1.

This message indicates that a link layer handover decision has been made and its execution is imminent.

MIH Header Fields (SID=2, Opcode=3, AID=7)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)

Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
OldLinkIdentifier (Link Identifier TLV)
NewLinkIdentifier (New Link Identifier TLV)
HandoverType (Handover Type TLV)
MacOldAccessRouter (optional) (Old Access Router TLV)
MacNewAccessRouter (optional) (New Access Router TLV)

8.6.2.8 MIH_Link_Handover_Complete indication

The corresponding MIH primitive of this message is defined in 7.4.14.1.

This message indicates that a link layer handover has been completed.

MIH Header Fields (SID=2, Opcode=3, AID=8)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
OldLinkIdentifier (Link Identifier TLV)
NewLinkIdentifier (New Link Identifier TLV)
MacOldAccessRouter (optional) (Old Access Router TLV)
MacNewAccessRouter (optional) (New Access Router TLV)
LinkHandoverStatus (Status TLV)

8.6.3 MIH messages for command service

8.6.3.1 MIH_Get_Link_Parameters request

The corresponding MIH primitive of this message is defined in 7.4.15.1.

This message is used to discover the status of currently available links.

MIH Header Fields (SID=3, Opcode=1, AID=1)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)

Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
DeviceStatesRequest (optional) (Device Status Request TLV)
LinkIdentifierList (Link Identifier List TLV)
GetStatusRequestSet (Get Status Request Set TLV)

8.6.3.2 MIH_Get_Link_Parameters response

The corresponding MIH primitive of this message is defined in 7.4.15.2.

This message is used by an MIHF to report the status of currently available links.

MIH Header Fields (SID=3, Opcode=2, AID=1)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
DeviceStatesResponseList (optional) (not included if Status does not indicate “Success”) (Device States Response TLV)
GetStatusResponseList (not included if Status does not indicate “Success”) (Get Status Response List TLV)
Status (Status TLV)

8.6.3.3 MIH_Link_Configure_Thresholds request

The corresponding MIH primitive of this message is defined in 7.4.16.1.

This message is used to configure thresholds of the lower layer link.

MIH Header Fields (SID=3, Opcode=1, AID=2)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
ConfigureRequestList (Configure Request List TLV)

8.6.3.4 MIH_Link_Configure_Thresholds response

The corresponding MIH primitive of this message is defined in 7.4.16.2.

This message returns the status of a thresholds configuration request.

MIH Header Fields (SID=3, Opcode=2, AID=2)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkIdentifier (Link Identifier TLV)
ConfigureResponseList (not included if Status does not indicate “Success”) (Configure Response List TLV)
Status (Status TLV)

8.6.3.5 MIH_Link_Actions request

The corresponding MIH primitive of this message is defined in 7.4.17.1.

This message is used to control the behavior of a set of lower layer links.

MIH Header Fields (SID=3, Opcode=1, AID=3)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkActionList (Link Action List TLV)

8.6.3.6 MIH_Link_Actions response

The corresponding MIH primitive of this message is defined in 7.4.17.2.

This message returns the result of an MIH_Link_Actions request.

MIH Header Fields (SID=3, Opcode=2, AID=3)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkActionsResponse (not included if Status does not indicate “Success”) (Link Actions Response TLV)
Status (Status TLV)

8.6.3.7 MIH_Net_HO_Candidate_Query request

The corresponding MIH primitive of this message is defined in 7.4.18.2.

This message is used for communication between the MIHF on an MN and the MIHF on a network. The function is used to communicate an intent of network initiated handover.

MIH Header Fields (SID=3, Opcode=1, AID=5)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SuggestedNewLinkList (List of Link PoA List TLV)
QueryResourceReportFlag (Query Resource Report Flag TLV)

8.6.3.8 MIH_Net_HO_Candidate_Query response

The corresponding MIH primitive of this message is defined in 7.4.18.4.

This message is used for communication between the MIHF on an MN and the MIHF on network. The function is used to respond to an intent of network initiated handover.

MIH Header Fields (SID=3, Opcode=2, AID=5)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SourceLinkIdentifier (Link Identifier TLV)
PreferredLinkList (not included if Status does not indicate “Success”) (List of Link PoA List TLV)
AvailableResourceSet (not included if Status does not indicate “Success”) (Handover Resource Response Set TLV)
HandoverStatus (not included if Status does not indicate “Success”) (Handover Status TLV)
IPConfigurationMethods (optional) (not included if Status does not indicate “Success”) (IP Address Configuration Methods TLV)
DHCPServerAddress (optional) (not included if Status does not indicate “Success”) (DHCP Server Address List TLV)
FAAddress (optional) (not included if Status does not indicate “Success”) (FA Address List TLV)
AccessRouterAddress (optional) (not included if Status does not indicate “Success”) (Access Router Address List TLV)
Status (Status TLV)

8.6.3.9 MIH_MN_HO_Candidate_Query request

The corresponding MIH primitive of this message is defined in 7.4.19.1.

This message is used by a network MIHF to communicate to an MIHF on the MN, of an intent to initiate handover.

MIH Header Fields (SID=3, Opcode=1, AID=6)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SourceLinkIdentifier (Link Identifier TLV)
CandidateLinkList (List of Link PoA List TLV)
QueryResourceList (Handover Resource Query List TLV)
IPConfigurationMethods (optional) (IP Address Configuration Methods TLV)
DHCPServerAddress (optional) (DHCP Server Address TLV)
FAAddress (optional) (FA Address TLV)
AccessRouterAddress (optional) (Access Router Address TLV)

8.6.3.10 MIH_MN_HO_Candidate_Query response

The corresponding MIH primitive of this message is defined in 7.4.19.3.

This message is used by an MIHF on the MN to respond to an MIH_Net_HO_Candidate_Query request message from a remote MIHF in the network.

MIH Header Fields (SID=3, Opcode=2, AID=6)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SourceLinkIdentifier (Link Identifier TLV)
PreferredCandidateLinkList (not included if Status does not indicate “Success”) (List of Link PoA List TLV)

AvailableResourceSet (not included if Status does not indicate “Success”) (Available Resource Set TLV)
IPConfigurationMethods (optional) (not included if Status does not indicate “Success”) (IP Address Configuration Methods TLV)
DHCPServerAddress (optional) (not included if Status does not indicate “Success”) (DHCP Server Address List TLV)
FAAddress (optional) (not included if Status does not indicate “Success”) (FA Address List TLV)
AccessRouterAddress (optional) (not included if Status does not indicate “Success”) (Access Router Address List TLV)
Status (Status TLV)

8.6.3.11 MIH_N2N_HO_Query_Resources request

The corresponding MIH primitive of this message is defined in 7.4.20.1.

This message is used by an MIHF on the MN to communicate to an MIHF in the network, of an intent to initiate handover. This message is also used to achieve IP address related information on the candidate network.

MIH Header Fields (SID=3, Opcode=1, AID=7)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
QueryResourceList (Handover Resource Query List TLV)
IPConfigurationMethods (optional) (IP Address Configuration Methods TLV)
DHCPServerAddress (optional) (DHCP Server Address TLV)
FAAddress (optional) (FA Address TLV)
AccessRouterAddress (optional) (Access Router Address TLV)
CandidateLinkList (optional) (Link Identifier List TLV)

8.6.3.12 MIH_N2N_HO_Query_Resources response

The corresponding MIH primitive of this message is defined in 7.4.20.3.

This message is used by an MIHF in the network to respond to an MIH_MN_HO_Candidate_Query request message from a remote MIHF on the MN. This is used to return the result of resource preparation of the impending handover and to notify the MIHF on the serving network of the link resource status and IP address related information of the candidate network.

MIH Header Fields (SID=3, Opcode=2, AID=7)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
ResourceStatus (not included if Status does not indicate “Success”) (Resource Status TLV)
AvailableResourceSet (not included if Status does not indicate “Success”) (Available Resource Set TLV)
IPConfigurationMethods (optional) (not included if Status does not indicate “Success”) (IP Address Configuration Methods TLV)
DHCPServerAddress (optional) (not included if Status does not indicate “Success”) (DHCP Server Address List TLV)
FAAddress (optional) (not included if Status does not indicate “Success”) (FA Address List TLV)
AccessRouterAddress (optional) (not included if Status does not indicate “Success”) (Access Router Address List TLV)
CandidateLinkList (optional) (not included if Status does not indicate “Success”) (Link Identifier List TLV)
IPAddressInformationStatus (optional) (not included if Status does not indicate “Success”) (IP Address Information Status TLV)
Status (Status TLV)

8.6.3.13 MIH_Net_HO_Commit request

The corresponding MIH primitive of this message is defined in 7.4.21.1.

This message is used by the MIHF to communicate the intent to commit to a handover request to a specific link and PoA.

MIH Header Fields (SID=3, Opcode=1, AID=8)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkType (Link Type TLV)
TargetNetworkInfoList (Target Network Info List TLV)
LinkActionList (Optional) (Link Action List TLV)

AssignedResourceSet (Assigned Resource Set TLV)

8.6.3.14 MIH_Net_HO_Commit response

The corresponding MIH primitive of this message is defined in 7.4.21.3.

This message is used by the MIHF to respond to a request to commit to a handover request to a specific link and PoA.

MIH Header Fields (SID=3, Opcode=2, AID=8)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
LinkType (not included if Status does not indicate “Success”) (Link Type TLV)
TargetNetworkInfo (not included if Status does not indicate “Success”) (Target Network Info TLV)
LinkActionsResponse (Optional) (not included if Status does not indicate “Success”) (Link Actions Response TLV)
Status (Status TLV)

8.6.3.15 MIH_N2N_HO_Commit request

The corresponding MIH primitive of this message is defined in 7.4.23.1.

This message is used by the MIHF on the serving network to communicate with its peer MIHF on the selected target network. This is used to request the target network to allocate resources to a mobile node that is about to move toward that network link and PoA.

MIH Header Fields (SID=3, Opcode=1, AID=10)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
MobileNodeIdentifier (Mobile Node MIHF ID TLV)
TargetMobileNodeLinkIdentifier (MN Link ID TLV)
TargetPoA (PoA TLV)
PreferredLinkList (Link PoA List TLV)
RequestedResourceSet (Requested Resource Set TLV)

8.6.3.16 MIH_N2N_HO_Commit response

The corresponding MIH primitive of this message is defined in 7.4.23.3.

This message is used by the MIHF on the selected target network to communicate with its peer MIHF on the serving network. This is used to respond to the MIH_N2N_HO_Commit request message.

MIH Header Fields (SID=3, Opcode=2, AID=10)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
MobileNodeIdentifier (Mobile Node MIHF ID TLV)
TargetLinkIdentifier (not included if Status does not indicate “Success”) (Link Identifier TLV)
AssignedResourceSet (not included if Status does not indicate “Success”) (Assigned Resource Set TLV)
Status (Status TLV)

8.6.3.17 MIH_MN_HO_Complete request

The corresponding MIH primitive of this message is defined in 7.4.24.1.

This message is used by the MIHF on the MN to communicate the status of handover operation to the MIHF on the target network.

MIH Header Fields (SID=3, Opcode=1, AID=11)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SourceLinkIdentifier (Link Identifier TLV)
TargetLinkIdentifier (New Link Identifier TLV)
HandoverResult (Handover Result TLV)

8.6.3.18 MIH_MN_HO_Complete response

The corresponding MIH primitive of this message is defined in 7.4.24.3.

This message is used by the MIHF on the target network to communicate the response following the completion of handover operation to the MN.

MIH Header Fields (SID=3, Opcode=2, AID=11)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SourceLinkIdentifier (Link Identifier TLV)
TargetLinkIdentifier (New Link Identifier TLV)
Status (Status TLV)

8.6.3.19 MIH_N2N_HO_Complete request

The corresponding MIH primitive of this message is defined in 7.4.25.1.

This message is used by the MIHF to communicate the status of handover operation.

MIH Header Fields (SID=3, Opcode=1, AID=12)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
SourceLinkIdentifier (Link Identifier TLV)
TargetLinkIdentifier (New Link Identifier TLV)
HandoverResult (Handover Result TLV)

8.6.3.20 MIH_N2N_HO_Complete response

The corresponding MIH primitive of this message is defined in 7.4.25.3.

This message is used by the MIHF to communicate the response following the completion of the handover operation. The message is used to communicate the preferred action to be taken w.r.t. resources associated with the previous connection. If the handover is successful, the resources may be released.

MIH Header Fields (SID=3, Opcode=2, AID=12)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)

SourceLinkIdentifier (Link Identifier TLV)
TargetLinkIdentifier (New Link Identifier TLV)
ResourceStatus (not included if Status does not indicate “Success”) (Resource Retention Status TLV)
Status (Status TLV)

8.6.4 MIH messages for information service

MIH Information service uses only two messages - MIH_Get_Information request and MIH_Get_Information response. Due to the need to support different query types and the need for flexibility to customize the query and response, the parameters and their usage in these two messages are substantially different from other MIH message parameters, and are therefore separately defined in the following sub-clauses.

8.6.4.1 MIH_Get_Information request

The corresponding MIH primitive of this message is defined in 7.4.26.1.

This message is used by an MIHF to retrieve a set of Information Elements provided by the information service. A single MIH_Get_Information request message can carry multiple queries of a combination of Binary, RDF_Data, RDF_Schema_URL and RDF_Schema types. However, there should be at least one query specified.

MIH Header Fields (SID=4, Opcode=1, AID=1)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
InfoQueryBinaryDataList (optional) (Info Query Binary Data List TLV)
InfoQueryRDFDataList (optional) (Info Query RDF Data List TLV)
InfoQueryRDFSchemaURL (optional) (Info Query RDF Schema URL TLV)
InfoQueryRDFSchemaList (optional) (Info Query RDF Schema List TLV)
MaxResponseSize (optional) (Max Response Size TLV)

8.6.4.2 MIH_Get_Information response

The corresponding MIH primitive of this message is defined in 7.4.26.3.

This is used as a response to the MIH_Get_Information request message. In one response message, one or more results can be returned.

MIH Header Fields (SID=4, Opcode=2, AID=1)
Source Identifier = sending MIHF ID (Source MIHF ID TLV)
Destination Identifier = receiving MIHF ID (Destination MIHF ID TLV)
InfoResponseBinaryDataList (optional) (Info Response Binary Data List TLV)
InfoResponseRDFDataList (optional) (Info Response RDF Data List TLV)
InfoResponseRDFSchemaURLList (optional) (Info Response RDF Schema URL List TLV)
InfoResponseRDFSchemaList (optional) (Info Response RDF Schema List TLV)
Status (Status TLV)

Annexes

Annex A Type identifiers for information elements

(normative)

Table A-1 lists the Type identifier values for different individual IEs and IE containers.

Table A-1—Type identifiers for information elements

Type	Identifier
TYPE_IE_NETWORK_TYPE	0x10000000
TYPE_IE_OPERATOR_IDENTIFIER	0x10000001
TYPE_IE_SERVICE_PROVIDER_IDENTIFIER	0x10000002
TYPE_IE_COUNTRY_CODE	0x10000003
TYPE_IE_ACCESS_NETWORK_IDENTIFIER	0x10000100
TYPE_IE_ACCESS_NETWORK_AUX_ID	0x10000101
TYPE_IE_ROAMING_PARTNERS	0x10000102
TYPE_IE_COST	0x10000103
TYPE_IE_NETWORK_SECURITY	0x10000104
TYPE_IE_NETWORK_QOS	0x10000105
TYPE_IE_NETWORK_DATA_RATE	0x10000106
TYPE_IE_NETWORK_CHANNEL_RANGE	0x10000107
TYPE_IE_NETWORK_IP_CONFIG_METHODS	0x10000108
TYPE_IE_NETWORK_CAPABILITIES	0x10000109
TYPE_IE_NETWORK_LIST_SUPPORTED_LCP	0x1000010A
TYPE_IE_NETWORK_MOBILITY_MGMT_PROT	0x1000010B
TYPE_IE_POA_MAC_ADDRESS	0x10000200
TYPE_IE_POA_LOCATION	0x10000201
TYPE_IE_POA_CHANNEL_RANGE	0x10000202
TYPE_IE_POA_SYSTEM_INFORMATION	0x10000203
TYPE_IE_POA_SUBNET_INFORMATION	0x10000204
TYPE_IE_POA_IP_ADDRESS	0x10000205
TYPE_IE_CONTAINER_LIST_OF_NETWORKS	0x10000300
TYPE_IE_CONTAINER_NETWORK	0x10000301
TYPE_IE_CONTAINER_POA	0x10000302

Annex B Data type definition

(normative)

B.1 General

This Annex defines data types used in the IEEE 802.21 standard. Any variable-length data type in this specification contains information needed for determining the end of data.

B.2 Basic data types

The data types defined in this subclause are used as the basis for defining any other data types. All basic data types are for general purpose. The "Binary Encoding Rule" column in Table B-1 describes the encoding rules used when the data types are carried in MIH protocol messages.

Table B-1—Basic data types

Type Name	Definition	Range	Binary Encoding Rule
BITMAP(size)	A bitmap of the specified size. Usually used to represent a list of IDs.	Each bit has a value of '0' or '1'.	A BITMAP(N), where N must be a multiple of 8, is made up of an N/8 octet values and encoded in network byte order.
CHAR(size)	An array of characters. Each character is encoded as an octet. The size specifies the length.	N/A	The characters are encoded in network byte order.
CHOICE(DATATYPE1,DATATYPE2[,…])	A data type that consists only one of data types DATATYPE1,DATATYPE2[,…]).	N/A	A one-octet Selector field, followed by a variable length Value field. The Selector value determines the data type. If Selector==i, (i+1)-th data type in the list of data types DATATYPE1,DATATYPE2,[,…] is selected. The Selector value is encoded as UNSIGNED_INT(1). The Value field is encoded using the encoding rule for the selected data type.
INFORMATION_ELEMENT	A binary encoded structure for Information Elements.	N/A	See Subclause 6.5.6
INTEGER(size)	A signed integer of the specified size in number of octets.	Each octet has a value of 0x00 to 0xff.	Each octet of an INTEGER(N) value [N=1,2,...] is encoded in network-byte order into an N-octet field. The most significant bit of the first octets is the sign bit. If the sign bit is set, it indicates a negative integer. Otherwise, it indicates a non-negative integer. A negative integer is encoded as 2's complement.
LIST(DATATYPE)	A list of values of DATATYPE	N/A	See subclause B.2.1 for details.

Table B-1—Basic data types

Type Name	Definition	Range	Binary Encoding Rule
NULL	A data type with empty data.	N/A	No octet is encoded for this data type. This data type is used to define an optional data type.
SEQUENCE(DATATYPE1, DATATYPE2[,...])	A data type that consist of two or more data types.	N/A	DATATYPE1,DATATYPE2,[,…] are encoded in the order of appearance. Each data type is encoded using the encoding rule for the data type.
UNSIGNED_INT(size)	An unsigned integer of the specified size in number of octets.	Each octet has a value of 0x00 to 0xff.	Each octet of an UNSIGNED_INT(N) value [N=1,2,...] is encoded in network-byte order into an N-octet field.

B.2.1 Encoding rule for data type LIST(DATATYPE)

A variable length *Length* field is followed by a variable length *Value* field. The *Length* field shall be interpreted as follows:

Case 1: If the number of list elements in the *Value* field is less than 128, the size of the *Length* field is always 1 octet and the MSB of the octet is set to the value ‘0’. The values of the other seven bits of this octet indicate the actual number of list elements in the *Value* field.

Case 2: If the number of list elements in the *Value* field is exactly 128, the size of the *Length* field is one octet. The MSB of the *Length* octet is set to the value '1' and the other seven bits of this octet are all set to the value ‘0’.

Case 3: If the number of list elements in the *Value* field is greater than 128, then the *Length* field is always greater than 1 octet. The MSB of the first octet of the *Length* field is set to the value ‘1’ and the remaining 7 bits of the first octet indicate the number of octets that are appended further. The number represented by the 2nd and subsequent octets of the *Length* field, when added to 128, indicates the total number of list elements in the *Value* field.

For example, an attribute of type LIST(LINK_ID) with two elements is encoded as following:
(LINK_ID is defined in subclause B.3.4)

LIST(LINK_ID) (total length = 29 octets)	
Length (1 octet)	Value (28 octets)

2	LINK_ID (14 octets)				LINK_ID (14 octets)			
	LINK_TYPE (type:UNSIGNED_INT(4)) (4-octet)	LINK_ADDRESS			LINK_TYPE (type:UNSIGNED_INT(4)) (4-octet)	LINK_ADDRESS		
		Choice	MAC_ADDRESS (type:OCTET_STRING =LIST(CHAR(1))) (9-octet)			Choice	MAC_ADDRESS (type:OCTET_STRING =LIST(CHAR(1))) (9-octet)	
	19 (802.11)	Length (1-octet)	Length (1-octet)	Value (8-octet)	27 (802.16)	Length (1-octet)	Length (1-octet)	Value (8-octet)
		0	8	2-octet address family followed by 6-octet IEEE MAC address.		0	8	2-octet address family followed by 6-octet IEEE MAC address.

B.3 Derived data types

B.3.1 General

Derived data types are those which are derived from other data types or parent data types. A derived data type uses the same encoding as the parent data type.

B.3.2 General data types

The derived data types defined in this subclause are for general purpose only.

Table B-2—General data types

Type Name	Derived From	Definition	Valid Range
ENUMERATED	UNSIGNED_INT(1)	Represents an enumerated attribute.	0-255
BOOLEAN	ENUMERATED	Represents an boolean attribute	0: False 1: True
OCTET_STRING	LIST(CHAR(1))	Represents an array of octets	N/A
PERCENTAGE	UNSIGNED_INT(1)	Represents a percentage	0-100
STATUS	ENUMERATED	Represents the status of a primitive execution	0 Success 1 Unspecified Failure 2 Rejected 3 Authorization Failure 4 Network Error

B.3.3 Data types for addresses

The data types defined in this subclause are related to addresses of network elements.

Table B-3—Data types for address

Type Name	Derived From	Definition	Valid Range
3GPP_2G_CELL_ID	SEQUENCE(PLMN_ID, LAC, CI)	A data type to represent 3GPP 2G cell identifier	N/A
3GPP_3G_CELL_ID	SEQUENCE(PLMN_ID, CELL_ID)	A data type to represent 3GPP 3G cell identifier.	N/A
CELL_ID	UNSIGNED_INT(4)	This data type identifies a cell uniquely within 3GPP UTRAN and consists of RNC-ID and C-ID as defined in 3GPP TS 25.401.	0..268435455
CI	CHAR(2)	The BSS and cell within the BSS are identified by Cell Identity (CI). See 3GPP TS 23.003.	0000..FFFF
IP_ADDRESS	TRANSPORT_ADDRESS	Represents an IP address	The AddressType is either 1 (IPv4) or 2 (IPv6).
LAC	CHAR(2)	Location Area Code (LAC) is a fixed length code (of 2 octets) identifying a location area within a PLMN. See 3GPP TS 23.003.	0001...FFFD and FFFF. That is 0000 and FFFE are not allowed.
LINK_ADDRESS	CHOICE(MAC_ADDRESS, 3GPP_3G_CELL_ID, 3GPP_2G_CELL_ID, OTHER_L2_ADDRESS)	A data type to represent an address of any link-layer.	N/A
MAC_ADDRESS	TRANSPORT_ADDRESS	Represents a MAC address.	The AddressType contains the one used for a specific link layer.
OTHER_L2_ADDRESS	OCTET_STRING	A data type to represent a link-layer address other than the address already defined. For example, SSID.	N/A
PLMN_ID	OCTET_STRING(3)	The Public Land Mobile Network (PLMN) unique identifier. PLMN_ID consists of Mobile Country Code (MCC; three digits) and Mobile Network Code (MNC; two or three digits). Reference 3GPP TS 23.003. This is to represent the access network identifier.	Coding of PLMN_ID is defined in 3GPP TS 25.413.
TRANSPORT_ADDRESS	OCTET_STRING	A type to represent a transport address using Address base type defined in RFC3588. The first 2-octets contain AddressType.	AddressType values are defined in http://www.iana.org/assignments/address-family-numbers.

B.3.4 Data types for link identification and manipulation

The data types defined in this subclause are used for representing attributes for identification and manipulation of links.

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
BATTERY_LEVEL	INTEGER(1)	Represents battery level.	0 - 100. -1: battery level unknown.
CHANNEL_ID	UNSIGNED_INT(2)	Channel identifier as defined in the specific link technology (e.g. SDO).	0-65535
CONFIG_STATUS	BOOLEAN	The status of link parameter configuration.	True: Success False: Error
DEVICE_INFO	OCTET_STRING	A non-NULL terminated string whose length shall not exceed 253 octets, representing information on manufacturer, model number, revision number of the software/ firmware and serial number in displayable text.	N/A
DEVICE_STATES_REQ	BITMAP(16)	A set of device status request.	Bit 0: DEVICE_INFO Bit 1: BATTERY_LEVEL Bit 2-15: (Reserved)
DEVICE_STATES_RSP	CHOICE(DEVICE_INFO, BATTERY_LEVEL)	Represents a device status.	N/A
LINK_AC_EXEC_TIME	UNSIGNED_INT(2)	Time (in ms) to elapse before an action needs to be taken. A value of 0 indicates that the action shall be taken immediately. Time elapsed shall be calculated from the instance the command arrives until the time when the execution of the action is carried out.	0-65535
LINK_AC_RESULT_CODE	ENUMERATED	Link action result.	0: Success 1: Failure
LINK_ACTION	SEQUENCE(LINK_ACTION_TYPE, LINK_ACTION_ATTRIBUTE)	Link action.	N/A

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
LINK_ACTION_ATTRIBUTE	BITMAP(8)	Link action attribute that can be executed along with a valid link action. Detail description of each attribute is in Table B-6	Bit 0: LINK_SCAN Bit 1: LINK_RESOURCE_RETAIN Bit 2: DATA_FORWARDING_REQUEST Bit 3-7: (Reserved)
LINK_ACTION_REQ	SEQUENCE(LINK_TUPLE_ID, LINK_ACTION, LINK_AC_EXEC_TIME)	A set of handover action request parameters.	N/A
LINK_ACTION_RSP	SEQUENCE(LINK_ID, LINK_AC_RESULT_CODE, CHOICE(NULL, LIST(LINK_SCAN_RSP))	A set of link action returned results.	N/A
LINK_ACTION_TYPE	ENUMERATED	An action for a link. The meaning of each link action is defined in Table B-5.	0: NONE 1: LINK_DISCONNECT 2: LINK_LOW_POWER 3: LINK_POWER_DOWN 4: LINK_POWER_UP 5-255: (Reserved)
LINK_COMMAND_LIST	BITMAP(32)	A set of link commands.	Bit 0: Link_Capability_Discover Bit 1: Link_Event_Subscribe Bit 2: Link_Event_Unsubscribe Bit 3: Link_Configure_Thresholds Bit 4: Link_Get_Parameters Bit 5: Link_Actions Bit 6-31: (Reserved)
LINK_CONFIG_PARAM	SEQUENCE(LINK_PARAM_TYPE, CHOICE(TIMER_INTERVAL, NULL), LIST(THRESHOLD))	A link configuration parameter.	N/A
LINK_CONFIG_STATUS	SEQUENCE(LINK_PARAM_TYPE, CONFIG_STATUS)	The status of link parameter configuration.	N/A
LINK_DESCRIPTORS_REQ	BITMAP(16)	A set of link descriptors	Bit 0: Number of Classes of Service Supported Bit 1: Number of Queues Supported Bits 2-15: (Reserved)
LINK_DESCRIPTORS_RSP	CHOICE(NUM_COS, NUM_QUEUE)	Descriptors of a link	N/A
LINK_DOWN_REASON	ENUMERATED	Represents the reason of a link down event.	See Table B-7.

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
LINK_EVENT_LIST	BITMAP(32)	A list of link events. The specified event is selected if the corresponding bit is set to 1.	Bit 0: Link_Up Bit 1: Link_Down Bit 2: Link_Going_Down Bit 3: Link_Detected Bit 4: Link_Parameters_Report Bit 5: Link_Event_Rollback Bit 6: Link_PDU_Transmit_Status Bit 7: Link_Handover_Imminent Bit 8: Link_Handover_Complete Bit 9-31: (Reserved)
LINK_GOING_DOWN_REASON	ENUMERATED	Represents the reason of a link going down.	See Table B-8.
LINK_ID	SEQUENCE(LINK_TYPE LINK_ADDRESS)	The identifier of a link that is not associated with any PoA. The LINK_ADDRESS contains a link address of MN. This may be used for the current link being used by MN or the link that was used by MN before handover.	N/A
LINK_MIH_CAPABILITY_FLAG	BITMAP(8)	Represents if MIH capability is supported or not. If the bit is set, it indicates that the capability is supported.	Bit 0: MIHF Supported Bit 1: MIHF unknown Bit 2: ES Supported Bit 3: CS Supported Bit 4: IS Supported Bit 5: MIH Capability unknown Bit 6 -7: (Reserved)
LINK_PARAM	SEQUENCE(LINK_PARAM_TYPE, CHOICE(LINK_PARAM_VALUE, QOS_LIST))	Represents a link parameter type and value pair.	N/A
LINK_PARAM_802_11	ENUMERATED	A type to represent a link parameter for 802.11.	0: RSSI of the beacon channel 1: Flag indicating no QoS resources available (this applies when the traffic stream to be transmitted is on an access category configured for HCCA operation and the request for bandwidth was denied by the available APs in the access network) 2-255: (Reserved)
LINK_PARAM_802_16	ENUMERATED	A type to represent a link parameter for 802.16.	0-255: (Reserved)
LINK_PARAM_802_20	ENUMERATED	A type to represent a link parameter for 802.20.	0-255: (Reserved)
LINK_PARAM_802_22	ENUMERATED	A type to represent a link parameter for 802.22.	0-255: (Reserved)

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
LINK_PARAM_CDMA2000	ENUMERATED	A type to represent a link parameter for CDMA2000.	0: PILOT_STRENGTH 1-255: (reserved)
LINK_PARAM_CDMA2000_HRPD	ENUMERATED	A type to represent a link parameter for CDMA2000 HRPD.	0: PILOT_STRENGTH 1-255: (Reserved)
LINK_PARAM_EDGE	ENUMERATED	A type to represent a link parameter for EDGE.	0-255: (Reserved)
LINK_PARAM_ETHERNET	ENUMERATED	A type to represent a link parameter for Ethernet.	0-255: (Reserved)
LINK_PARAM_GENERAL	ENUMERATED	A type to represent a link parameter that is applicable to any link type.	0: Speed 1: Signal Strength 2: SINR 3:Throughput 4: Packet Error Rate 5-255: (Reserved)
LINK_PARAM_GSM_GPRS	ENUMERATED	A type to represent a link parameter for GSM and GPRS. See 3GPP TS 25.008.	0: RxQual 1: RsLev 2: Mean BEP 3: StDev BEP 4-255: (Reserved)
LINK_PARAM_QOS_LIST	ENUMERATED	A type to represent QOS_LIST parameters.	0: Maximum number of differentiable classes of service supported. 1: Throughput 2: Packet Error Rate 3: Minimum packet transfer delay for all CoS 4: Average packet transfer delay for all CoS 5: Maximum packet transfer delay for all CoS 6: Packet transfer delay jitter for all CoS 7: Packet loss rate for all Cos 8-255: (Reserved)
LINK_PARAM_REPORT	SEQUENCE(LINK_PARAM, LINK_REPORT_REASON)	Represents a link parameter report.	N/A

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
LINK_PARAM_TYPE	CHOICE(LINK_PARAM_GENERAL, LINK_PARAM_QOS_LIST, LINK_PARAM_GSM, LINK_PARAM_GPRS, LINK_PARAM_EDGE, LINK_PARAM_ETHERNET, LINK_PARAM_802_11, LINK_PARAM_CDMA2000, LINK_PARAM_WCDMA_FDD, LINK_PARAM_CDMA2000_HRPD, LINK_PARAM_802_16, LINK_PARAM_802_20, LINK_PARAM_802_22)))	Measurable link parameter for which thresholds are being set.	N/A
LINK_PARAM_VALUE	UNSIGNED_INT(2)	The current value of the parameter. The format of the media-dependent value is defined in the respective media specification standard and the equivalent number of bits (i.e. first bits) of this data type is used. In case that there are remaining unused bits in the data type, these are marked as all-zeros (‘0’).	0-65535
LINK_PARAM_WCDMA_FDD	ENUMERATED	A type to represent a link parameter for UMTS. See 3GPP TS 25.215.	0: CPICH RSCP 1: PCCPCH RSCP 2: UTRA carrier RSSI 3: GSM carrier RSSI 4: CPICH Ec/No 5: Transport channel BLER 6: UE transmitted power 7-255: (Reserved)
LINK_POA_LIST	SEQUENCE(LINK_ID, LIST(LINK_ADDRESS))	A list of PoAs for a particular link. The LIST(LINK_ADDRESS) is a list of PoA link addresses and is sorted from most preferred first to least preferred last.	N/A
LINK_REPORT_REAONS	ENUMERATED	The reason for generating a link parameter report.	0: PERIODIC 1: ABOVE_THRESHOLD 2: BELOW_THRESHOLD 2-255: (Reserved)

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
LINK_RESOURCE_RETENTION_STATUS	BOOLEAN	Represent a status of resource.	True: Retain resource. False: Release resource.
LINK_RESOURCE_STATUS	BOOLEAN	Indicates if a resource is available or not.	True: Available False: Not available
LINK_SCAN_RSP	SEQUENCE(LINK_ADDRESS, ACCESS_NETWORK_ID, SIGNAL_STRENGTH)	Represents a scan response. The LINK_ADDRESS contains the PoA link address. The PoA belongs to the ACCESS_NETWORK_ID with the given SIGNAL_STRENGTH.	N/A
LINK_STATES_REQ	BITMAP(16)	Link status to be requested.	Bit 0: OPERATION_MODE Bit 1: CHANNEL_ID Bit 2-15: (Reserved)
LINK_STATES_RSP	CHOICE(OPERATION_MODE, CHANNEL_ID)	Represents a link state.	N/A
LINK_STATUS_REQ	SEQUENCE(CHOICE(LINK_STATES_REQ, NULL), CHOICE(LIST(LINK_PARAM_TYPE), NULL), CHOICE(LINK_DESCRIPTORS_REQ, NULL))	Represents the possible information to request from a link.	N/A
LINK_STATUS_RSP	SEQUENCE(LINK_ID, LIST(LINK_STATES_RSP), LIST(LINK_PARAM), LIST(LINK_DESCRIPTORS_RSP))	A set of link status parameters for a given LINK_ID	N/A
LINK_TUPLE_ID	SEQUENCE(LINK_ID, CHOICE(LINK_ADDRESS, NULL))	The identifier of a link that may be associated with a PoA. The optional LINK_ADDRESS contains a link address of PoA.	N/A

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
LINK_TYPE	UNSIGNED_INT(4)	Represents the link type. Note, the values defined are made consistent with RADIUS NAS-Port-Type definitions as specified by Internet Assigned Numbers Authority (IANA). (see RFC 2865)	0: Reserved 1: Wireless - GSM 2: Wireless - GPRS 3: Wireless - EDGE 15: Ethernet 18: Wireless - Other 19: Wireless - IEEE 802.11 22: Wireless - CDMA2000 23: Wireless - UMTS 24: Wireless - cdma2000-HRPD 27: Wireless - IEEE 802.16 28: Wireless - IEEE 802.20 29: Wireless - IEEE 802.22
NUM_COS	UNSIGNED_INT(1)	The maximum number of differentiable classes of service supported.	0-255
NUM_QUEUE	UNSIGNED_INT(1)	The maximum number of differentiable classes of service supported.	0-255
OPERATION_MODE	ENUMERATED	The link power mode.	0: Normal Mode 1: Power Saving Mode 2: Power Down 3-255: (Reserved)
OPERATION_MODE_STATUS	ENUMERATED	Operation mode setting status.	0: Success 1: Error 2-255: (Reserved)
SIGNAL_STRENGTH	PERCENTAGE	Represents the signal strength.	00 - 24: Poor 25 - 49: Average 50 - 74: Good 75 -100: Excellent
THRESHOLD	SEQUENCE(THRESHOLD_VALUE, THRESHOLD_CROSSING_DIRECTION)	A link threshold. The threshold is considered crossed when the value of the link parameter passes the threshold in the specified direction.	N/A
THRESHOLD_CROSSING_DIRECTION	ENUMERATED	The direction the threshold is to be crossed.	0: ABOVE_THRESHOLD 1: BELOW_THRESHOLD 2-255: (Reserved)

Table B-4—Data types for links

Type Name	Derived From	Definition	Valid Range
THRESHOLD_VALUE	UNSIGNED_INT(2)	Threshold value. The format of the media-dependent value is defined in the respective media specification standard and the equivalent number of bits (i.e. first bits) of this data type is used. In case that there are remaining unused bits in the data type, these are marked as all-zeros (‘0’).	0-65535
TIMER_INTERVAL	UNSIGNED_INT(2)	This timer value (ms) is used to set the interval between periodic reports.	0-65535

Table B-5—Link actions

Action Name	Description
LINK_DISCONNECT	Disconnect the link connection directly.
LINK_LOW_POWER	Cause the link to adjust its battery power level to be low power consumption.
LINK_POWER_DOWN	Cause the link to power down and turn off the radio.
LINK_POWER_UP	Cause the link to power up and establish L2 connectivity. For UMTS link type, power up lower layers and establish PDP context.

Table B-6—Link action attributes

Action Name	Description
DATA_FORWARDING_REQUEST	This indication requires the buffered data at the old serving PoS entity to be forwarded to the new target PoS entity in order to avoid data loss. This action can be taken immediately after the old serving PoS receives MIH_N2N_HO_Commit response message from the new target PoS, or the old serving PoS receives MIH_Net_HO_Commit response message from the MN. This is not valid on UMTS link type.
LINK_RESOURCE_RETAIN	The link will be disconnected but the resource for the link connection still remains so reestablishing the link connection later can be more efficient.
LINK_SCAN	Cause the link to perform a scan.

Table B-7—Link Down reason code

Reason code	Name	Description
0	EXPLICIT DISCONNECT	The link is down because of explicit disconnect procedures initiated either by MN or network
1	PACKET TIMEOUT	The link is down because no acknowledgements were received for transmitted packets within the specified time limit.
2	NO RESOURCE	The link is down because there were no resources to maintain the connection
3	NO BROADCAST	The link is down because broadcast messages (control frames such as beacons) could not be received by MN
4	AUTHENTICATION FAILURE	Authentication failure
5	BILLING FAILURE	Billing failure
6-127	(*Reserved*)	Reserved for IEEE 802.21 future use.
128-255	Vendor specific reason codes	Vendors may specify their own specific reason codes in this range.

Table B-8—Link Going Down reason code

Reason code	Name	Description
0	EXPLICIT DISCONNECT	The link is going to be down because explicit disconnect procedures will be initiated either by MN or network. For example, when a BS has decided to shutdown for administrative reasons or an operator of the terminal has decided to execute a handover manually, a Link_Going_Down trigger may be sent to the MIHF.
1	LINK PARAMETER DEGRADING	The link is going to be down because broadcast messages (control frames such as beacons) could not be received by MN.
2	LOW POWER	The link is going to be down because the power level of the terminal is low and the current link will not be maintained in such a low power level. Mobile terminals usually have limited battery supply, and when the battery level of the terminal is low, a terminal may choose a link that has lower power consumption for handover according to the received Link_Going_Down triggers with the reason code RC_LOW_POWER. This will lengthen the usable time for the terminal
3	NO RESOURCE	The link is going to be down because there will be no resources to maintain the current connection. For example, a BS may have too many users and send Link_Going_Down indications to terminals when the links with them can not be kept because of insufficient resources. An another example is that the users with higher priority may preempt the ones with lower priority when no more resources can be allocated in 3GPP, and this also may cause a Link_Going_Down indication with RC_NORESOURCE reason code.
4-127	(*Reserved*)	Reserved for IEEE 802.21 future use.
128-255	Vendor specific reason codes	Vendors may specify their own specific reason codes in this range.

B.3.5 Data types for QoS

The data types defined in this subclause are related to QoS.

.

Table B-9—Data types for QoS

Type Name	Derived From	Definition	Valid Range
QOS_LIST	SEQUENCE(NUM_COS_TYPES, LIST(COS_MIN_PACKET_TRANSFER_DELAY), LIST(COS_AVG_PACKET_TRANSFER_DELAY), LIST(COS_MAX_PACKET_TRANSFER_DELAY) , LIST(COS_PACKET_DELAY_JITTER), LIST(COS_PACKET_LOSS_RATE))	A list of Class of Service parameters (CoS).	N/A
NUM_QOS_TYPES	UNSIGNED_INT(1)	A type to represent the maximum number of differentiable classes of service supported.	0 – 255
THROUGHPUT	UNSIGNED_INT(4)	A type to represent the maximum data transfer rate achievable. It is measured in kb/s.	0 - 2^32-1
PACKET_ERROR_RATE	UNSIGNED_INT(2)	A type to represent a value equal to the integer part of the result of multiplying -100 times the log10 of the ratio between the number of packets received in error and the total number of packets transmitted in a link population of interest.	N/A
COS_MIN_PACKET_TRANSFER_DELAY	SEQUENCE(COS_ID, UNSIGNED_INT(2))	A type to represent the minimum packet transfer delay in ms for the specific CoS specified by the COS_ID.	N/A
COS_AVG_PACKET_TRANSFER_DELAY	SEQUENCE(COS_ID, UNSIGNED_INT(2))	A type to represent the average packet transfer delay in ms for the specific CoS specified by the COS_ID.	N/A
COS_MAX_PACKET_TRANSFER_DELAY	SEQUENCE(COS_ID, UNSIGNED_INT(2))	A type to represent the maximum packet transfer delay in ms for the specific CoS specified by the COS_ID.	N/A
COS_PACKET_DELAY_JITTER	SEQUENCE(COS_ID, UNSIGNED_INT(2))	A type to represent the packet transfer delay jitter in ms for the specific CoS specified by the COS_ID.	N/A

Table B-9—Data types for QoS

Type Name	Derived From	Definition	Valid Range
COS_PACKET_LOSS_RATE	SEQUENCE(COS_ID, UNSIGNED_INT(2))	A type to represent the packet loss rate for the specific CoS specified by the COS_ID. The loss rate is equal to the integer part of the result of multiplying -100 times the log10 of the ratio between the number of packets lost and the total number of packets transmitted in the class population of interest.	N/A
COS_ID	UNSIGNED_INT(1)	A type to represent a class of service identifier.	0 - 255

B.3.6 Data types for location

Table B-10—Data types for location

Type name	Derived from	Definition	Valid Range
LOCATION	CHOICE(CIVIC_LOCATION, GEO_LOCATION, CELL_ID)	A type to represent the format and value of the location information. The location can be civic location, geospacial location, or a cellular ID value as reference location.	N/A
CIVIC_LOCATION	CHOICE(BINARY_CIVIC_LOCATION, XML_CIVIC_LOCATION)	A type to represent a civic address.	N/A
BINARY_CIVIC_LOCATION	SEQUENCE(CIVIC_COUNTRY_CODE, CIVIC_ADDR_ELEM)	A type to represent a binary-formatted civic address.	See CIVIC_COUNTRY_CODE and CIVIC_ADDR_ELEM definitions.
XML_CIVIC_LOCATION	OCTET_STRING	A type to represent an XML-formatted civic location.	Civic address elements, as described in RFC4119.
CIVIC_COUNTRY_CODE	CHAR(2)	A type to represent country code in BINARY_CIVIC_LOCATION.	Two letter ISO 3166-1 country code in capital ASCII letters.

Table B-10—Data types for location

Type name	Derived from	Definition	Valid Range
CIVIC_ADDR_ELEM	OCTET_STRING	A type to represent civic address elements in BINARY_CIVIC_LOCATION.	Civic address elements, as described in RFC4776.
GEO_LOCATION	CHOICE(BINARY_GEO_LOCATION, XML_GEO_LOCATION)	A type to represent a geospatial location.	N/A
BINARY_GEO_LOCATION	CHAR(16)	A type to represent a binary-formatted geospatial location.	See Table B-11.
XML_GEO_LOCATION	OCTET_STRING	A type to represent an XML-formatted geospatial location.	Geo address elements as described in IETF RFC 4119. For example, <gml:location> <gml:Point gml:id="point1" srsName="epsg:4326"> <gml:coordinates>37:46:30N 122:25:10W</gml:coordinates> </gml:Point> </gml:location>

Table B-11—Value field format of PoA location information (geospatial location)

Syntax	Length (bits)	Notes (See RFC 3825 for details)
LatitudeResolution (LaRes)	6	Latitude resolution. 6 bits indicating the number of valid bits in the fixed-point value of Latitude. Any bits entered to the right of this limit should not be considered valid and might be purposely false, or zeroed by the sender.
Latitude	34	A 34 bit fixed point value consisting of 9 bits of integer and 25 bits of fraction. Latitude should be normalized to within +/- 90 degrees. Positive numbers are north of the equator and negative numbers are south of the equator.
LongitudeResolution (LoRes)	6	Longitude resolution. 6 bits indicating the number of valid bits in the fixed-point value of Longitude. This value is the number of high-order Longitude bits that should be considered valid. Any bits entered to the right of this limit should not be considered valid and might be purposely false, or zeroed by the sender.
Longitude	34	A 34 bit fixed point value consisting of 9 bits of integer and 25 bits of fraction. Longitude should be normalized to within +/- 180 degrees. Positive values are East of the prime meridian and negative (2s complement) numbers are West of the prime meridian.

Table B-11—Value field format of PoA location information (geospatial location)

Syntax	Length (bits)	Notes (See RFC 3825 for details)
AltitudeType (AT)	4	Following codes are defined: 1: Meters: in 2s-complement fixed-point 22-bit integer part with 8-bit fraction. If AT = 1, an AltRes value 0.0 would indicate unknown altitude. The most precise Altitude would have an AltRes value of 30. Many values of AltRes would obscure any variation due to vertical datum differences. 2: Floors: in 2s-complement fixed-point 22-bit integer part with 8-bit fraction. AT = 2 for Floors enables representing altitude in a form more relevant in buildings which have different floor-to-floor dimensions.
AltitudeResolution (AltRes)	6	Altitude resolution. 6 bits indicating the number of valid bits in the altitude. Values above 30 (decimal) are undefined and reserved.
Altitude	30	A 30 bit value defined by the AT field.
Datum	8	Following codes are defined: 1: WGS 2: NAD 83 (with associated vertical datum for North American vertical datum for 1998) 3: NAD 83 (with associated vertical datum for Mean Lower Low Water (MLLW))

B.3.7 Data types for IP configuration

Table B-12—Data types for IP configuration

Type Name	Derived From	Definition	Valid Range
IP_CONFIG_METHODS	BITMAP(32)	A set of IP configuration methods.	Bit 0: IPv4 static configuration Bit 1: IPv4 dynamic configuration (DHCPv4) Bit 2: Mobile IPv4 with foreign agent (FA) care of address (CoA) (FA-CoA) Bit 3: Mobile IPv4 without FA (Co-located CoA) Bits 4-10: reserved for IPv4 address configurations Bit 11: IPv6 stateless address configuration Bit 12: IPv6 stateful address configuration (DHCPv6) Bit 13: IPv6 manual configuration Bits 14-31: (Reserved)
IP_CONFIG_STATUS	BITMAP(8)	Status of the IP configuration methods.	Bit 0: IP configuration Method is not available Bit 1: DHCP Server address is not available Bit 2: FA address is not available Bit 3: Access Router Address is not available Bit 4: No information is provided due to accessibility of same entity (FA, Access Router, DHCP Server, etc.) Bit 5-7: (Reserved)
IP_MOBILITY_MGMT	BITMAP(16)	Indicates the supported mobility management protocols.	Bit 0: Mobile IPv4 (RFC3344) Bit 1: Mobile IPv4 Regional Registration (RFC4857) Bit 2: Mobile IPv6 (RFC3375) Bit 3: Hierarchical Mobile IPv6 (RFC4140) Bit 4: Low Latency Handoffs (RFC4881) Bit 5: Fast Handovers for Mobile IPv6 (RFC4068) Bit 6: IKEv2 Mobility and Multihoming Protocol (RFC4555) Bit 7-15: (Reserved)
IP_PREFIX_LEN	UNSIGNED_INT(1)	The length of an IP subnet prefix.	0 - 32 for IPv4 subnet. 0 - 64, 66-127 for IPv6 subnet.

Table B-12—Data types for IP configuration

Type Name	Derived From	Definition	Valid Range
IP_RENEWAL_FLAG	BOOLEAN	Indicates whether MN's IP address needs to be changed or not.	True: Change required. False: Change not required.
IP_SUBNET_INFO	SEQUENCE(IP_PREFIX_LEN, IP_ADDRESS)	Represent an IP subnet. The IP_PREFIX_LEN contains the bit length of the prefix of the subnet to which the IP_ADDRESS belongs.	N/A

B.3.8 Data types for information elements

Data types defined in this subclause are used only by IEs.

Table B-13—Data types for information elements

Type Name	Derived From	Definition	Valid Range
ACCESS_NETWORK_AUX_ID	OCTET_STRING	A type to represent an auxiliary access network identifier.	This is SSID if network type is IEEE 802.11.
ACCESS_NETWORK_ID	OCTET_STRING	A type to represent a network identifier.	A non-NULL terminated string whose length shall not exceed 253 octets.
BAND_CLASS	INTEGER(1)	CDMA band class.	
BANDWIDTH	INTEGER(1)	Channel bandwidth.	
BASE_ID	INTEGER(2)	Base station identifier.	
BS ID	INTEGER(6)	Base Station Identifier.	
BURTST_PROFILE	SEQUENCE(DOWN_BURST_PRPOFILE, UP_BURST_PROFILE)		
CHANNEL_RANGE	SEQUENCE(UNSIGNED_INT(4), UNSIGNED_INT(4))	A type contains two numbers. The first unsigned integer is the low range. The second unsigned integer is the high range. Both values are in KHz.	The first unsigned integer value should always be less or equal to the second unsigned integer. For example, [2412000, 5000000] would indicate a channel range from 2.4GHz to 5.0GHz.

Table B-13—Data types for information elements

Type Name	Derived From	Definition	Valid Range
COST	SEQUENCE(COST_UNIT, COST_VALUE, COST_CURRENCY)	A type to represent a cost.	N/A
COST_CURRENCY	CHAR(3)	A type to represent the currency of a cost.	A three-letter currency code (e.g., “USD”) specified by ISO 4217 [ISO 4217].
COST_UNIT	ENUMERATED	A type to represent the unit of a cost.	0: second 1: minute 2: hours 3: day 4: week 5: month 6: year 7: free 8: flat rate 9-255: (Reserved)
COST_VALUE	SEQUENCE(UNSIGNED_INT(4), UNSIGNED_INT(2))	A type to represent the value of a cost.	The first 4-octet contains the integer part of the cost. The last 2-octet contains the fraction part where it represents a 3-digit fraction. Therefore, the value range of the fraction part is [0,999]. For example, for a value of “0.5”, the integer part is zero and the fraction part is 500.
COUNTRY_CODE	CHAR(2)	A type to represent the country code.	Two letter ISO 3166-1 country code in capital ASCII letters.
DATA_RATE	UNSIGNED_INT(4)	A type to represent the maximum data rate in kbps.	0 - 2^32-1
DCD_UCD	SEQUENCE(BS ID, BANDWIDTH, FREQUENCY, EIRP, GAP, BURST_PROFILE, RANGING_CDMA_CODE)	A type to represent DCD _UCD.	N/A
DOWN_BURST_PROFILE	BITMAP(256)	A List of FEC Code Type for Downlink burst.	Refer to subclause 11.4.1 in IEEE 802.16Rev2/D0d [ref].
EIRP	INTEGER(1)	BS’s effective isotropic radiated power level.	Signed in units of 1 dBM.
FREQUENCY	INTEGER(4)	Downlink/Uplink center frequency.	

Table B-13—Data types for information elements

Type Name	Derived From	Definition	Valid Range
FREQUENCY	INTEGER(2)	Identifier the carrier frequency.	
FREQUENCY_CODE_NUMBER	INTEGER(2)	UMTS scrambling code, cdma2000 Walsh code)	0-65535
GAP	SEQUENCE(TTG, RTG)		
HANDOVER_CODE	INTEGER(1)	HANDOVER_RANGING_CODE.	Refer to subclause 11.3.1 in IEEE 802.16Rev2/D0d [ref].
INITIAL_CODE	INTEGER(1)	INITIAL_RANGING_CODE.	Refer to subclause 11.3.1 in IEEE 802.16Rev2/D0d [ref].
IP_MOBILITY_MGMT	BITMAP(16)	Indicates the supported mobility management protocols.	Bit 0: Mobile IPv4 Bit 1: Mobile IPv4 Regional Registration Bit 2: Mobile IPv6 Bit 3: Hierarchical Mobile IPv6 Bit 4: Low Latency Handoffs Bit 5: Fast Handovers for Mobile IPv6 Bit 6: IKEv2 Mobility and Multihoming Protocol Bit 7-15: (Reserved)
NETWORK_CAPABILITIES	BITMAP(32)	A type to represents a set of network capabilities.	Bit 0: Security Bit 1: QoS Bit 2: Internet Access Bit 3: Emergency Services Bit 4: MIH Capability Bit 5-31: (Reserved)
NETWORK_TYPE	SEQUENCE(LINK_TYPE, CHOICE(REVISION, NULL), CHOICE(TYPE_EXTENSION, NULL))	A type to represent a network type and its revision.	See Table B-14.
OPERATOR_ID	SEQUENCE(OPERATOR_NAME, OPERATOR_NAMESPACE)	A type to represent an operator identifier.	See below.

Table B-13—Data types for information elements

Type Name	Derived From	Definition	Valid Range
OPERATOR_NAME	OCTET_STRING	A type to represent a operator name. The value uniquely identifies the operator name within the scope of the OPERATOR_NAMESPACE.	The value is a non NULL terminated string whose length shall not exceed 253 octets.
OPERATOR_NAMESPACE	ENUMERATED	A type to represent a type of operator name.	0: GSM/UMTS 1: CDMA 2: REALM (as defined in [30]). 3: ITU-T/TSB 4-255: (Reserved)
PARAMETERS	CHOICE (DCD_UCD, SIB, SYSTEM_PARAMETERS)	A data type to represent system information depending on Network type.	DCD_UCD: IEEE 802.16 SIB: UMTS SYSTEM_PARAMETERS: cdma2000
PILOT_PN	INTEGER(2)	Pilot PN sequence offset index.	
RANGING_CDMA_CODES	SEQUENCE(INITIAL_CODE, HANDOVER_CODE)	A set of CDMA ranging codes.	
REVISION	BITMAP(64)	A type to represent a network type revision.	See Table B-14.
ROAMING_PARTNERS	LIST(OPERATOR_ID)	A type to represent a list of roaming partners.	N/A
RTG	INTEGER(1)	Receive transition gap.	Physical slot unit.
SERVICE_PROVIDER_ID	OCTET_STRING	A type to represent a service provider identifier.	A non-NULL terminated string whose length shall not exceed 253 octets.
SIB	SEQUENCE(CELL_ID, FREQUENCY_CODE_NUMBER)	A type to represent SIB	

Table B-13—Data types for information elements

Type Name	Derived From	Definition	Valid Range
SUPPORTED_LCP	ENUMERATED	A type represent supported Location Configuration Protocol. (LCP). LbyR: Location by Reference.	Values represent LCPs: 0: NULL 1: LLDP 2: LbyR with LLDP 3-10: (Reserved) 11: LLDP-MED 12: LbyR with LLDP-MED 13-20: (Reserved) 21: U-TDoA 22: D-TDoA 23-30: (Reserved) 31: DHCP 32: LbyR with DHCP 33-40: (Reserved) 41: OMA SUPL 42-50: (Reserved) 51: HELD 52: LbyR with HELD 53-255: (Reserved)
SYSTEM_INFORMATION	SEQUENCE (NETWORK_TYPE, MAC_ADDRESS, PARAMETERS)	A type to represent system information.	N/A
SYSTEM_PARAMETERS	SEQUENCE (BASE_ID, PILOT_PN, FREQUENCY, BAND_CLASS)	A type to represent SYSTEM_PARAMETERS.	
TTG	INTEGER(2)	Transmit transition gap.	Physical slot unit.
TYPE_EXTENSION	OCTET_STRING	A generic type extension contained indicating a flexible length and format field. The content is to be defined and filled by the appropriate SDO or service provider consortium, etc.	The value is a non NULL terminated string whose length shall not exceed 253 octets.
UP_BURST_PROFILE	BITMAP(256)	A List of FEC Code Type for Uplink burst.	Refer to subclause 11.3.1 in IEEE 802.16Rev2/D0d [ref].

Table B-14—Network type and revision representation

Network	Link Type	Revision
(*Reserved*)	0	N/A
Wireless - GSM	1	N/A

Table B-14—Network type and revision representation

Network	Link Type	Revision
Wireless - GPRS	2	N/A
Wireless - EDGE	3	N/A
Ethernet	15	N/A
Wireless - Other	18	N/A
Wireless - IEEE 802.11	19	(PHY/MAC Features) Bit 0: Access point (AP) Bit 1: Independent station (not an AP) Bit 2: FHSS PHY for 2.4GHz band Bit 3: DSSS PHY for 2.4GHz band Bit 4: IR PHY Bit 5: OFDM PHY for 5GHz band Bit 6: High-speed PHY Bit 7: Multi-domain operation capability implemented Bit 8: Extended Rate PHY (ERP) Bit 9: Spectrum management operation supported Bit 10: Regulatory class capability implemented Bit 11: QoS Supported Bit 12: RSN supported Bit 13-63: (Reserved)
Wireless - CDMA2000	22	N/A
Wireless - UMTS	23	Bit 0: Rel-99 Bit 1: Rel-4 Bit 2: Rel-5 (w/ HSDPA) Bit 3: Rel-6 (w/ HSUPA) Bit 4: Rel-7 (MIMO/OFDM) Bit 5: Rel-8 Bit 6-63: (Reserved)
Wireless - cdma2000-HRPD	24	Bit 0: Rev-0 Bit 1: Rev-A Bit 2: Rev-B Bit 3: Rev-C Bit 4-63: (Reserved)
Wireless - IEEE 802.16	27	Bit 0: IEEE 802.16-2001 Bit 1: IEEE 802.16c-2002 Bit 2: IEEE 802.16a-2003 Bit 3: IEEE 802.16-2004 Bit 4: IEEE 802.16e-2005 Bit 5: IEEE 802.16g-2007 Bit 6-63: (Reserved)
Wireless - IEEE 802.20	28	N/A
Wireless - IEEE 802.22	29	N/A

Note, the *Type* values in Table B-14 are deliberately made consistent with RADIUS network access server (NAS)-Port-Type definitions as specified by Internet Assigned Numbers Authority (IANA).

B.3.9 Data types for information service query

B.3.9.1 Binary representation

Table B-15—Data types for binary query

Type Name	Derived From	Definition	Valid Range
CURRENCY_PREFERENCE	COST_CURRENCY	A type to indicate currency preference.	N/A
IE_TYPE	UNSIGNED_INT(4)	A type to represent an IE type.	See Table A-1
INFO_QUERY_BINARY_DATA	SEQUENCE(CHOICE(QUERIER_LOCATION, NULL), CHOICE(NETWORK_TYPE_INCLUSION, NULL), CHOICE(NETWORK_INCLUSION, NULL), CHOICE(REPORTING_TEMPLATE, NULL), CHOICE(REPORT_LIMITATION, NULL), CHOICE(CURRENCY_PREFERENCE, NULL))	Represents a binary query. There should exist at least one of the query data type QUERIE_LOCATION, NETWORK_TYPE_INCLUSION, and NETWORK_INCLUSION. One CURRENCY_PREFERENCE at most is included in an Info Query Binary TLV. If included, it indicates to the MIIS server the preferred currency the returned cost should be represented in. If the MIIS server cannot return the cost in the specified currency, it may return the cost in other currencies.	N/A
NEIGHBORHOOD_RADIUS	UNSIGNED_INT(4)	The radius in meters from the center point of querier's location.	0 to 2^32-1
NETWORK_INCLUSION	LIST(ACCESS_NETWORK_ID)	A type to represent a list of network identifiers.	N/A

Table B-15—Data types for binary query

Type Name	Derived From	Definition	Valid Range
NETWORK_TYPE_INCLUSION	BITMAP(32)	A type to represent a set of link types.	The value is a 4 octet bitmap: Bit 0: Wireless - GSM Bit 1: Wireless - GPRS Bit 2: Wireless - EDGE Bit 3: IEEE 802.3 (Ethernet) Bit 4: Wireless - Other Bit 5: Wireless - IEEE 802.11 Bit 6: Wireless - CDMA2000 Bit 7: Wireless - UMTS Bit 8: Wireless - cdma2000-HRPD Bit 9: Wireless - IEEE 802.16 Bit 10: Wireless - IEEE 802.20 Bit 11: Wireless - IEEE 802.22 Bit 12-31: (*Reserved AND shall not be used and shall be always set to "0"*)
QUERIER_LOCATION	SEQUENCE(CHOICE(LOCATION, NULL), CHOICE(MAC_ADDRESS, NULL), CHOICE(NEIGHBORHOOD_RADIUS, NULL))	A type to represent a querier's location. It is not valid to use both LOCATION and MAC_ADDRESS at the same time.	N/A

Table B-15—Data types for binary query

Type Name	Derived From	Definition	Valid Range
REPORT_LIMITATION	SEQUENCE(UNSIGNED_INT(2), UNSIGNED_INT(2))	A type to represent a report limitation. The first UNSIGNED_INT(2) contains the maximum number of IEs in the INFO_RSP_BINARY_DATA. The second UNSIGNED_INT(2) contains the starting entry number (offset = 1 points to the first entry) from which a chunk of entries are to be included in the INFO_QUERY_BINARY_DATA. It is assumed that the IS server generates the same ordered list of entries for queries from the same IS client with the same INFO_RSP_BINARY_DATA content (except for REPORT_LIMITATION) before the limitation on the REPORT_LIMITATION is applied.	N/A
REPORTING_TEMPLATE	LIST(IE_TYPE)	A type to represent a list of IE types. Inclusion of any IE type is optional.	N/A.

B.3.9.2 RDF representation

Table B-16—Data type for RDF query

Type Name	Derived From	Definition	Valid Range
INFO_QUERY_RDF_SCHEMA	OCTET_STRING	A type to represent the URL of an RDF schema to obtain.	N/A

Table B-16—Data type for RDF query

Type Name	Derived From	Definition	Valid Range
INFO_QUERY_RDF_DATA	SEQUENCE(CHOICE(MIME_TYPE, NULL), OCTET_STRING)	Represents RDF query. If MIME_TYPE is omitted, MIME type “application/sparql-query” is used. Each OCTET_STRING is formatted with the MIME type.	N/A
MIME_TYPE	OCTET_STRING	Represents MIME type.	N/A

B.3.10 Data types for information service response

B.3.10.1 Binary representation

Table B-17—Data type for binary information query response

Type Name	Derived From	Definition	Valid Range
INFO_RSP_BINARY_DATA	LIST(INFORMATION_ELEMENT)	A type to represent a binary query response data.	N/A

B.3.10.2 RDF representation

Table B-18—Data type for RDF information query response

Type Name	Derived From	Definition	Valid Range
INFO_RSP_RDF_DATA	SEQUENCE(CHOICE(MIME_TYPE, NULL), OCTET_STRING)	Represents RDF data query result. If MIME_TYPE is omitted, MIME type “application/ sparql-results+xml” is used, the OCTET_STRING is formatted with the MIME type.	N/A
INFO_RSP_SCHEMA_URL	OCTET_STRING	An URL of an RDF schema.	N/A
INFO_RSP_RDF_SCHEMA	SEQUENCE{ CHOICE(MIME_TYPE, NULL), OCTET_STRING)	Represents an RDF schema. If MIME_TYPE is omitted, MIME type “application/xml” is used. The OCTET_STRING is formatted with the MIME type.	N/A

B.3.11 Data type for MIHF identification

Table B-19—Data type for MIHF identification

Type Name	Derived From	Definition	Valid Range
MIHF_ID	OCTET_STRING	The MIHF Identifier. MIHF-ID shall be invariant even if the device obtains different interfaces or other components in its registration lifetime. For example, MIHF-ID may be an FQDN or NAI of the sender of the registration request	N/A

B.3.12 Data type for MIH capabilities

Table B-20—Data type for MIH capabilities

Type Name	Derived From	Definition	Valid Range
EVENT_CONFIGURATION_INFO	SEQUENCE(MIH_EVENT_ID, CHOICE(LIST(LINK_DETECT_CONFIG), NULL)))	Represents additional configuration information for event subscription.	N/A
LINK_DETECT_CONFIG	SEQUENCE(CHOICE(ACCESS_NETWORK_ID, NULL), CHOICE(SIGNAL_STRENGTH, NULL))	A data type for configuring link detected event trigger.	N/A
MBB_HANDOVER_SUPPORT	SEQUENCE(NETWORK_TYPE, NETWORK_TYPE, BOOLEAN)	Indicates if make before break is supported FROM the first network type TO the second network type.	The BOOLEAN value assignment: True: Make before break is supported. False: Make before break is not supported.

Table B-20—Data type for MIH capabilities

Type Name	Derived From	Definition	Valid Range
MIH_COMMAND_LIST	BITMAP(32)	A set of MIH commands.	Bit 0: MIH_Get_Link_Parameters Bit 1: MIH_Link_Configure_Thresholds Bit 2: MIH_Link_Actions Bit 3: MIH_Net_HO_Candidate_Query MIH_Net_HO_Commit MIH_N2N_HO_Query_Resources MIH_N2N_HO_Commit MIH_N2N_HO_Complete Bit 4: MIH_MN_HO_Candidate_Query MIH_MN_HO_Commit MIH_MN_HO_Complete Bit 5-31: (Reserved)
MIH_EVENT_ID	ENUMERATED	MIH event identifiers.	0: MIH_Link_Up 1: MIH_Link_Down 2: MIH_Link_Going_Down 3: MIH_Link_Detected 4: MIH_Link_Parameters_Report 5: MIH_Link_Event_Rollback 6: MIH_Link_PDU_Transmit_Status 7: MIH_Link_Handover_Imminent 8: MIH_Link_Handover_Complete 9-255: (Reserved)
MIH_EVENT_LIST	BITMAP(32)	A set of MIH events.	Bit 0: MIH_Link_Up Bit 1: MIH_Link_Down Bit 2: MIH_Link_Going_Down Bit 3: MIH_Link_Detected Bit 4: MIH_Link_Parameters_Report Bit 5: MIH_Link_Event_Rollback Bit 6: MIH_Link_PDU_Transmit_Status Bit 7: MIH_Link_Handover_Imminent Bit 8: MIH_Link_Handover_Complete Bit 9-31: (Reserved)

Table B-20—Data type for MIH capabilities

Type Name	Derived From	Definition	Valid Range
MIH_IS_QUERY_TYPE_LIST	BITMAP(64)	A set of IS query types.	Bit 0: BINARY Bit 1: RDF_DATA Bit 2: RDF_SCHEMA_URL Bit 3: RDF_SCHEMA Bit 4: TYPE_IE_NETWORK_TYPE Bit 5: TYPE_IE_OPERATOR_IDENTIFIER Bit 6: TYPE_IE_SERVICE_PROVIDER_IDENTIFIER Bit 7: TYPE_IE_COUNTRY_CODE Bit 8: TYPE_IE_ACCESS_NETWORK_IDENTIFIER Bit 9: TYPE_IE_ROAMING_PARTNERS Bit 10: TYPE_IE_COST Bit 11: TYPE_IE_NETWORK_SECURITY Bit 12: TYPE_IE_NETWORK_QOS Bit 13: TYPE_IE_NETWORK_DATA_RATE Bit 14: TYPE_IE_NETWORK_CHANNEL_RANGE Bit 15: TYPE_IE_NETWORK_IP_CONFIG_METHODS Bit 16: TYPE_IE_NETWORK_CAPABILITIES Bit 17: TYPE_IE_LIST_SUPPORTED_LCP Bit 18: TYPE_IE_POA_MAC_ADDRESS Bit 19: TYPE_IE_POA_LOCATION Bit 20: TYPE_IE_POA_CHANNEL_RANGE Bit 21: TYPE_IE_POA_SYSTEM_INFORMATION Bit 22: TYPE_IE_POA_SUBNET_INFORMATION Bit 23: TYPE_IE_POA_IP_ADDRESS Bit 24- 63: (Reserved)
MIH_TRANSPORT_LIST	BITMAP(16)	List of supported transports.	Bit 0: UDP Bit 1: TCP Bit 2-15: (Reserved)
NETWORK_TYPE_AND_EVENTS	SEQUENCE(NETWORK_TYPE, MIH_EVENT_LIST)	Represent the supported events on a given network type.	N/A
NETWORK_TYPE_AND_MAC	SEQUENCE(NETWORK_TYPE, MAC_ADDRESS)	Represent an MAC address of a specific network type.	N/A

B.3.13 Data type for MIH registration

Table B-21—Data type for MIH registration

Type Name	Derived From	Definition	Valid Range
REG_REQUEST_CODE	ENUMERATED	The registration code.	0 - Registration 1 - Re-Registration

B.3.14 Data types for handover operation

Table B-22—Data type for handover operation

Type Name	Derived From	Definition	Valid Range
HANDOVER_TYPE	BOOLEAN	Represents the handover type.	TRUE: Intra-technology handover FALSE: Inter-technology handover
HANDOVER_STATUS	ENUMERATED	Represents the permission for handover.	0: HandoverPermitted 1: HandoverDeclined
HANDOVER_RESULT	ENUMERATED	Handover result.	0: Success 1: Failure 2: Rejected
TARGET_NETWORK_INFO	SEQUENCE (CHOICE (ACCESS_NETWORK_IDENTIFIER, NULL) CHOICE (ACCESS_NETWORK_AUX_ID, NULL) CHOICE (LINK_ADDRESS, NULL))	Represents the handover commit information. LINK_ADDRESS corresponds to the target PoA.	N/A

B.3.15 Data types for MIH_NET_SAP primitives

Table B-23—Data type for MIH_NET_SAP primitives

Type Name	Derived From	Definition	Valid Range
TRANSPORT_TYPE	ENUMERATED	The transport type supported.	0: L2 1: L3 or higher layer protocols

Annex C MIH protocol message code assignments

(normative)

Table C-1—AID assignment

MIH messages	AID
MIH messages for Service Management	
MIH_Capability_Discover	1
MIH_Register	2
MIH_DeRegister	3
MIH_Event_Subscribe	4
MIH_Event_Unsubscribe	5
MIH messages for Event Service	
MIH_Link_Up	1
MIH_Link_Down	2
MIH_Link_Going_Down	3
MIH_Link_Event_Rollback	4
MIH_Link_Detected	5
MIH_Link_Parameters_Report	6
MIH_Link_Handover_Imminent	7
MIH_Link_Handover_Complete	8
MIH messages for Command Service	
MIH_Get_Link_Parameters	1
MIH_Link_Configure_Thresholds	2
MIH_Link_Actions	3
MIH_Net_HO_Candidate_Query	4
MIH_MN_HO_Candidate_Query	5
MIH_N2N_HO_Query_Resources	6
MIH_Net_HO_Commit	7
MIH_N2N_HO_Commit	9
MIH_MN_HO_Complete	10
MIH_N2N_HO_Complete	11
MIH messages for Information Service	
MIH_Get_Information	1

Table C-2 provides the TLV type value assignment for MIH messages. The type value can be extracted from the binary encoding method of the corresponding data type. TLV type value 110-127 is reserved for experimental TLVs.

Table C-2—Type values for TLV encoding

TLV Type Name	TLV Type Value	Data Type
Source MIHF ID	1	MIHF_ID
Destination MIHF ID	2	MIHF_ID
Status	3	STATUS
List of Network Type and Events	4	LIST(NETWORK_TYPE_AND_EVENTS)
MIH Event List	5	MIH_EVENT_LIST
MIH Command List	6	MIH_COMMAND_LIST
MIIS Query Type List	7	MIH_IS_QUERY_TYPE_LIST
Transport Option List	8	MIH_TRANSPORT_LIST
Link MACs	9	LIST(NETWORK_TYPE_AND_MAC)
MBB Handover Support	10	LIST(MBB_HANDOVER_SUPPORT)
Register Request Code	11	REG_REQUEST_CODE
Valid Time Interval	12	UNSIGNED_INT(4)
Link Identifier	13	LINK_TUPLE_ID
New Link Identifier	14	LINK_TUPLE_ID
Old Access Router	15	LINK_ADDRESS
New Access Router	16	LINK_ADDRESS
IP Renewal Flag	17	IP_RENEWAL_FLAG
Mobility Management Support	18	IP_MOBILITY_MGMT
IP Address Configuration Methods	19	IP_CONFIG_METHODS
Link Down Reason Code	20	LINK_DOWN_REASON
Time Interval	21	UNSIGNED_INT(2)
Link Going Down Reason	22	LINK_GOING_DOWN_REASON
Unique Event Identifier	23	UNSIGNED_INT(2)
MIH Capability Flag	24	LINK_MIH_CAPABILITY_FLAG
Link Parameter Report List	25	LIST(LINK_PARAM_REPORT)
Handover Type	26	HANDOVER_TYPE
Device States Request	27	DEVICE_STATES_REQ
Link Identifier List	28	LIST(LINK_ID)
Device States Response	29	DEVICE_STATES_RSP
Operation Mode	30	OPERATION_MODE

Table C-2—Type values for TLV encoding

TLV Type Name	TLV Type Value	Data Type
Operation Mode Status	31	OPERATION_MODE_STATUS
Get Status Request Set	32	LINK_STATUS_REQ
Get Status Response List	33	LIST(LINK_STATUS_RSP)
Configure Request List	34	LIST(LINK_CONFIG_PARAM)
Configure Response List	35	LIST(LINK_CONFIG_STATUS)
Scan Link Identifier	36	LINK_ID
Scan Response Set	37	LIST(LINK_SCAN_RSP)
List of Link PoA List	38	LIST(LINK_POA_LIST)
Link Action	39	LINK_ACTION
Handover Resource Response Set	40	QOS_LIST
Handover Resource Query List	41	QOS_LIST
Handover Status	42	HANDOVER_STATUS
Access Router Address	43	IP_ADDRESS
DHCP Server Address	44	IP_ADDRESS
FA Address	45	IP_ADDRESS
Access Router Address List	46	LIST(IP_ADDRESS)
DHCP Server Address List	47	LIST(IP_ADDRESS)
FA Address List	48	LIST(IP_ADDRESS)
IP Address Information Status	49	IP_CONFIG_STATUS
Link Action List	50	LIST(LINK_ACTION_REQ)
Link Actions Response	51	LIST(LINK_ACTION_RSP)
Handover Result	52	HANDOVER_RESULT
Resource Status	53	LINK_RESOURCE_STATUS
Available Resource Set	54	LIST(QOS_LIST)
Resource Retention Status	55	LINK_RESOURCE_RETENTION_STATUS
Info Query Binary Data List	56	LIST(INFO_RSP_BINARY_DATA)
Info Query RDF Data List	57	LIST(INFO_QUERY_RDF_DATA)
Info Query RDF Schema URL	58	NULL
Info Query RDF Schema List	59	LIST(INFO_QUERY_RDF_SCHEMA)
Max Response Size	60	UNSIGNED_INT(2)
Info Response Binary Data list	61	LIST(INFO_RSP_BINARY_DATA)
Info Response RDF Data List	62	LIST(INFO_RSP_RDF_DATA)
Info Response RDF Schema URL List	63	LIST(INFO_RSP_SCHEMA_URL)

Table C-2—Type values for TLV encoding

TLV Type Name	TLV Type Value	Data Type
Info Response RDF Schema List	64	LIST(INFO_RSP_RDF_SCHEMA)
Mobile Node MIHF ID	65	MIHF_ID
Query Resource Report Flag	66	BOOLEAN
Target Link Action	67	LINK_ACTION
Link Action Result	68	LINK_AC_RESULT_CODE
Target Link Action Result	69	LINK_AC_RESULT_CODE
Event Configuration Info List	70	LIST(EVENT_CONFIG_INFO)
Target Network Info	71	TARGET_NETWORK_INFO
List of Target Network Info	72	LIST(TARGET_NETWORK_INFO)

Annex D MIIS basic schema

(normative)

The remaining text in this section defines the RDF vocabularies for MIIS.

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
 <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
 <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
 <!ENTITY mihbasic "URL_TO_BE_ASSIGNED">
 <!ENTITY owl "http://www.w3.org/2002/07/owl#">
 <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
]>

<rdf:RDF xmlns:rdf="&rdf;" xmlns:rdfs="&rdfs;"
xmlns:mihbasic="&mihbasic;" xml:base="&mihbasic;"
xmlns:owl="&owl;" xmlns:xsd="&xsd;">

<owl:Ontology rdf:about="">
 <rdfs:label>
  Basic Schema for IEEE 802.21 Information Service
 </rdfs:label>
 <owl:versionInfo>1.0</owl:versionInfo>
</owl:Ontology>

<owl:DatatypeProperty rdf:ID="ie_type_identifier">
 <rdfs:subPropertyOf rdf:resource="&rdfs;label"/>
 <rdfs:range rdf:resource="&xsd;hexBinary"/>
 <rdfs:comment>
  A type identifier values for Information Elements.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_container_list_of_network">
 <mihbasic:ie_type_identifier>10000300</mihbasic:ie_type_identifier>
 <rdfs:range rdf:resource="#LIST_OF_NETWORKS"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="LIST_OF_NETWORKS">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#type_ie_container_network"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:ObjectProperty rdf:ID="type_ie_container_network">
 <mihbasic:ie_type_identifier>10000301</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#LIST_OF_NETWORKS"/>
 <rdfs:range rdf:resource="#NETWORK"/>
 <rdfs:comment>
  This class contains General Information depicting and Access
  Network Specific Information.
 </rdfs:comment>
```

```
</owl:ObjectProperty>

<owl:Class rdf:ID="NETWORK">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#type_ie_network_type"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#type_ie_operator_identifier"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#type_ie_container_poa"/>
   <owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:minCardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:ObjectProperty rdf:ID="type_ie_network_type">
 <mihbasic:ie_type_identifier>0x10000000</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="#NETWORK_TYPE"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="NETWORK_TYPE">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#link_type"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="link_type">
 <rdfs:domain rdf:resource="#NETWORK_TYPE"/>
 <rdfs:range rdf:resource="&xsd;unsignedInt"/>
<rdfs:comment>
 Link type of a network. The following values are assigned:
 1: Wireless - GSM
 2: Wireless - GPRS
 3: Wireless - EDGE
 15: Ethernet
 18: Wireless - Other
 19: Wireless - IEEE 802.11
 22: Wireless - CDMA2000
 23: Wireless - UMTS
 24: Wireless - cdma-2000-HRPD
 27: Wireless - IEEE 802.16
 28: Wireless - IEEE 802.20
 29: Wireless - IEEE 802.22
```

```
</rdfs:comment>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="revision">
 <rdfs:subPropertyOf rdf:resource="#bit_number"/>
 <rdfs:domain rdf:resource="#NETWORK_TYPE"/>
 <rdfs:comment>
  The range of #bit_number is 0-63.
 </rdfs:comment>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="type_ie_operator_identifier">
 <mihbasic:ie_type_identifier>0x10000001</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="#OPERATOR_ID"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="OPERATOR_ID">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#operator_namespace"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#operator_name"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="operator_namespace">
 <rdfs:domain rdf:resource="#OPERATOR_ID"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  A value of Operator Type:
  0: GSM/UMTS
  1: CDMA
  2: REALM
  3: ITU-T/TSB
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="operator_name">
 <rdfs:domain rdf:resource="#OPERATOR_ID"/>
 <rdfs:range rdf:resource="&xsd;string"/>
 <rdfs:comment>
  The value is a non NULL terminated
  string whose length shall not exceed 253 octets.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="type_ie_service_provider_identifier">
 <mihbasic:ie_type_identifier>0x10000002</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="&xsd;string"/>
```

```
 <rdfs:comment>
  A non-NULL terminated string whose length shall not exceed 253 octets.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="type_ie_access_network_identifier">
 <mihbasic:ie_type_identifier>0x10000100</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="&xsd;string"/>
 <rdfs:comment>
  A non-NULL terminated string whose length shall not exceed 253 octets.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="type_ie_access_network_aux_id">
 <mihbasic:ie_type_identifier>0x10000101</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="&xsd;string"/>
 <rdfs:comment>
  It is SSID if network type is IEEE 802.11.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_roaming_partner">
 <mihbasic:ie_type_identifier>0x10000102</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="#OPERATOR_ID"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="type_ie_cost">
 <mihbasic:ie_type_identifier>0x10000103</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="#COST"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="COST">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#cost_unit"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#cost_value"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#cost_currency"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>
```

```
<owl:DatatypeProperty rdf:ID="cost_unit">
 <rdfs:domain rdf:resource="#COST"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  The unit of the cost:
  0: second
  1: minute
  2: hours
  3: day
  4: week
  5: month
  6: year
  7: free
  8: flat rate
  9-255: Reserved
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cost_value">
 <rdfs:domain rdf:resource="#COST"/>
 <rdfs:range rdf:resource="&xsd;double"/>
 <rdfs:comment>
  The cost value in Currency/Unit
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cost_currency">
 <rdfs:domain rdf:resource="#COST"/>
 <rdfs:range rdf:resource="&xsd;string"/>
 <rdfs:comment>
  A three-letter currency code(e.g. "USD") specified by
  ISO 4217.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="bit_number">
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  This property represents a bit number that has
  the value as true.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_network_qos">
 <mihbasic:ie_type_identifier>0x10000106</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="#QOS_LIST"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="QOS_LIST">
</owl:Class>

<owl:DatatypeProperty rdf:ID="num_qos_types">
 <rdfs:domain rdf:resource="#QOS_LIST"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="throughput">
 <rdfs:domain rdf:resource="#QOS_LIST"/>
```

```
 <rdfs:range rdf:resource="&xsd;unsignedInt"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="packet_error_rate">
 <rdfs:domain rdf:resource="#QOS_LIST"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
</owl:DatatypeProperty>

<owl:Class rdf:ID="COS">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#cos_id"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#cos_value"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="cos_id">
 <rdfs:domain rdf:resource="#COS"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
 <rdfs:comment>
  A type to represent a class of service identifier.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cos_value">
 <rdfs:domain rdf:resource="#COS"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="cos_min_packet_transfer_delay">
 <rdfs:domain rdf:resource="#QOS_LIST"/>
 <rdfs:range rdf:resource="#COS"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="cos_avg_packet_transfer_delay">
 <rdfs:domain rdf:resource="#QOS_LIST"/>
 <rdfs:range rdf:resource="#COS"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="cos_max_packet_transfer_delay">
 <rdfs:domain rdf:resource="#QOS_LIST"/>
 <rdfs:range rdf:resource="#COS"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="cos_packet_transfer_delay_jitter">
 <rdfs:domain rdf:resource="#QOS_LIST"/>
 <rdfs:range rdf:resource="#COS"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="cos_packet_loss_rate">
```

```
 <rdfs:domain rdf:resource="#QOS_LIST"/>
 <rdfs:range rdf:resource="#COS"/>
</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="type_ie_network_data_rate">
 <mihbasic:ie_type_identifier>0x10000107</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="&xsd;unsignedInt"/>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_network_channel_range">
 <mihbasic:ie_type_identifier>0x????????</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="#CHANNEL_RANGE"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="type_ie_network_ip_config_methods">
 <mihbasic:ie_type_identifier>0x10000108</mihbasic:ie_type_identifier>
 <rdfs:subPropertyOf rdf:resource="#bit_number"/>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:comment>
  The range of #bit_number is 0-31.
 </rdfs:comment>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="type_ie_network_capabilities">
 <mihbasic:ie_type_identifier>0x10000109</mihbasic:ie_type_identifier>
 <rdfs:subPropertyOf rdf:resource="#bit_number"/>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:comment>
  The range of #bit_number is 0-31.
 </rdfs:comment>
</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="type_ie_supported_lcp">
 <mihbasic:ie_type_identifier>0x1000010A</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="type_ie_country_code">
 <mihbasic:ie_type_identifier>0x????????</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="&xsd;string"/>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_container_poa">
 <mihbasic:ie_type_identifier>0x10000302</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#NETWORK"/>
 <rdfs:range rdf:resource="#POA"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="POA">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#type_ie_poa_mac_address"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
```

```
  </rdfs:subClassOf>
  <rdfs:subClassOf>
   <owl:Restriction>
    <owl:onProperty rdf:resource="#type_ie_poa_location"/>
    <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
    </owl:cardinality>
   </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
   <owl:Restriction>
    <owl:onProperty rdf:resource="#type_ie_poa_channel_range"/>
    <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
    </owl:cardinality>
   </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
   <owl:Restriction>
    <owl:onProperty rdf:resource="#type_ie_poa_system_information"/>
    <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
    </owl:cardinality>
   </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
   <owl:Restriction>
    <owl:onProperty rdf:resource="#type_ie_poa_subnet_information"/>
    <owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1
    </owl:minCardinality>
   </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:comment>
   This class contains all the information depicting a PoA.
  </rdfs:comment>
 </owl:Class>

 <owl:ObjectProperty rdf:ID="type_ie_poa_mac_address">
  <mihbasic:ie_type_identifier>0x10000302</mihbasic:ie_type_identifier>
  <rdfs:domain rdf:resource="#POA"/>
  <rdfs:range rdf:resource="#TRANSPORT_ADDRESS"/>
 </owl:ObjectProperty>

 <owl:ObjectProperty rdf:ID="type_ie_poa_location">
  <mihbasic:ie_type_identifier>0x10000201</mihbasic:ie_type_identifier>
  <rdfs:domain rdf:resource="#POA"/>
  <rdfs:range rdf:resource="#LOCATION"/>
 </owl:ObjectProperty>

 <owl:Class rdf:ID="LOCATION">
 </owl:Class>

 <owl:Class rdf:ID="BINARY_GEO_LOCATION">
  <rdfs:subClassOf rdf:resource="#LOCATION"/>
  <rdfs:comment>
   This class has properties that represent geographic coordinate.
   The format is based on the Location Configuration Information (LCI)
   defined in RFC 3825.
  </rdfs:comment>
 </owl:Class>

 <owl:DatatypeProperty rdf:ID="la_res">
```

```
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="latitude">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;double"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="lo_res">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="longitude">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;double"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="altitude_type">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  Following codes are defined:
  1: Meters: in 2s-complement fixed-point 22-bit integer part with
  8-bit fraction. If AT = 1, an AltRes value 0.0 would indicate
  unknown altitude. The most precise Altitude would have an AltRes
  value of 30. Many values of AltRes would obscure any variation
  due to vertical datum differences.
  2: Floors: in 2s-complement fixed-point 22-bit integer part with
  8-bit fraction. AT = 2 for Floors enables representing altitude in
  a form more relevant in buildings which have different
  floor-to-floor dimensions.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="alt_res">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  Altitude resolution. 6 bits indicating the number of valid bits
  in the altitude. Values above 30 (decimal) are undefined and
  reserved.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="altitude">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;double"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="datum">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  Following codes are defined:
  1: WGS
  2: NAD 83 (with associated vertical datum for North American
  vertical datum for 1998)
```

```
  3: NAD 83 (with associated vertical datum for Mean Lower Low Water
  (MLLW))
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="geo_lo_method">
 <rdfs:domain rdf:resource="#BINARY_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  Way location information was derived or discovered:
  0: GPS
  1: Assisted GPS
  2: Manual
  3: Provided by DHCP
  4: Triangulation
  5-255: Reserved
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:Class rdf:ID="XML_GEO_LOCATION">
 <rdfs:subClassOf rdf:resource="#LOCATION"/>
</owl:Class>

<owl:DatatypeProperty rdf:ID="xml_geo_location">
 <rdfs:domain rdf:resource="#XML_GEO_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;String"/>
 <rdfs:comment>
  Geo address elements as described in RFC4119.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:Class rdf:ID="BINARY_CIVIC_LOCATION">
 <rdfs:subClassOf rdf:resource="#LOCATION"/>
 <rdfs:comment>
  This class has properties that represent civic address.
  The format is defined in IETF RFC 4676.
 </rdfs:comment>
</owl:Class>

<owl:DatatypeProperty rdf:ID="civic_country_code">
 <rdfs:domain rdf:resource="#BINARY_CIVIC_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;string"/>
 <rdfs:comment>
  Two-letter ISO 3166 country code in capital ASCII letters.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="civic_addr_elem">
 <rdfs:domain rdf:resource="#BINARY_CIVIC_LOCATION"/>
 <rdfs:range rdf:resource="#CIVIC_ADDR_ELEM"/>
 <rdfs:comment>
  This property contains the civic address elements.
  The format of the civic address elements is described
  in Section 3.4 of IETF RFC 4676 with a TLV pair
  (whereby the Type and Length fields are one octet long).
 </rdfs:comment>
</owl:ObjectProperty>

<owl:Class rdf:ID="CIVIC_ADDR_ELEM">
```

```
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#catype"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#cavalue"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="catype">
 <rdfs:domain rdf:resource="#CIVIC_ADDR_ELEM"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  A one-octet descriptor of the data civic address value.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cavalue">
 <rdfs:domain rdf:resource="#CIVIC_ADDR_ELEM"/>
 <rdfs:range rdf:resource="&xsd;string"/>
 <rdfs:comment>
  The civic address value.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:Class rdf:ID="XML_CIVIC_LOCATION">
 <rdfs:subClassOf rdf:resource="#LOCATION"/>
</owl:Class>

<owl:DatatypeProperty rdf:ID="xml_civic_location">
 <rdfs:domain rdf:resource="#XML_CIVIC_LOCATION"/>
 <rdfs:range rdf:resource="&xsd;String"/>
 <rdfs:comment>
  Geo address elements as described in RFC4119.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_poa_channel_range">
 <mihbasic:ie_type_identifier>0x10000202</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#POA"/>
 <rdfs:range rdf:resource="#CHANNEL_RANGE"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="CHANNEL_RANGE">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#low_channel_range"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
```

```
    <owl:Restriction>
     <owl:onProperty rdf:resource="#high_channel_range"/>
     <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
     </owl:cardinality>
    </owl:Restriction>
   </rdfs:subClassOf>
  </owl:Class>

  <owl:DatatypeProperty rdf:ID="low_channel_range">
   <rdfs:domain rdf:resource="#CHANNEL_RANGE"/>
   <rdfs:range rdf:resource="&xsd;unsignedInt"/>
   <rdfs:comment>
    Lowest channel frequency in MHz
   </rdfs:comment>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:ID="high_channel_range">
   <rdfs:domain rdf:resource="#CHANNEL_RANGE"/>
   <rdfs:range rdf:resource="&xsd;unsignedInt"/>
   <rdfs:comment>
    Highest channel frequency in MHz
   </rdfs:comment>
  </owl:DatatypeProperty>

  <owl:ObjectProperty rdf:ID="type_ie_poa_system_information">
   <mihbasic:ie_type_identifier>0x10000203</mihbasic:ie_type_identifier>
   <rdfs:domain rdf:resource="#POA"/>
   <rdfs:range rdf:resource="#SYSTEM_INFORMATION"/>
  </owl:ObjectProperty>

  <owl:Class rdf:ID="SYSTEM_INFORMATION">
  </owl:Class>

  <owl:Class rdf:ID="DCD_UCD">
   <rdfs:subClassOf rdf:resource="#SYSTEM_INFORMATION"/>
  </owl:Class>

  <owl:DatatypeProperty rdf:ID="bs_id">
   <rdfs:domain rdf:resource="#DCD_UCD"/>
   <rdfs:range rdf:resource="#TRANSPORT_ADDRESS"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:ID="bandwidth">
   <rdfs:domain rdf:resource="#DCD_UCD"/>
   <rdfs:range rdf:resource="&xsd;unsignedByte"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:ID="dcd_ucd_frequency">
   <rdfs:domain rdf:resource="#DCD_UCD"/>
   <rdfs:range rdf:resource="&xsd;unsignedInt"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:ID="eirp">
   <rdfs:domain rdf:resource="#DCD_UCD"/>
   <rdfs:range rdf:resource="&xsd;unsignedByte"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:ID="ttg">
   <rdfs:domain rdf:resource="#DCD_UCD"/>
```

```xml
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="rtg">
 <rdfs:domain rdf:resource="#DCD_UCD"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="down_burst_profile">
 <rdfs:subPropertyOf rdf:resource="#bit_number"/>
 <rdfs:domain rdf:resource="#DCD_UCD"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="up_burst_profile">
 <rdfs:subPropertyOf rdf:resource="#bit_number"/>
 <rdfs:domain rdf:resource="#DCD_UCD"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="initial_code">
 <rdfs:domain rdf:resource="#DCD_UCD"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="handover_code">
 <rdfs:domain rdf:resource="#DCD_UCD"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:Class rdf:ID="SIB">
 <rdfs:subClassOf rdf:resource="#SYSTEM_INFORMATION"/>
</owl:Class>

<owl:DatatypeProperty rdf:ID="cell_id">
 <rdfs:domain rdf:resource="#SIB"/>
 <rdfs:range rdf:resource="&xsd;hexBinary"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="frequency_code_number">
 <rdfs:domain rdf:resource="#SIB"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
</owl:DatatypeProperty>

<owl:Class rdf:ID="SYSTEM_PARAMETERS">
 <rdfs:subClassOf rdf:resource="#SYSTEM_INFORMATION"/>
</owl:Class>

<owl:DatatypeProperty rdf:ID="base_id">
 <rdfs:domain rdf:resource="#SYSTEM_PARAMETERS"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="pilot_pn">
 <rdfs:domain rdf:resource="#SYSTEM_PARAMETERS"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="frequency">
 <rdfs:domain rdf:resource="#SYSTEM_PARAMETERS"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
```

```
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="band_class">
 <rdfs:domain rdf:resource="#SYSTEM_PARAMETERS"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_poa_subnet_information">
 <mihbasic:ie_type_identifier>0x10000204</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#POA"/>
 <rdfs:range rdf:resource="#IP_SUBNET_INFO"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="IP_SUBNET_INFO">
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#subnet_address"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#ip_prefix_len"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:ObjectProperty rdf:ID="subnet_address">
 <rdfs:domain rdf:resource="#IP_SUBNET_INFO"/>
 <rdfs:range rdf:resource="#TRANSPORT_ADDRESS"/>
 <rdfs:comment>
  An IP address of the PoA encoded as Address base type defined in
  RFC3588. The first 2-octet contains AddressType, which may be
  either 1 (IPv4) or 2 (IPv6). If AddressType==1, the subnet_address
  property contains a 4-octet IPv4 address. If AddressType==2, the
  subnet_address property contains a 16-octet IPv6 address.
 </rdfs:comment>
</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="ip_prefix_len">
 <rdfs:domain rdf:resource="#IP_SUBNET_INFO"/>
 <rdfs:range rdf:resource="&xsd;unsignedByte"/>
 <rdfs:comment>
  The bit length of the prefix of the subnet to which subnet_address
  property belongs. The prefix_length is less than or equal to 32
  for IPv4 subnet and less than or equal to 128 for IPv6 subnet.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="type_ie_poa_ip_address">
 <mihbasic:ie_type_identifier>0x10000205</mihbasic:ie_type_identifier>
 <rdfs:domain rdf:resource="#POA"/>
 <rdfs:range rdf:resource="#TRANSPORT_ADDRESS"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="TRANSPORT_ADDRESS">
```

```
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#address_type"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#address_value"/>
   <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
   </owl:cardinality>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="address_type">
 <rdfs:domain rdf:resource="#TRANSPORT_ADDRESS"/>
 <rdfs:range rdf:resource="&xsd;unsignedShort"/>
 <rdfs:comment>
  An Address Family defined in
  http://www.iana.org/assignments/address-family-numbers.
 </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="address_value">
 <rdfs:domain rdf:resource="#TRANSPORT_ADDRESS"/>
 <rdfs:range rdf:resource="&xsd;hexBinary"/>
 <rdfs:comment>
  An address value specific to address_type.
 </rdfs:comment>
</owl:DatatypeProperty>

</rdf:RDF>
```

Annex E Bibliography

(informative)

[B1] 3GPP TR 43.901, Feasibility Study on Generic Access to A/Gb Interface.

[B2] 3GPP TS 23.002, Network Architecture.

[B3] 3GPP TS 23.060, General Packet Radio Service (GPRS); Service Description; Stage 2.

[B4] 3GPP TS 23.234, 3GPP system to Wireless Local Area Network (WLAN) inter-working; System description.

[B5] 3GPP TS 24.007, Mobile Radio Interface Signaling, Layer 3, General Aspects.

[B6] 3GPP TS 25.331, Radio Resource Control Specification.

[B7] 3GPP2 C.S0001-D (2004-02), Introduction to cdma2000 Standards for Spread Spectrum Systems.

[B8] 3GPP2 C.S0002-D (2004-02), Physical Layer Standard for cdma2000 Spread Spectrum Systems.

[B9] 3GPP2 C.S0003-D (2004-02), Medium Access Control (MAC) Standard for cdma2000 Spread Spectrum Systems.

[B10]3GPP2 C.S0005-D (2004-02), Upper Layer (Layer 3) Signaling Standard for cdma2000 Spread Spectrum Systems.

[B11]3GPP2 C.S0006-D (2004-02), Analog Signaling Standard for cdma2000 Spread Spectrum Systems.

[B12]3GPP2 C.S0024-A (V. 2.0, 2005-07), cdma2000 High Rate Packet Data Air Interface Specification.

[B13]3GPP2 C.S0063-A (V. 1.0, 2006-04), cdma2000 High Rate Packet Data Supplemental Services.

[B14]3GPP2 S.R0087, cdma2000 – WLAN Interworking.

[B15]3GPP2 X.S0028-100 (V. 1.0, 2006-08), Wireless Local Area Network (WLAN) Interworking, Access to internet.

[B16]3GPP2 X.S0028-200 (V. 1.0), Wireless Local Area Network (WLAN) Interworking, Access to Operator Services and Mobility.

[B17]IEEE 802.1ag™ (Draft D7.1, 2006-11), Draft Standard for Local and Metropolitan Area Networks — Virtual Bridged Local Area Networks —Amendment 5: Connectivity Fault Management.

[B18]IEEE P802.21™ (Document 21-04-0087-12-0000, 2004-09), Media Independent Handover Service - Draft Technical Requirements.

[B19]IEEE Std 802.3ah™-2004, (Amendment to IEEE 802.3) for EFM Operation, Administration and Maintenance.

[B20]IEEE Std 802.11™-2007, Information Technology - Telecommunications and information exchange between system-Local and metropolitan area networks-Specific Requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

[B21]IEEE P802.16g/D9-2007, Draft IEEE Standard for Local and metropolitan area networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Management Plane Procedures and Services.

[B22]IEEE Std 802.3™-2002, Information Technology- Telecommunications and information exchange between system-Local and metropolitan area networks-Specific Requirements-Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications.

[B23]IETF RFC 2988 (2000-11), Computing TCP's Retransmission Timer.

[B24]IETF RFC 3753 (2004-06), Mobility Related Terminology.

[B25]IETF RFC 4443 (2006-03), Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification.

[B26]IETF Internet Draft (draft-ietf-geopriv-radius-lo-10.txt, 2006-09), Carrying Location Objects in RADIUS.

[B27]ITU-T Recommendation X.690, Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).

[B28]ITU-T Recommendation Y.1540, Internet protocol data communication service - IP packet transfer and availability performance parameters.

[B29]ITU-T Recommendation Y.1541, Network performance objectives for IP-based services.

[B30]W3C Recommendation, OWL Web Ontology Language Reference.

[B31]W3C Recommendation, RDF Vocabulary Description Language 1.0: RDF Schema.

[B32]W3C Recommendation, SPARQL Query Results XML Format.

Annex F Example UML representation of basic schema

(informative)

Figures F-1 and F-2 show an example UML representation of the IEEE 802.21 MIIS basic schema. A dotted arrow indicates range of a property. An attribute indicates a property of a class.

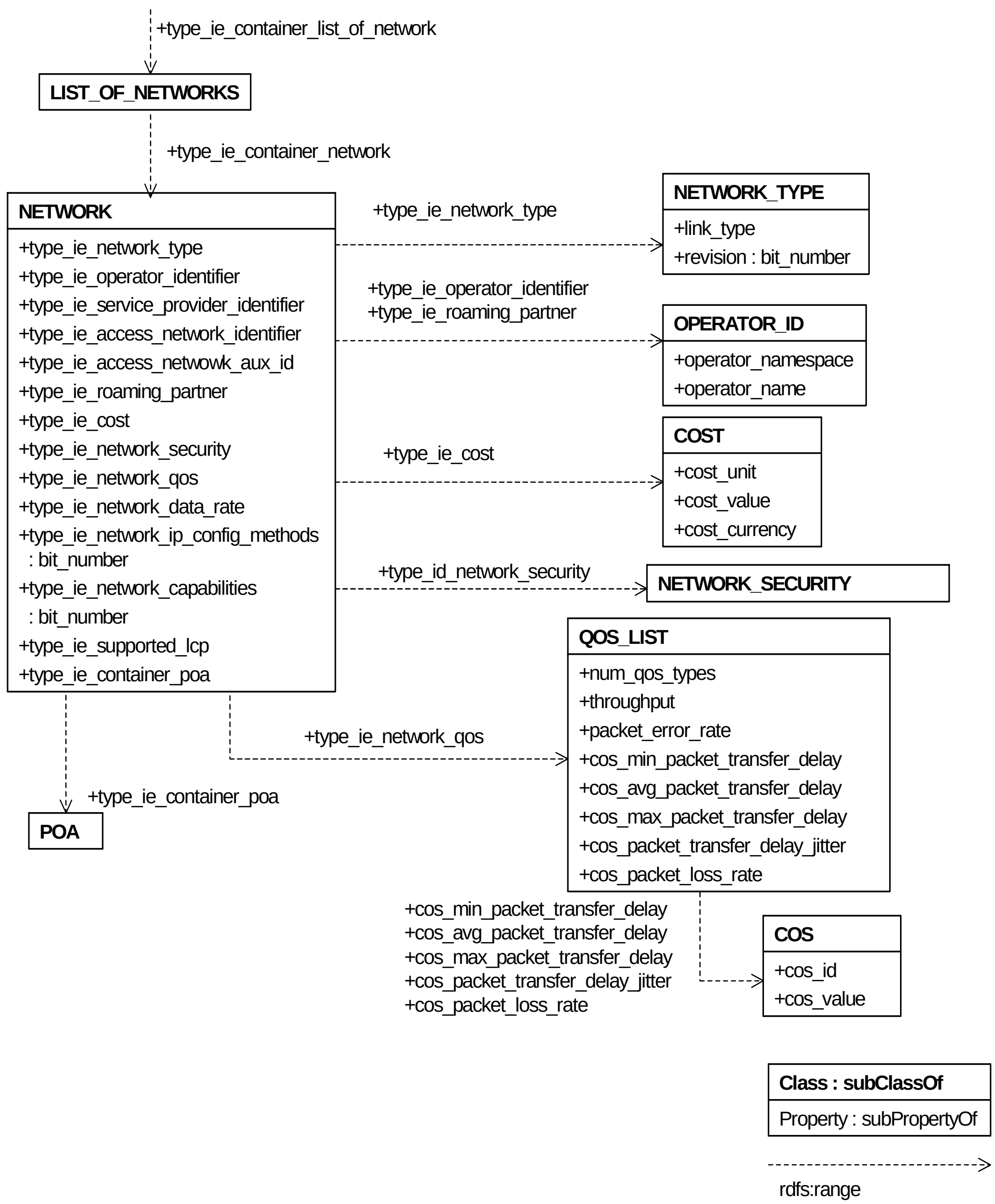


Figure F-1—UML representation of basic schema

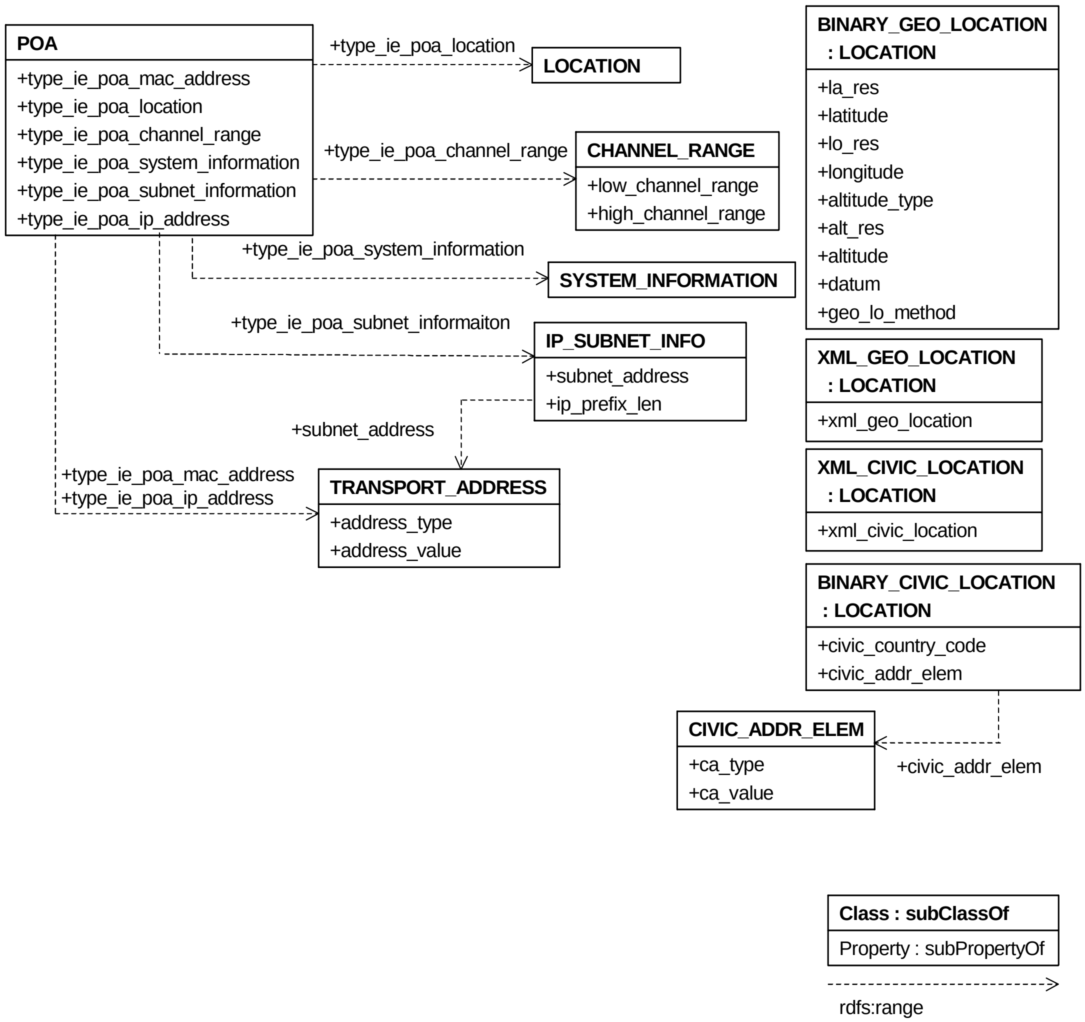


Figure F-2—UML representation of basic schema (cont.)

Annex G Making User Extensions to MIIS schema

(informative)

In this subclause, how to create an extended schema is described. How to create “IP address of Mobile IP home agent” is used as an example.

It is possible to support an Extended Schema without defining additional IEs or TLVs (including vendor specific ones) other than that are defined in this specification. An extended schema can be defined as an XML document. An example Extended Schema definition is shown as follows.

```
<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [
 <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
 <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
 <!ENTITY mihbasic "URL_TO_BE_ASSIGNED">
 <!ENTITY mihextended "http://www.informationservice.org/2006/08/extended-schema#">
 <!ENTITY owl "http://www.w3.org/2002/07/owl#">
 <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
]>

<rdf:RDF xmlns:rdf="&rdf;" xmlns:rdfs="&rdfs;" xmlns:mihbasic="&mihbasic;"
xmlns:mihextended="&mihextended;" xml:base="&mihextended;" xmlns:owl="&owl;" xmlns:xsd="&xsd;">

 <owl:Ontology rdf:about="">
  <rdfs:label>Extended Schema</rdfs:label>
 </owl:Ontology>

<owl:ObjectProperty rdf:ID="ha_address">
  <rdfs:domain rdf:resource="&mihbasic;NETWORK"/>
  <rdfs:range rdf:resource="&mihbasic;TRANSPORT_ADDRESS"/>
 </owl:ObjectProperty>

</rdf:RDF>
```

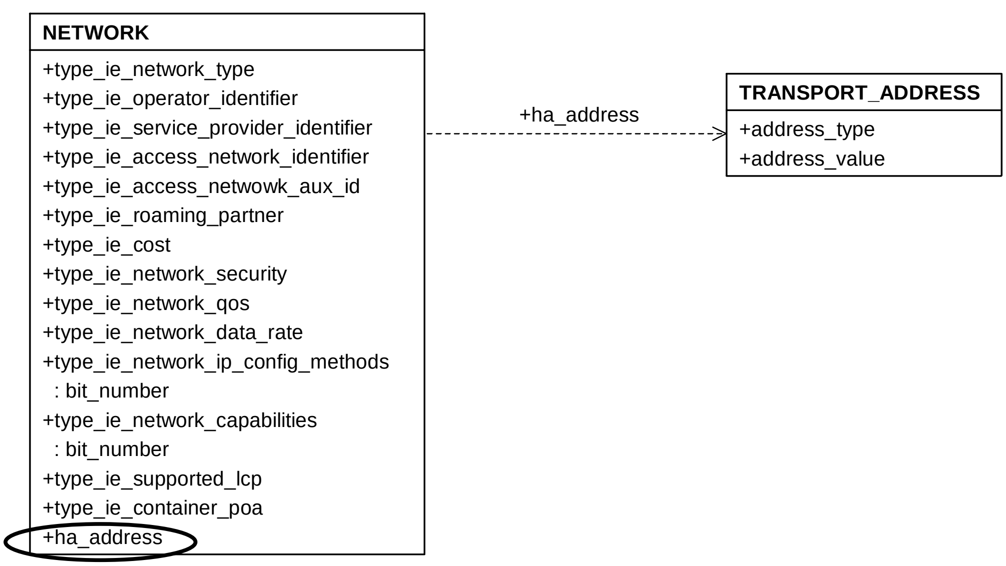


Figure G-1—Extended schema

In this example, the new property of “NETWORK” class, “ha_address” is created in the Extended Schema where “ha_address” has the same format as “TRANSPORT_ADDRESS” class.

Annex H Handover procedures

(informative)

H.1 Mobile-initiated handover procedure

The Mobile-initiated Handover Procedure operates as follows:

1) Mobile Node is connected to the serving network via Current PoS and it has access to MIH Information Server.

2) Mobile Node queries information about neighboring networks by sending the MIH_Get_Information Request to Information Server. Information Server responds with MIH_Get_Information Response. This information query may be attempted as soon as Mobile Node is first attached to the network.

3) Mobile Node triggers a mobile-initiated handover by sending MIH_MN_HO_Candidate_Query Request to Serving PoS. This request contains the information of potential candidate networks.

4) Serving PoS queries the availability of resources at the candidate networks by sending MIH_N2N_HO_Query_Resources Request to one or multiple Candidate PoSs.

5) Candidate PoSs respond with MIH_N2N_HO_Query_Resources Response and Serving PoS notifies the Mobile Node of the resulting resource availability at the candidate networks through MIH_MN_HO_Candidate_Query Response.

6) Mobile Node decides the target of the handover and requests resource preparation by sending the MIH_Link_Handover_Imminent Indication to Serving PoS.

7) Serving PoS sends MIH_N2N_HO_Commit Request to Target PoS to request resource preparation at the target network. Target PoS responds the result of the resource preparation by MIH_N2N_HO_Commit Response.

8) New layer 2 connection is established and a certain mobility management protocol procedures are carried out between Mobile Node and target network.

10) Mobile Node may send MIH_MN_HO_Complete Request to Target PoS. Target PoS sends MIH_N2N_HO_Complete Request to previous Serving PoS to release resource which was allocated to Mobile Node. After identifying that resource is successfully released, Target PoS may send MIH_MN_HO_Complete Response to Mobile Node.

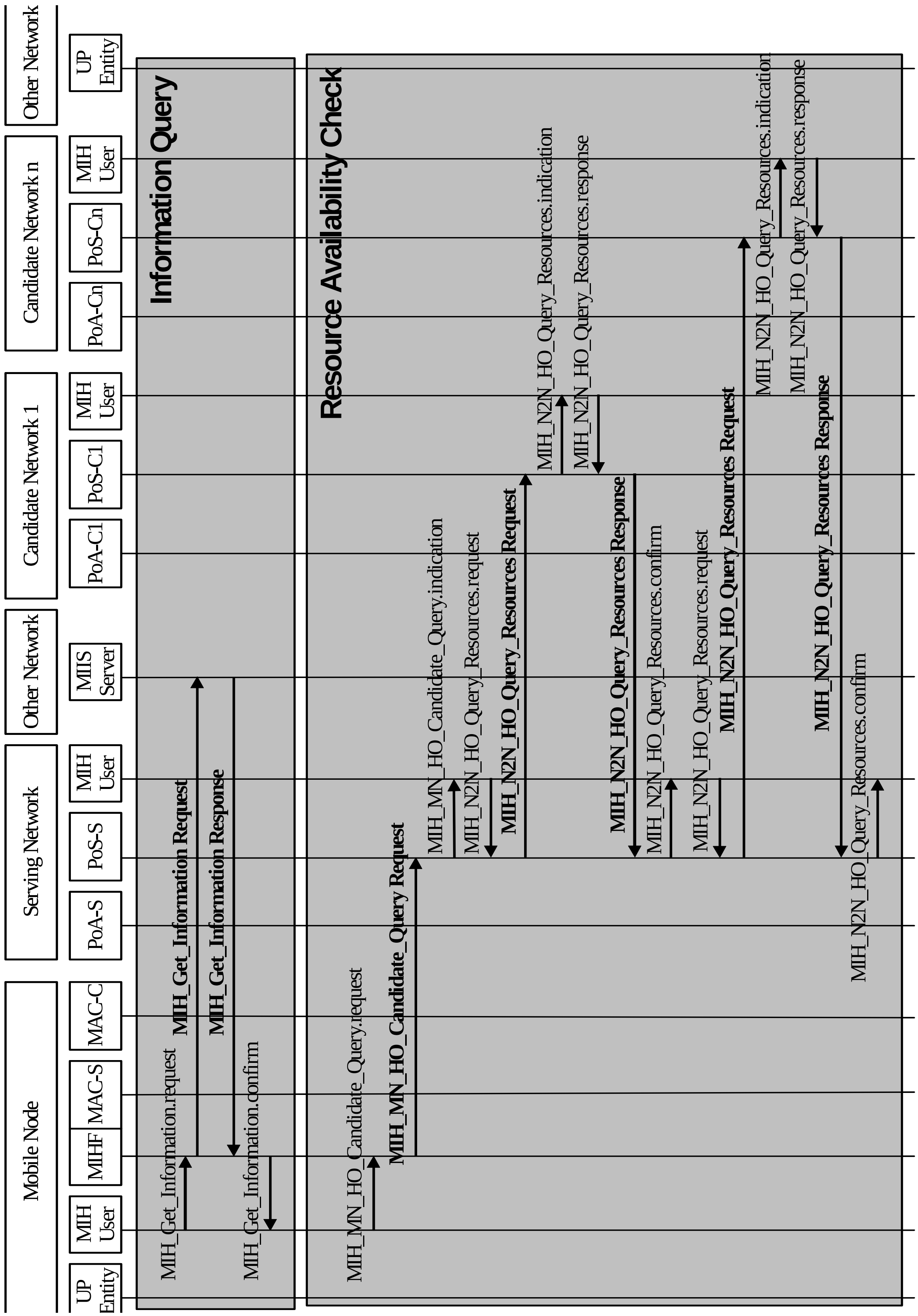


Figure H-1—Mobile-initiated handover procedure

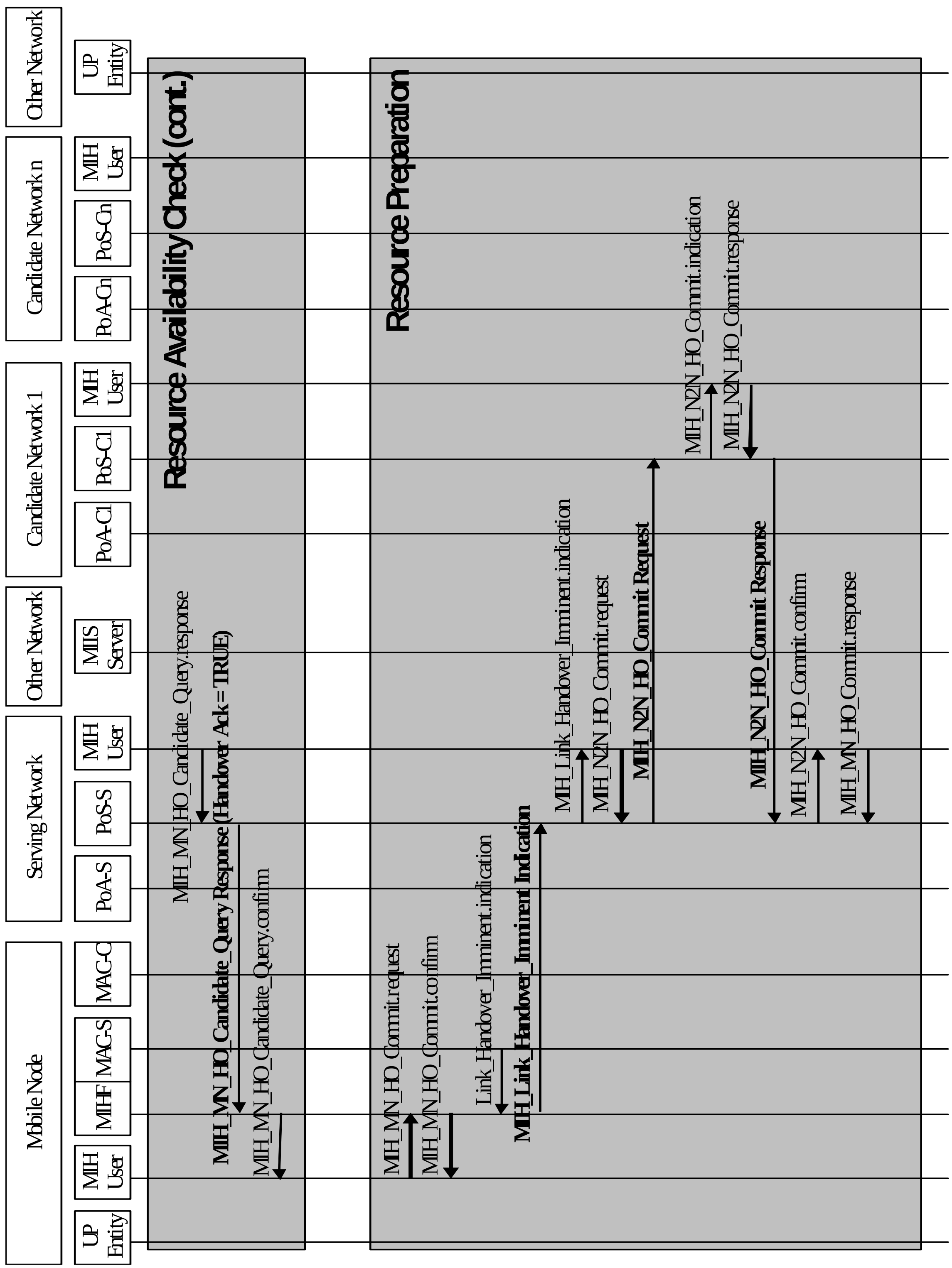
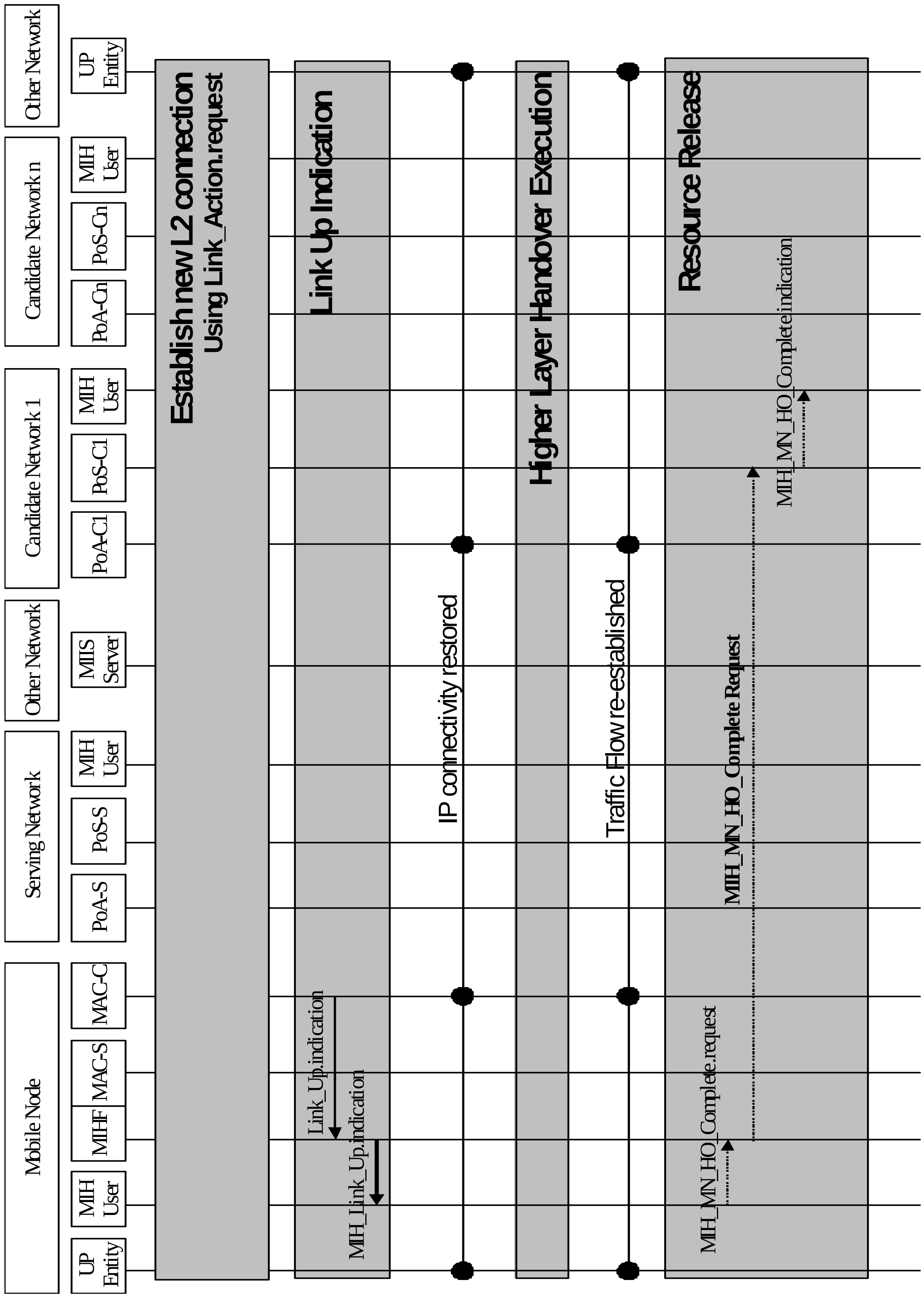


Figure H-1a—Mobile-initiated handover procedure (cont.)

Figure H-1b—Mobile-initiated handover procedure (cont.)


Mobile Node
Serving Network
Other Network
Candidate Network 1
Candidate Network n
Other Network
UP Entity
MIH User
MIHF
MAC-S
MAC-C
PoA-S
PoS-S
MIH User
MIIS Server
PoA-C1
PoS-C1
MIH User
PoA-Cn
PoS-Cn
MIH User
UP Entity
Establish new L2 connection
Using Link_Action.request
Link Up Indication
Link_Up.indication
MIH_Link_Up.indication
IP connectivity restored
Higher Layer Handover Execution
Traffic Flow re-established
Resource Release
MIH_MN_HO_Complete.request
MIH_MN_HO_Complete Request
MIH_MN_HO_Complete.indication

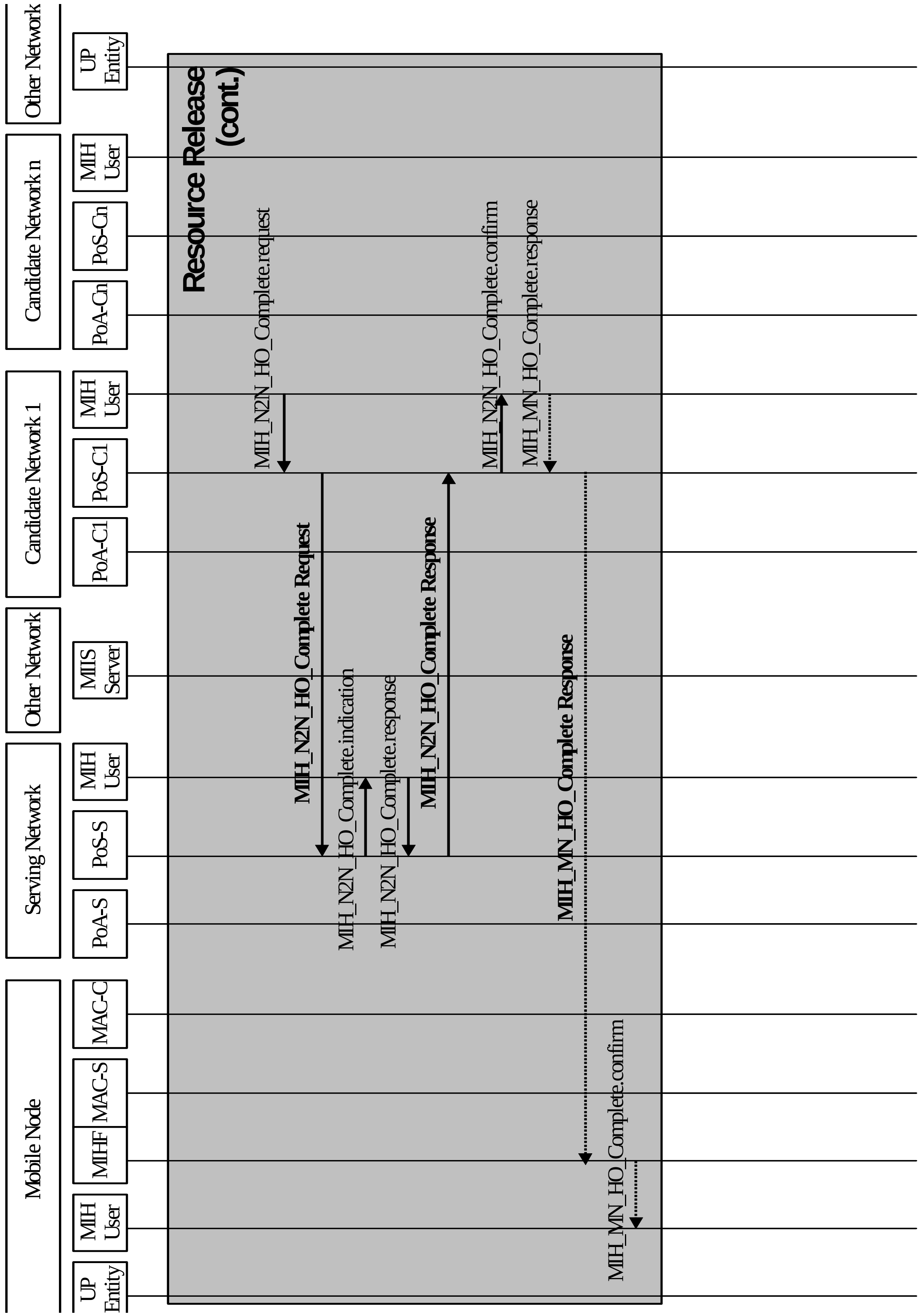


Figure H-1c—Mobile-initiated handover procedure (cont.)

H.2 Network-initiated handover procedure

The Network-initiated Handover Procedure operates as follows:

1) Serving PoS sends MIH_Get_Information Request to Information Server to get neighboring network information and Information Server responds by sending MIH_Get_Information Response.

2) Serving PoS triggers a network-initiated handover by sending MIH_Net_HO_Candidate_Query Request to Mobile Node. The MN responds through MIH_Net_HO_Candidate_Query Response which contains Mobile Node's acknowledgement about the handover and its preferred link and PoS lists.

3) Serving PoS sends MIH_N2N_HO_Query_Resources Request to one or more Candidate PoSs to check the availability of the resource at candidate networks. Candidate PoS responds by sending MIH_N2N_HO_Query_Resources Response to Serving PoS.

4) Serving PoS decides the target of the handover based on the available resource status at candidate networks.

5) Serving PoS sends MIH_N2N_HO_Commit Request to Target PoS to prepare resource at the target network. Target PoS responds the result of the resource preparation by sending MIH_N2N_HO_Commit Response.

6) After identifying that resource is successfully prepared, Serving PoS commands Mobile Node to commit handover toward the specified network type and PoA through MIH_Net_HO_Commit Request.

7) New layer 2 connection is established and Mobile Node sends MIH_Net_HO_Commit Response to Serving PoS.

8) After higher layer handover execution, Mobile Node may send MIH_MN_HO_Complete Request to Target PoS. Target PoS sends MIH_N2N_HO_Complete Request to previous Serving PoS to release resource which was allocated to Mobile Node. After identifying that resource is successfully released, Target PoS may send MIH_MN_HO_Complete Response to Mobile Node.

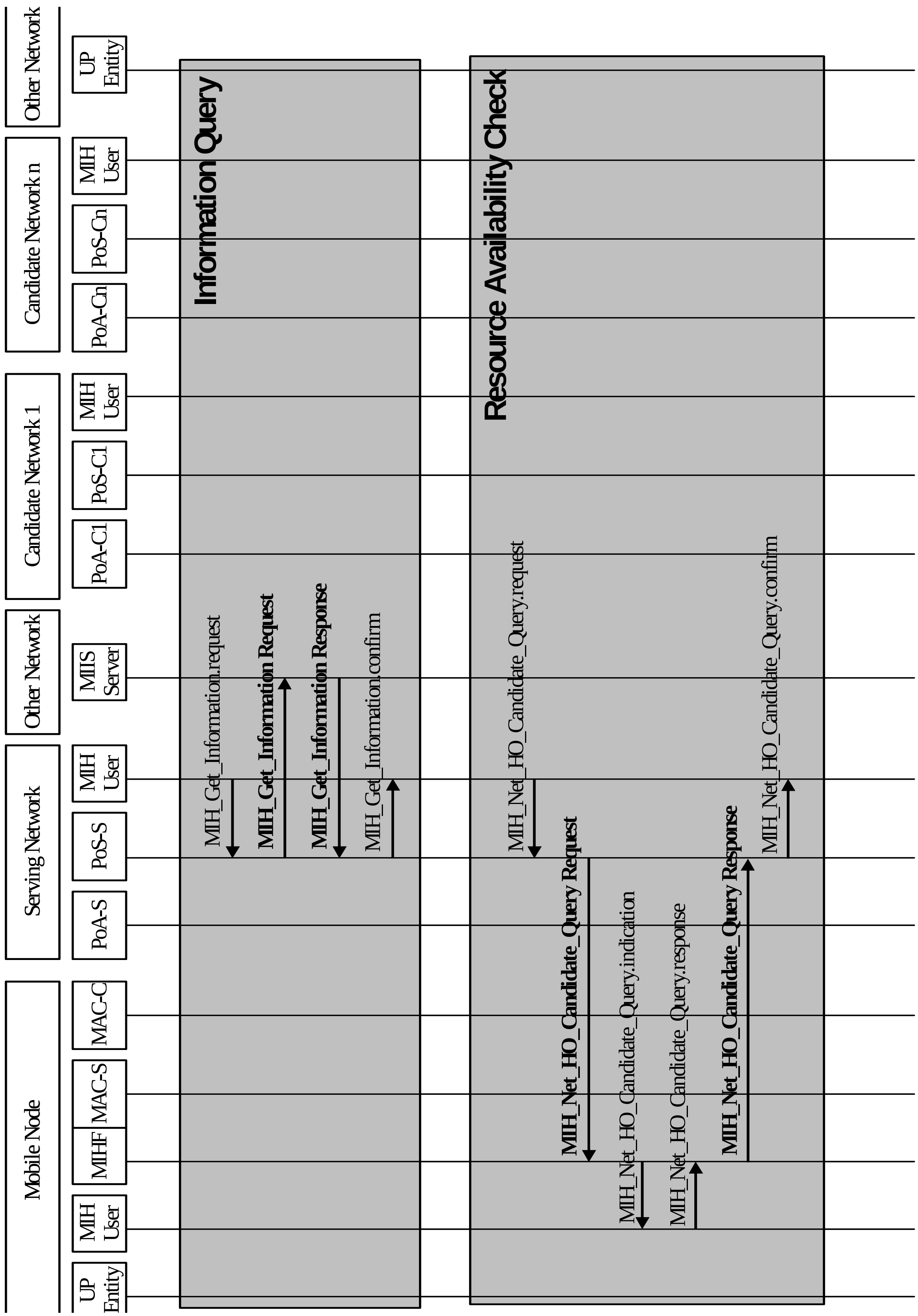


Figure H-2—Network-initiated handover procedure

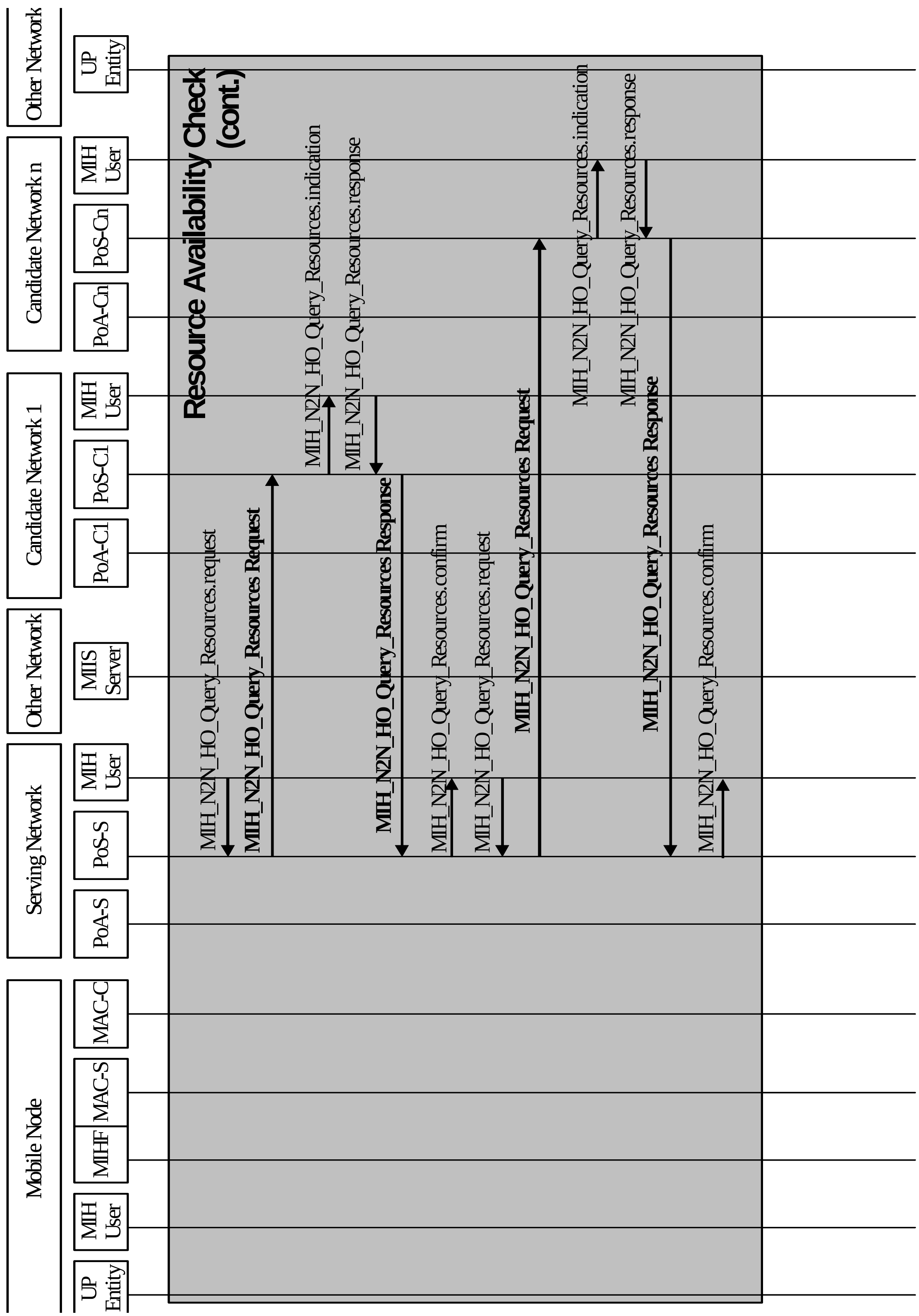


Figure H-2a—Network-initiated handover procedure (cont.)

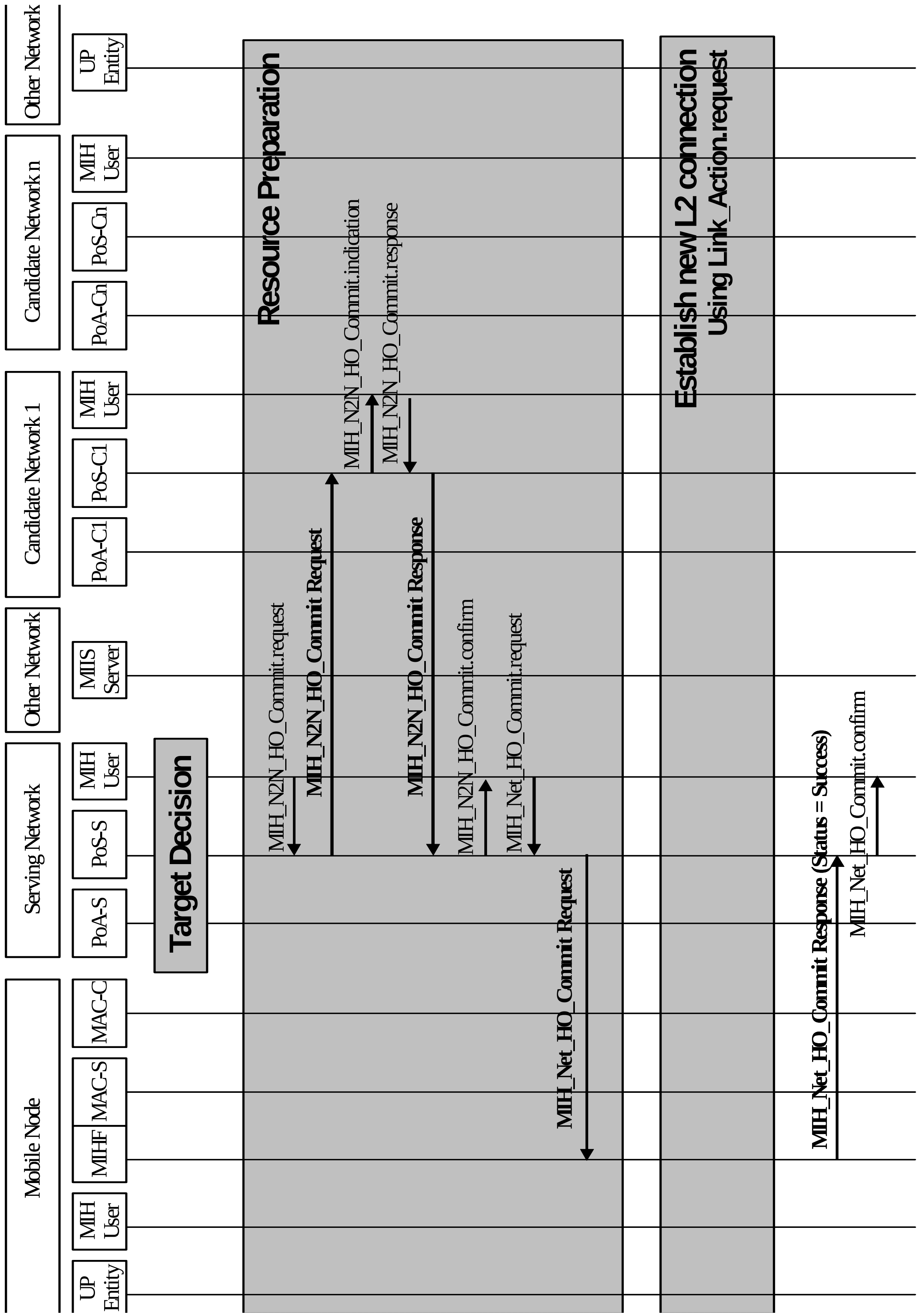


Figure H-2b—Network-initiated handover procedure (cont.)

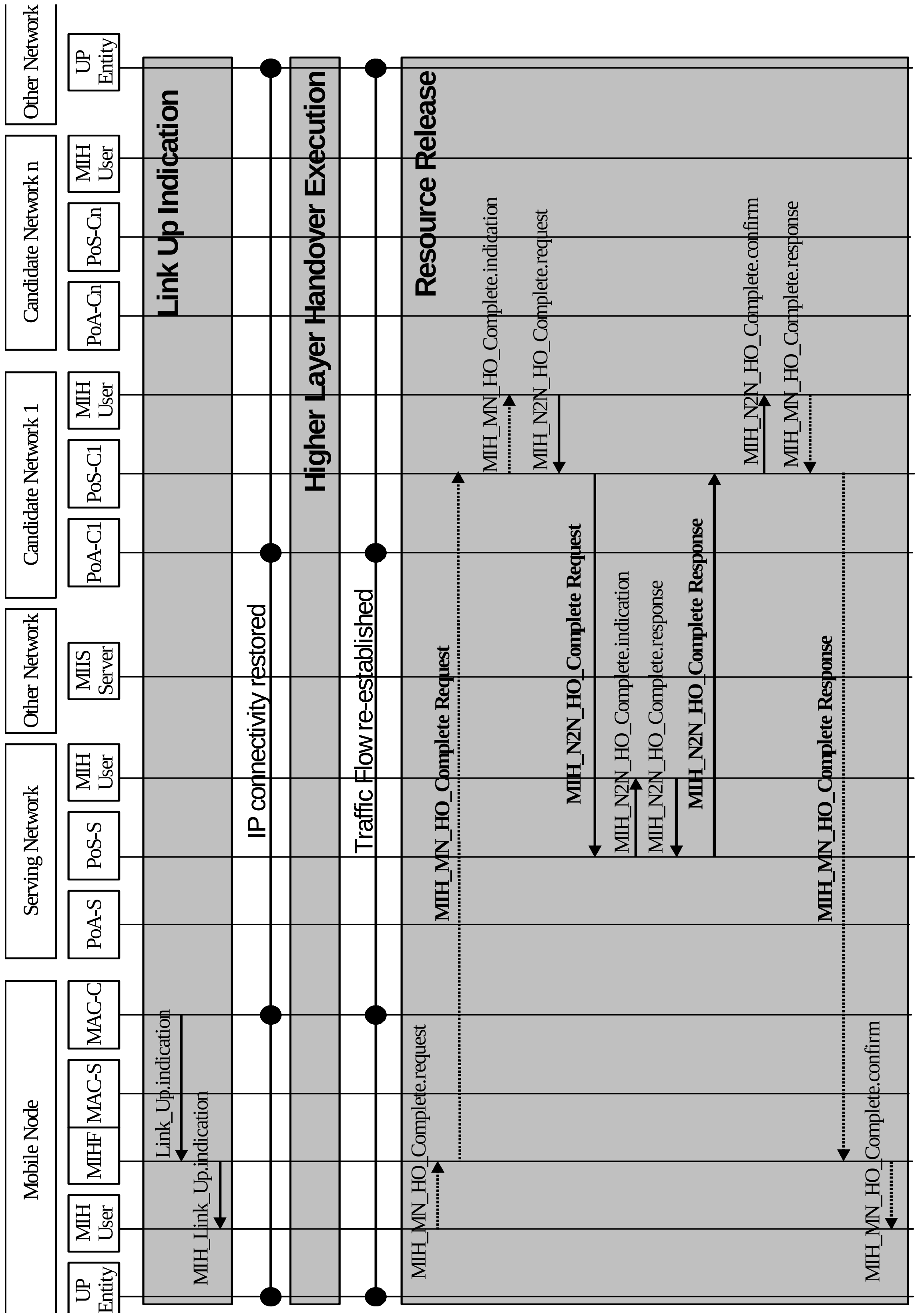


Figure H-2c—Network-initiated handover procedure (cont.)

H.3 Network-initiated handover from 802.16 to 3G

H.3.1 Network-initiated handover from 802.16 to 3G (Single PoS)

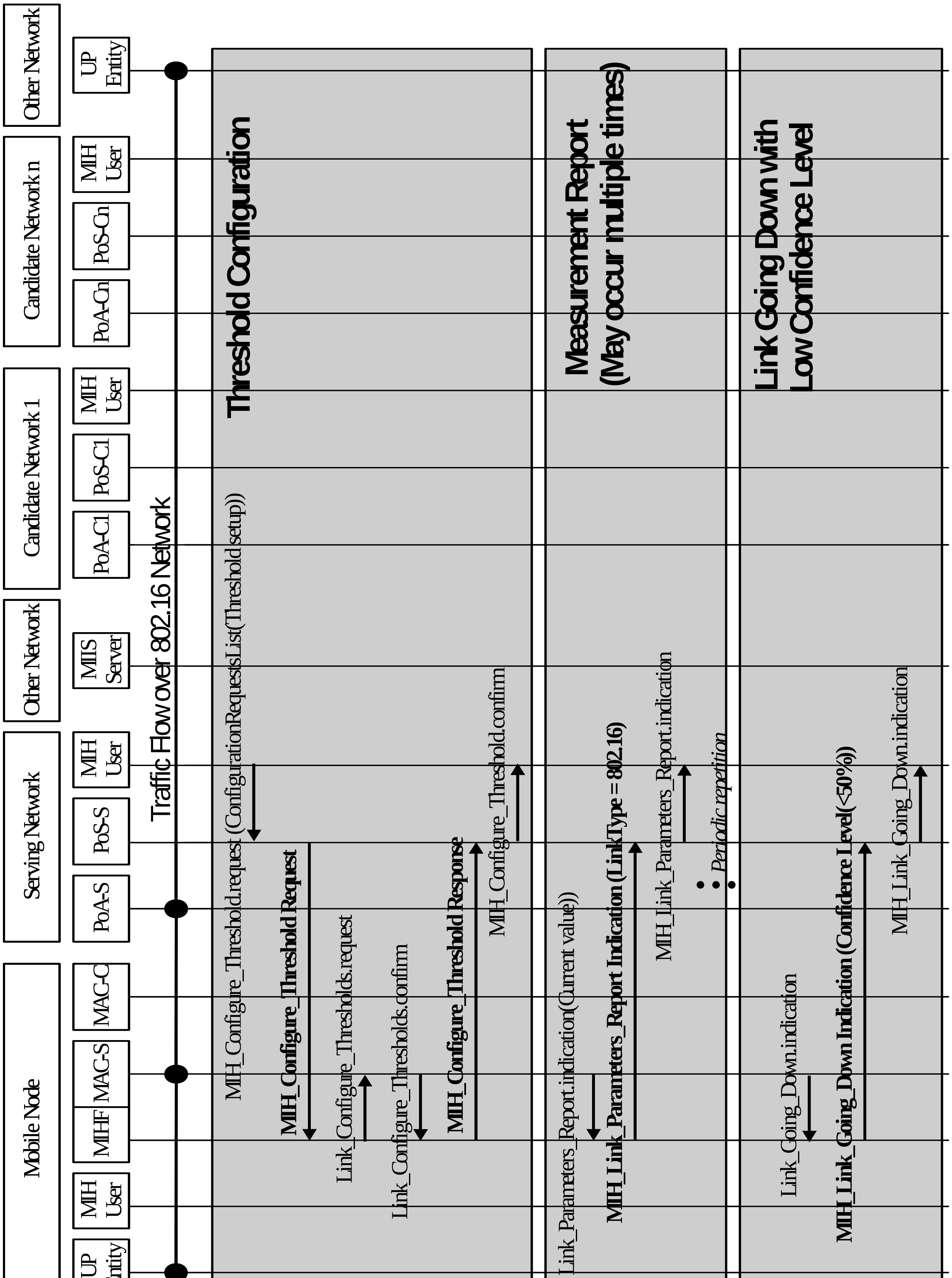


Figure H-3—Network-initiated handover from 802.16 to 3G (Single PoS)

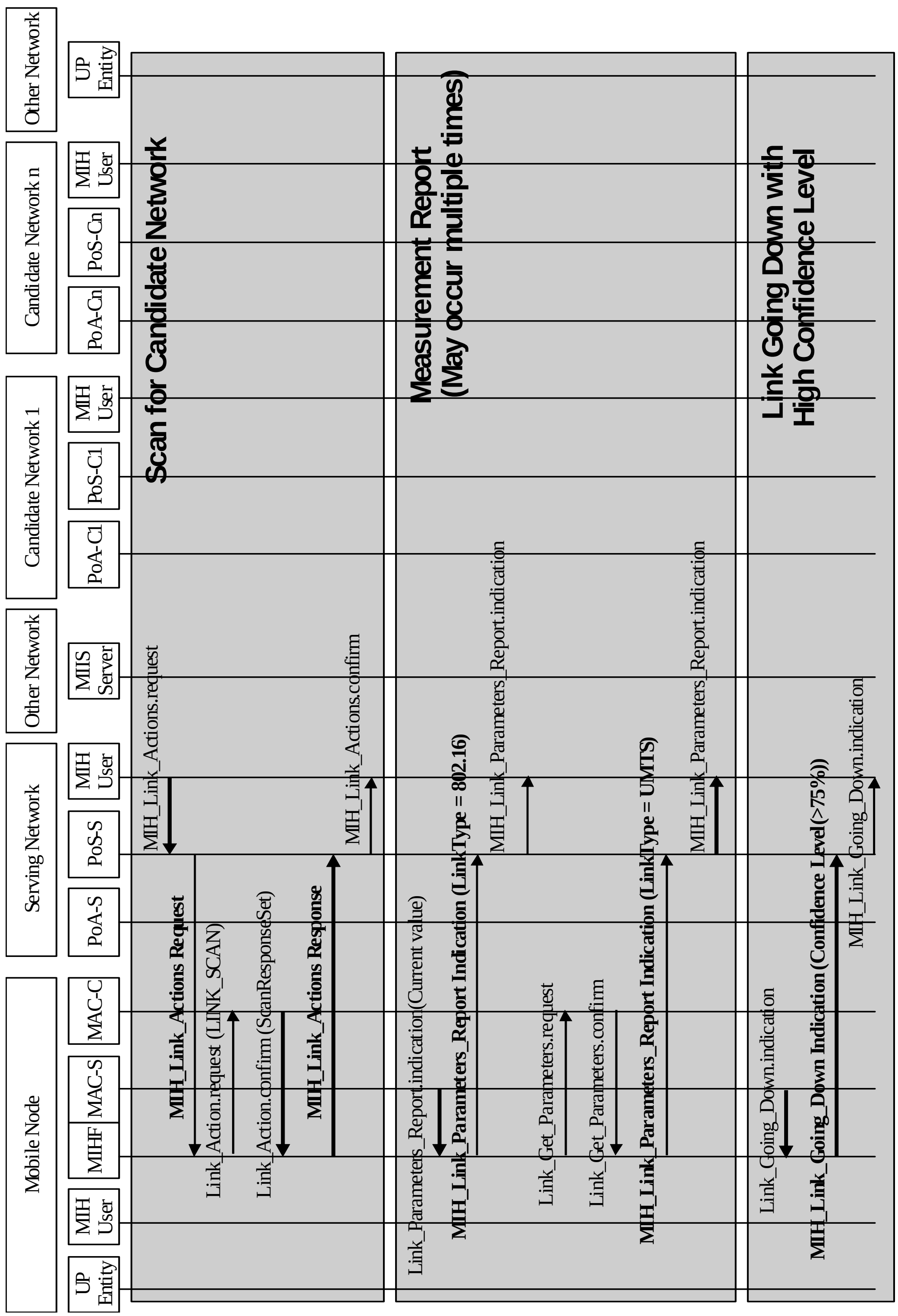


Figure H-3a—Network-initiated handover from 802.16 to 3G (Single PoS) (cont.)

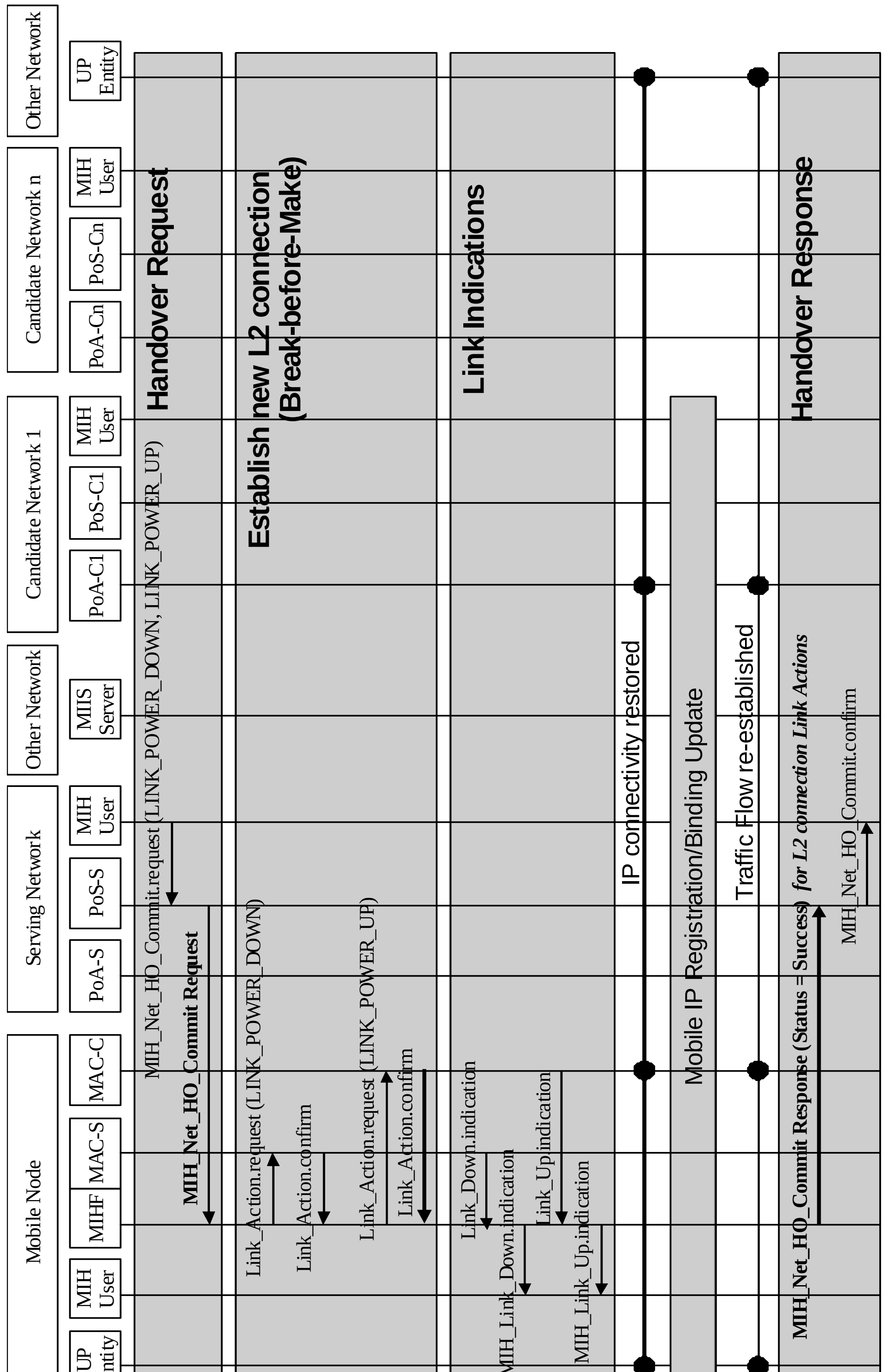


Figure H-3b—Network-initiated handover from 802.16 to 3G (Single PoS) (cont.)

H.3.2 Network-initiated handover from 802.16 to 3G (Multiple PoS)

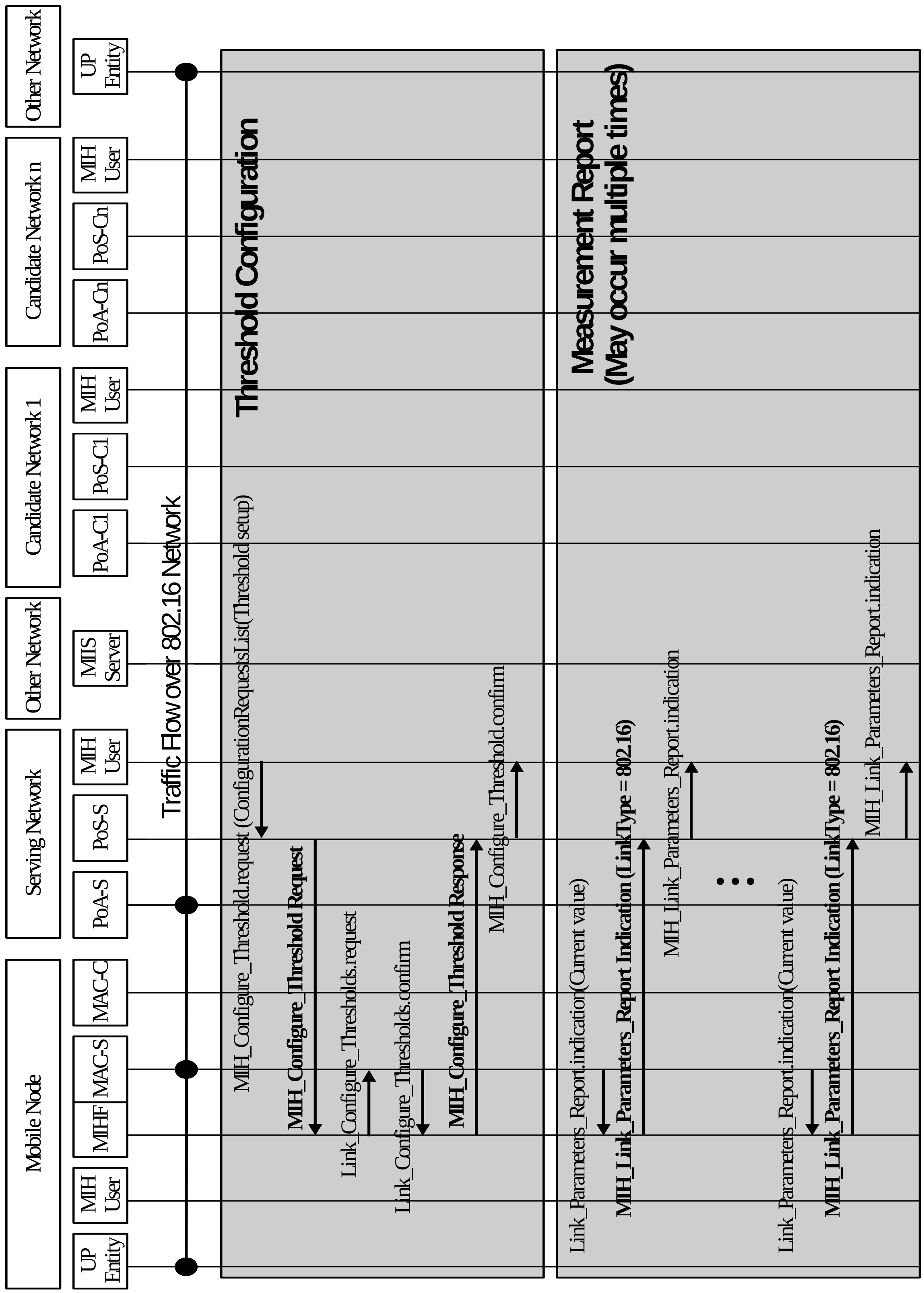


Figure H-4—Network-initiated handover from 802.16 to 3G (Multiple PoS)

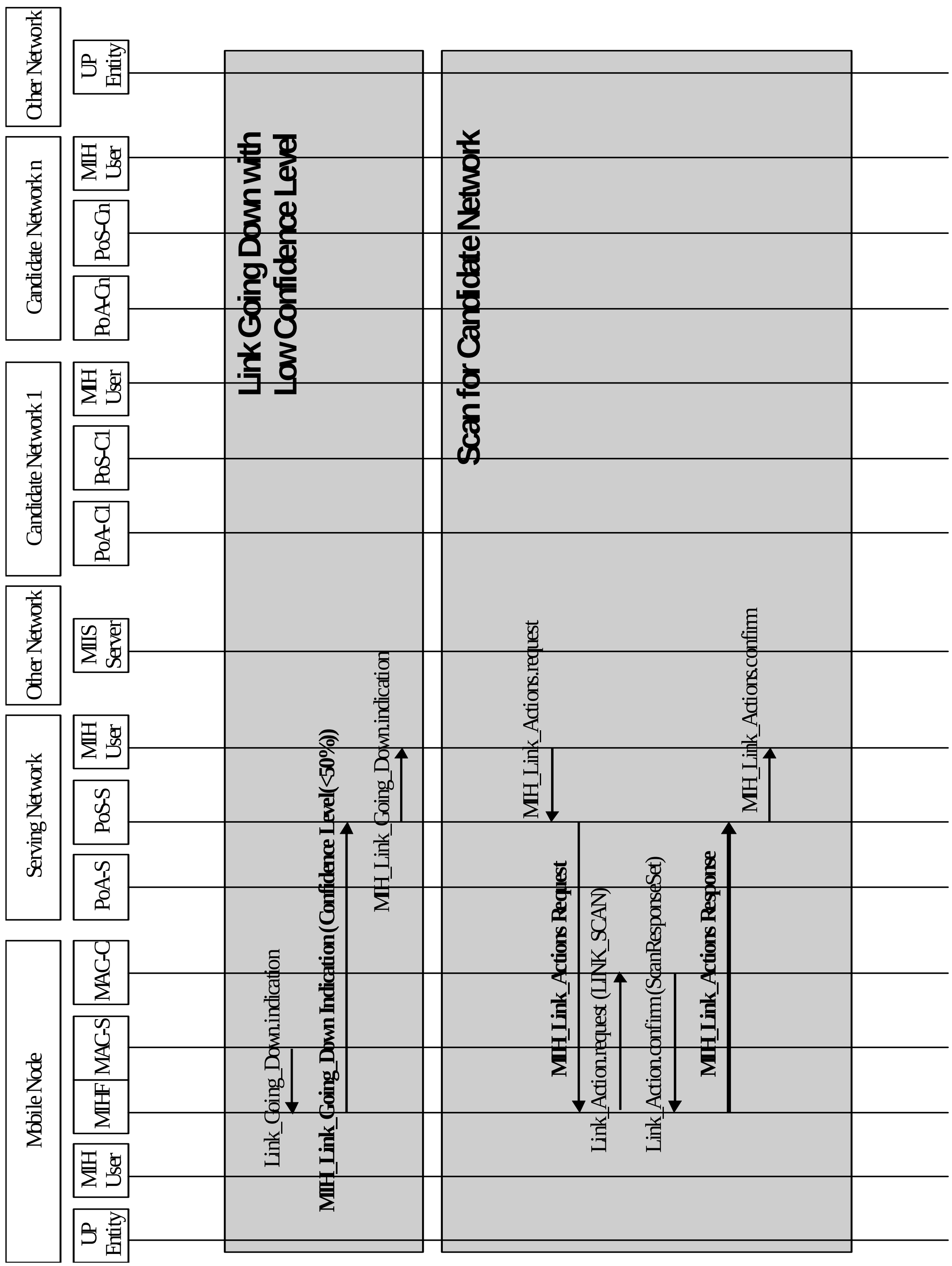


Figure H-4a—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.)

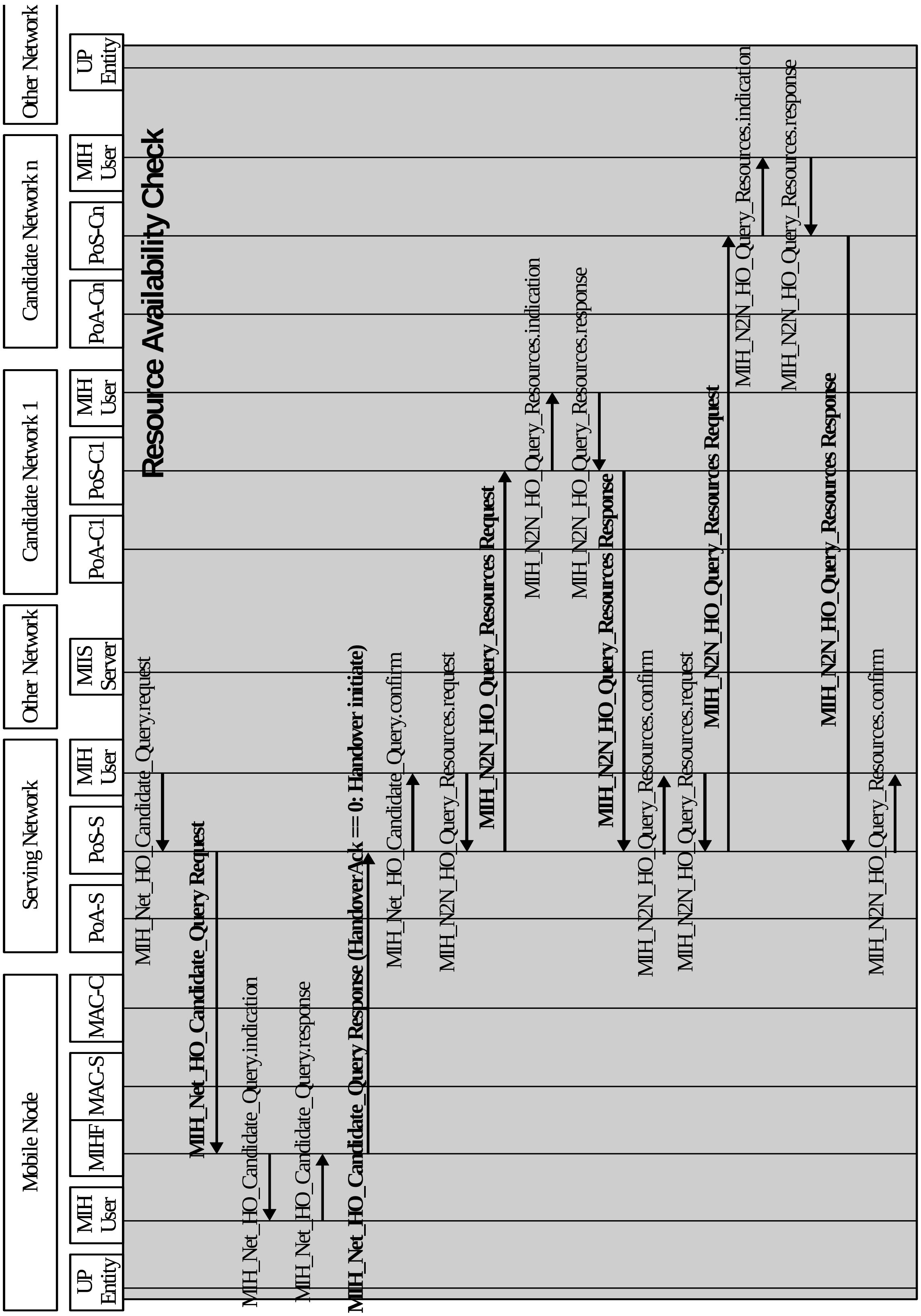


Figure H-4b—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.)

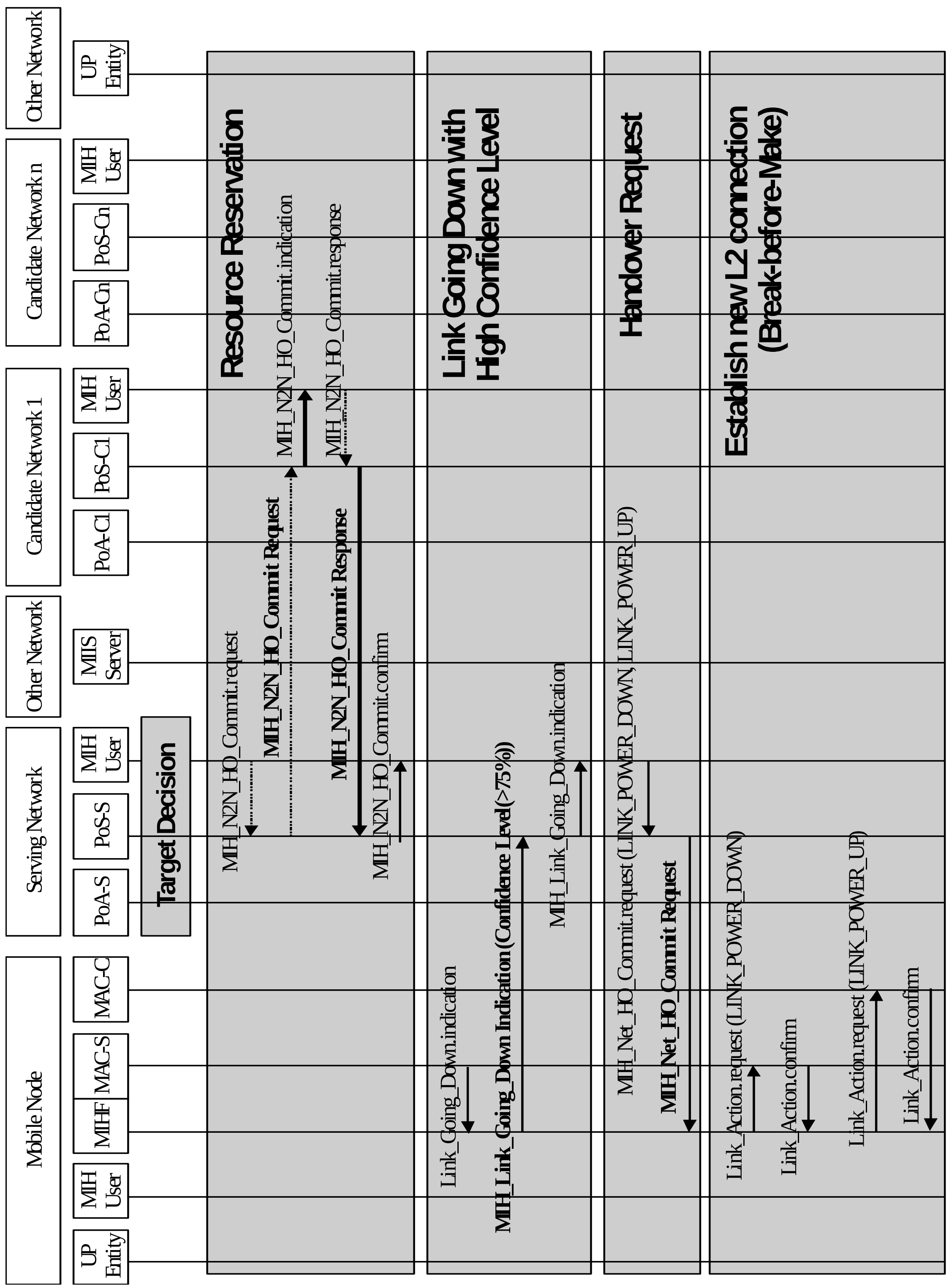


Figure H-4c—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.)

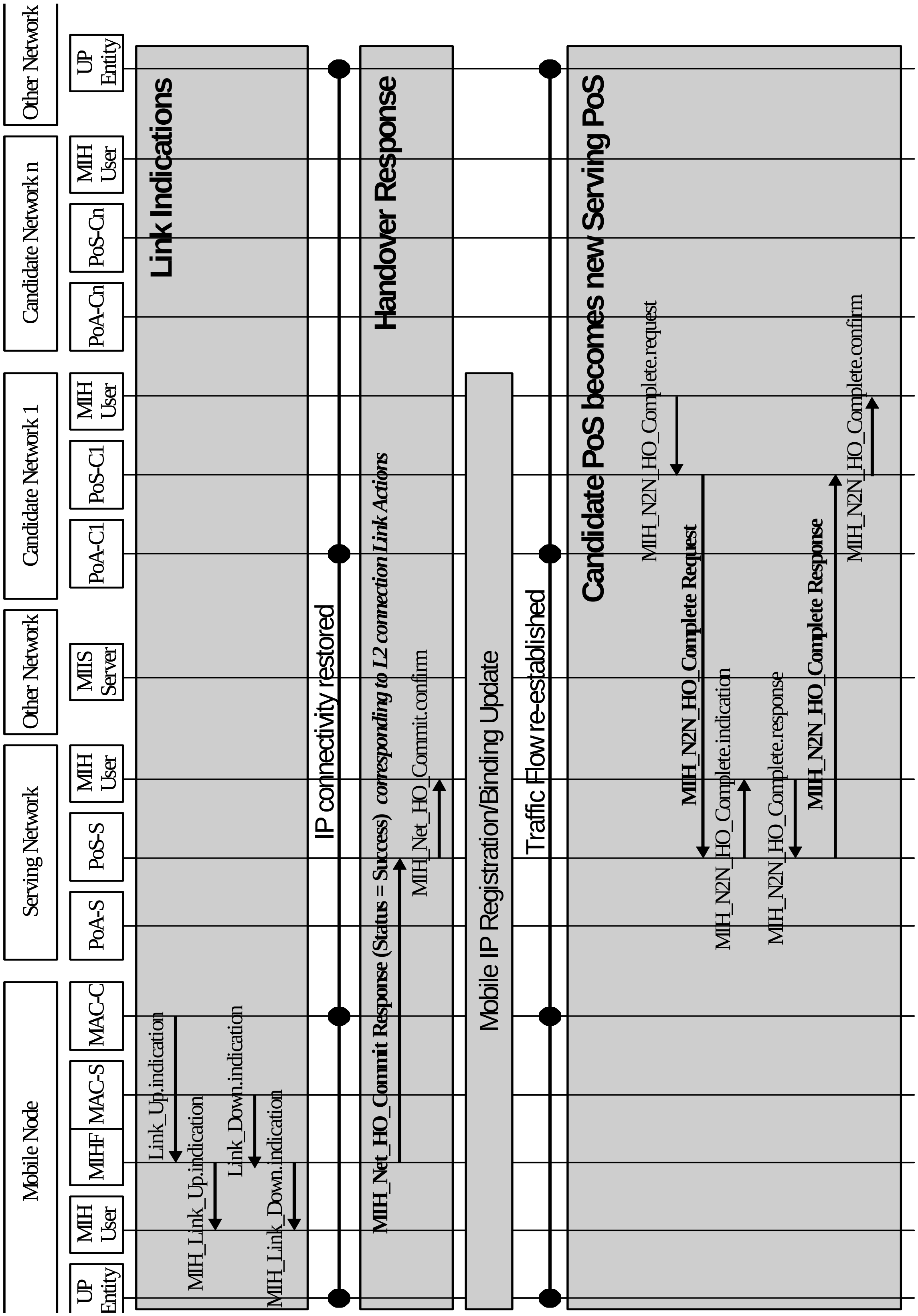


Figure H-4d—Network-initiated handover from 802.16 to 3G (Multiple PoS) (cont.)

H.4 Handover from 3G to WLAN

H.4.1 Network-initiated handover from 3G to WLAN

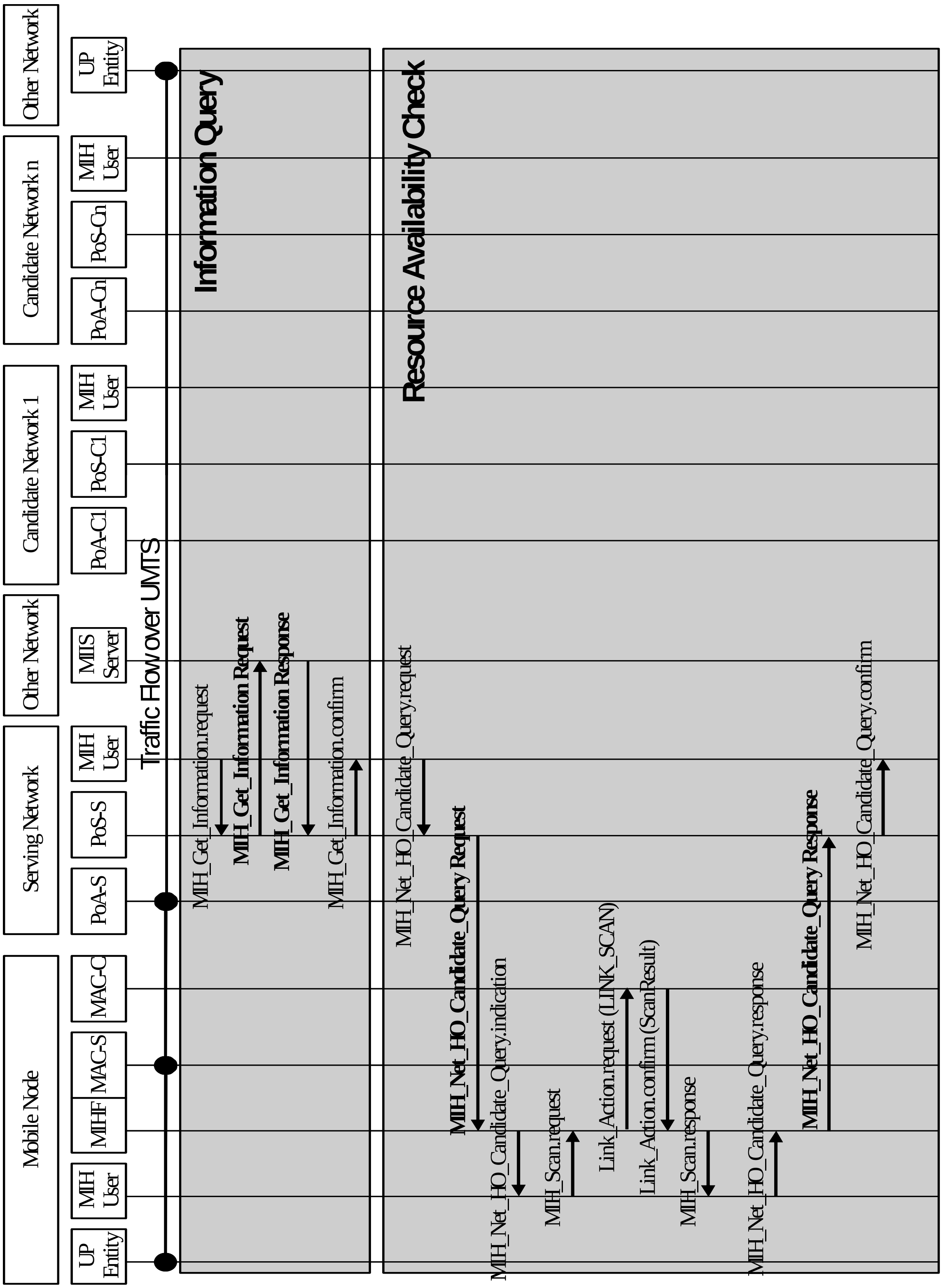


Figure H-5—Network-initiated handover from 3G to WLAN

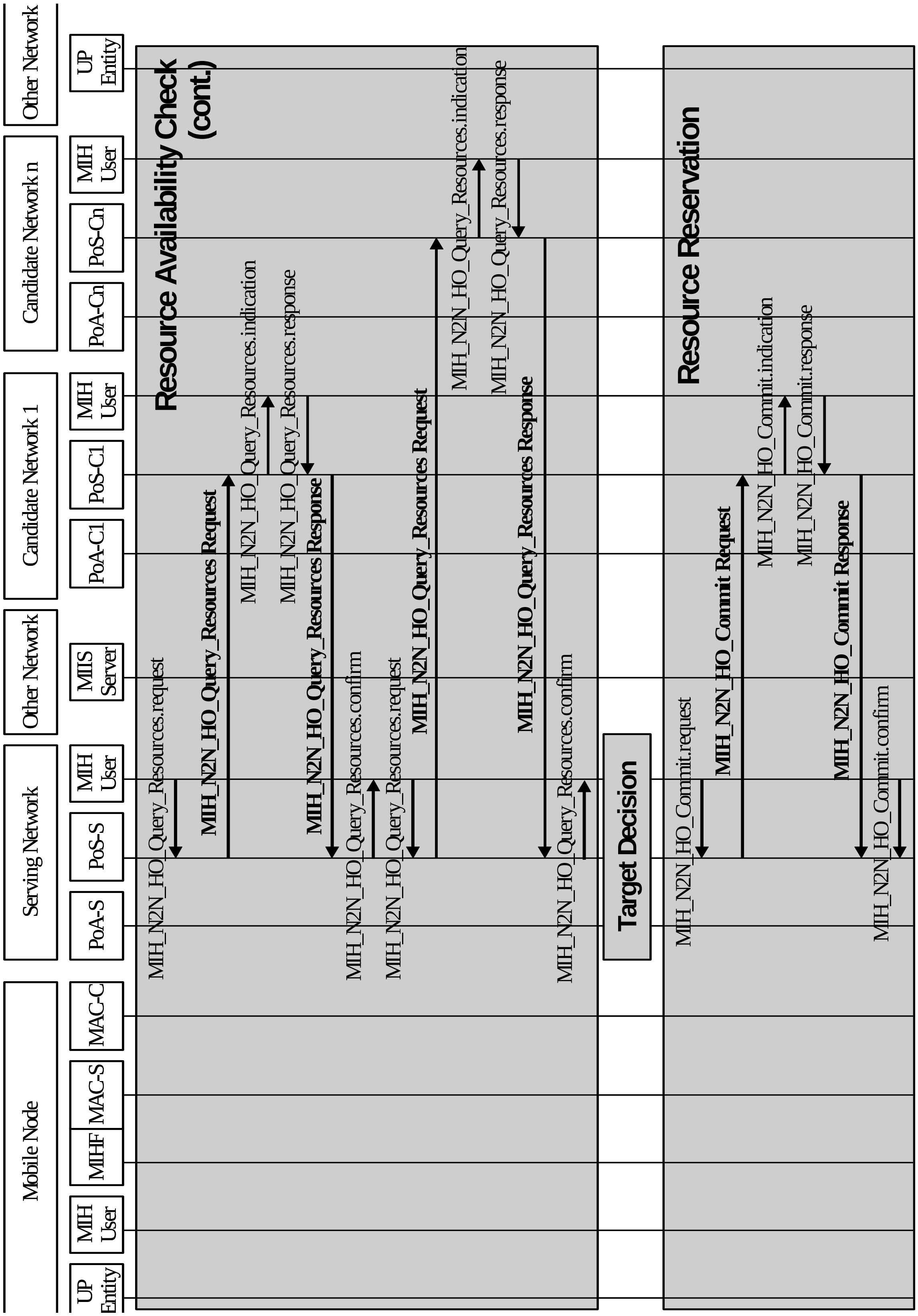


Figure H-5a—Network-initiated handover from 3G to WLAN (cont.)

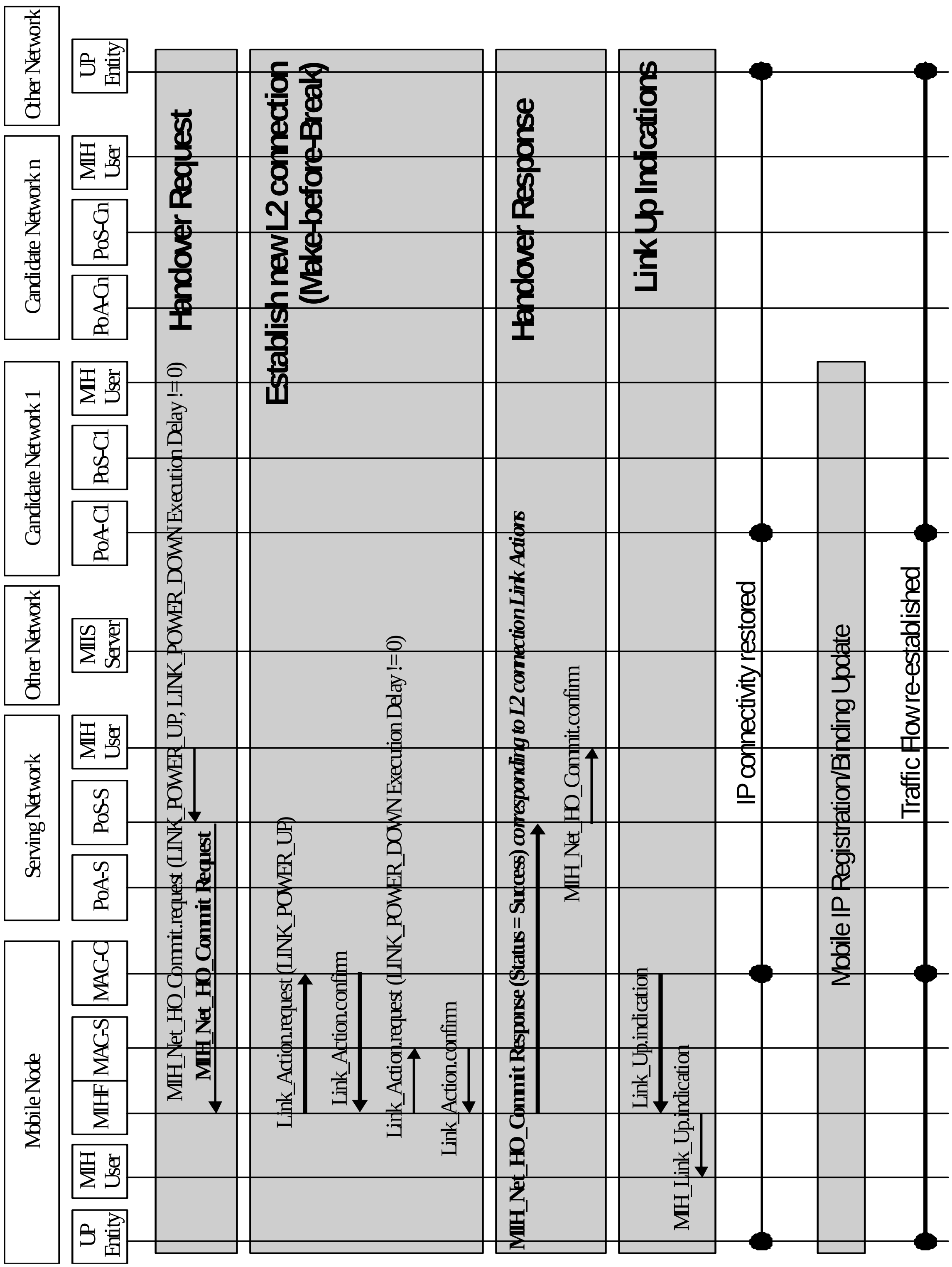


Figure H-5b—Network-initiated handover from 3G to WLAN (cont.)

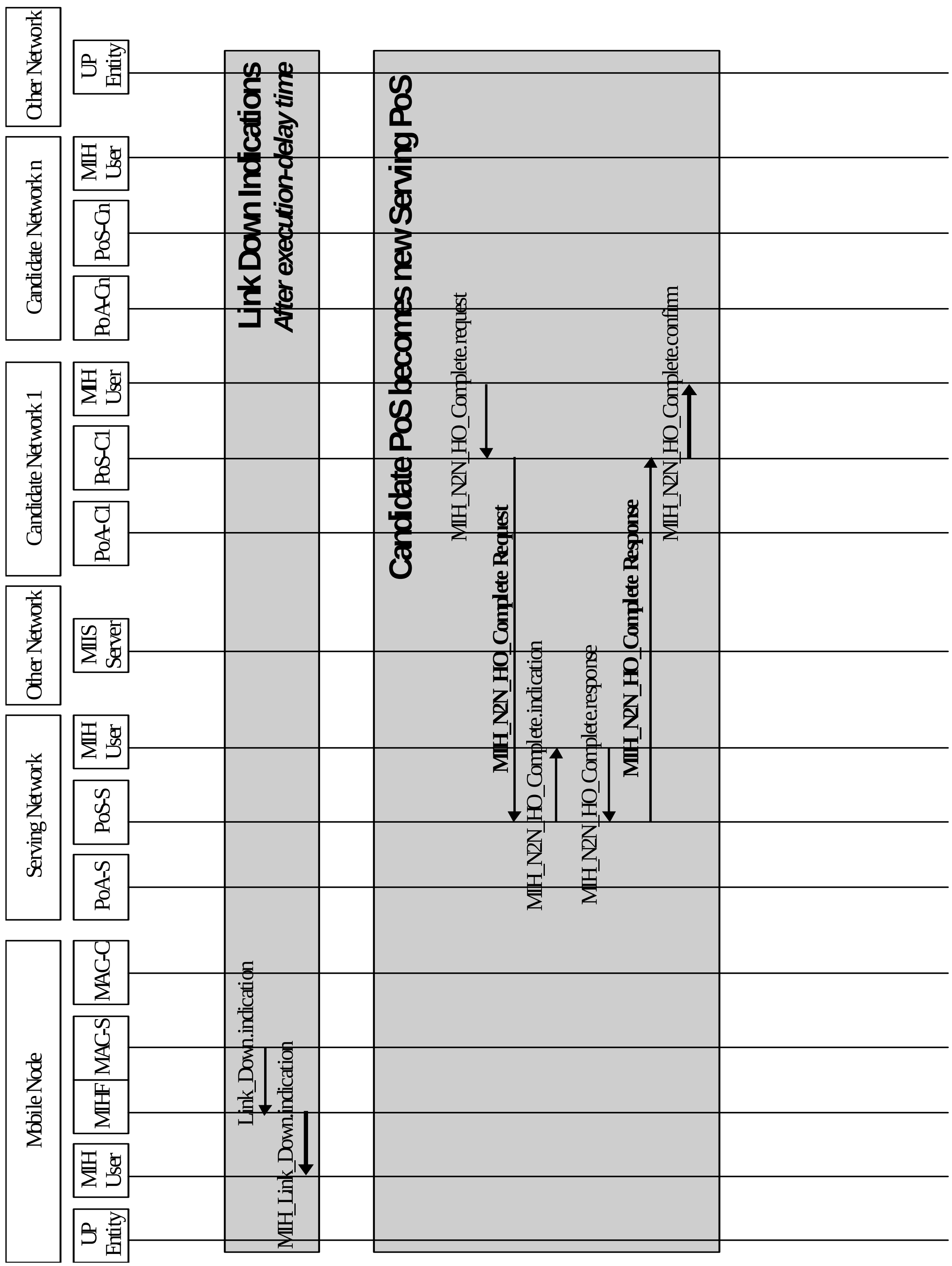


Figure H-5c—Network-initiated handover from 3G to WLAN (cont.)

H.4.2 Mobile-initiated handover from 3G to WLAN

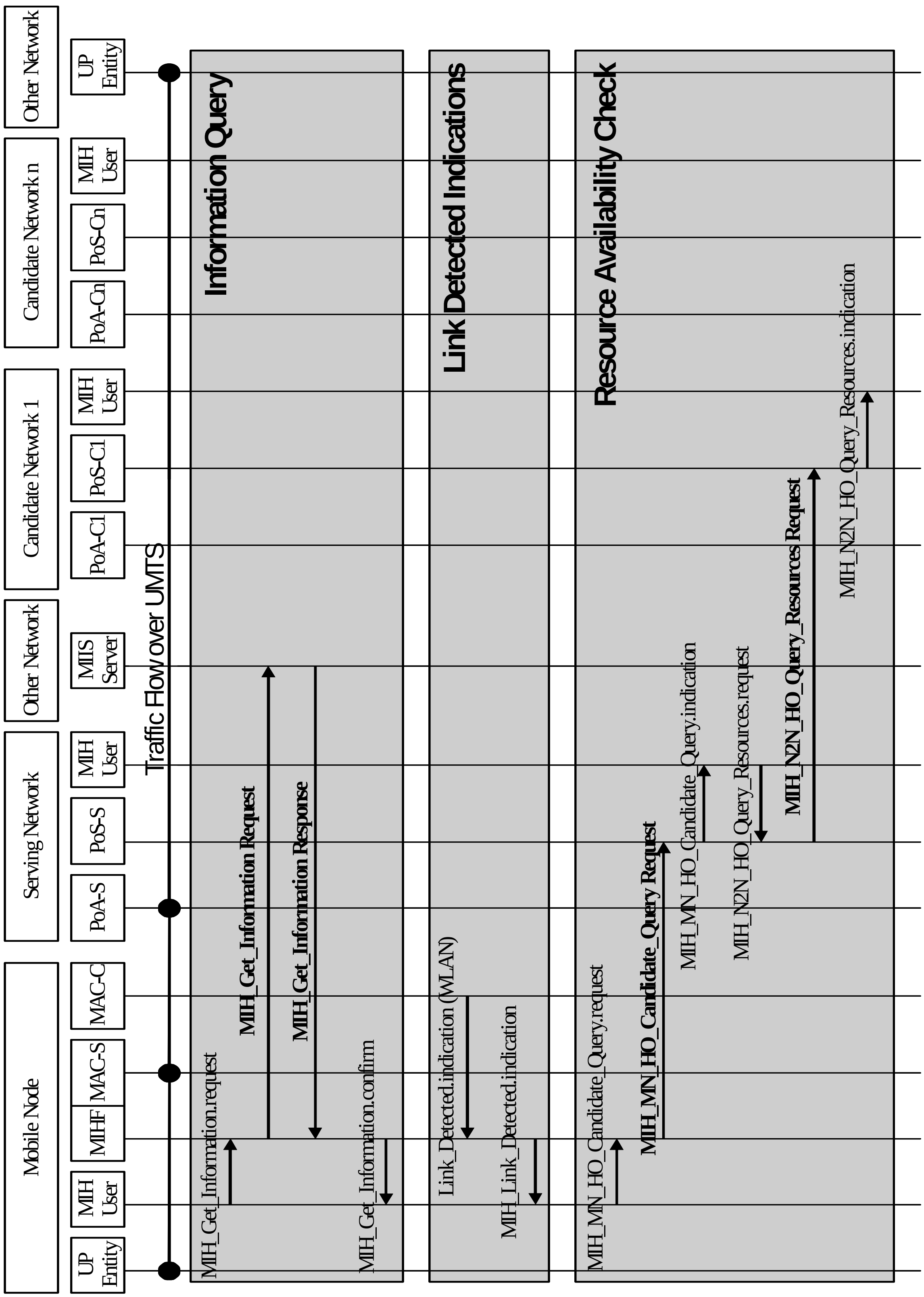


Figure H-6—Mobile-initiated handover from 3G to WLAN

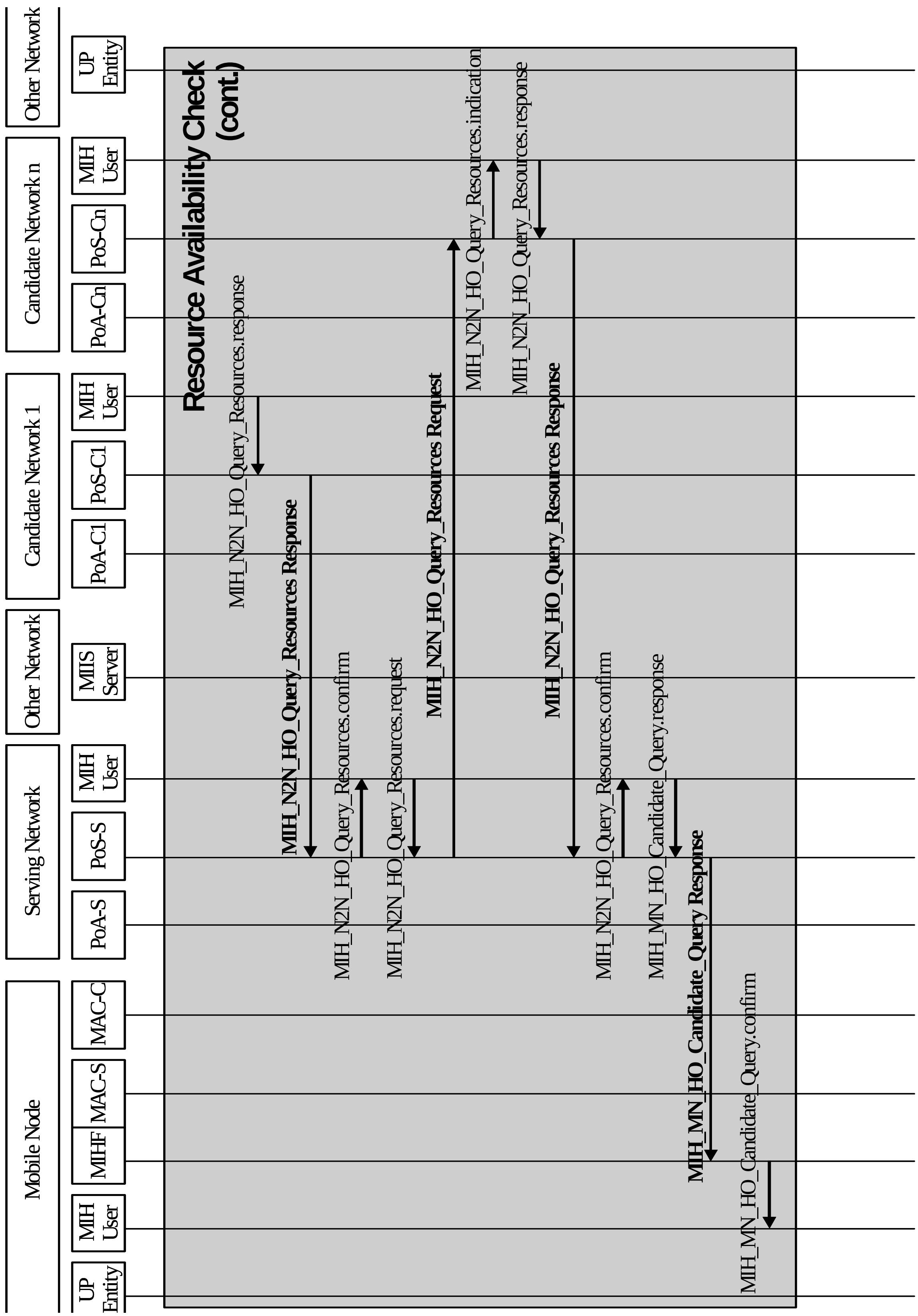


Figure H-6a—Mobile-initiated handover from 3G to WLAN (cont.)

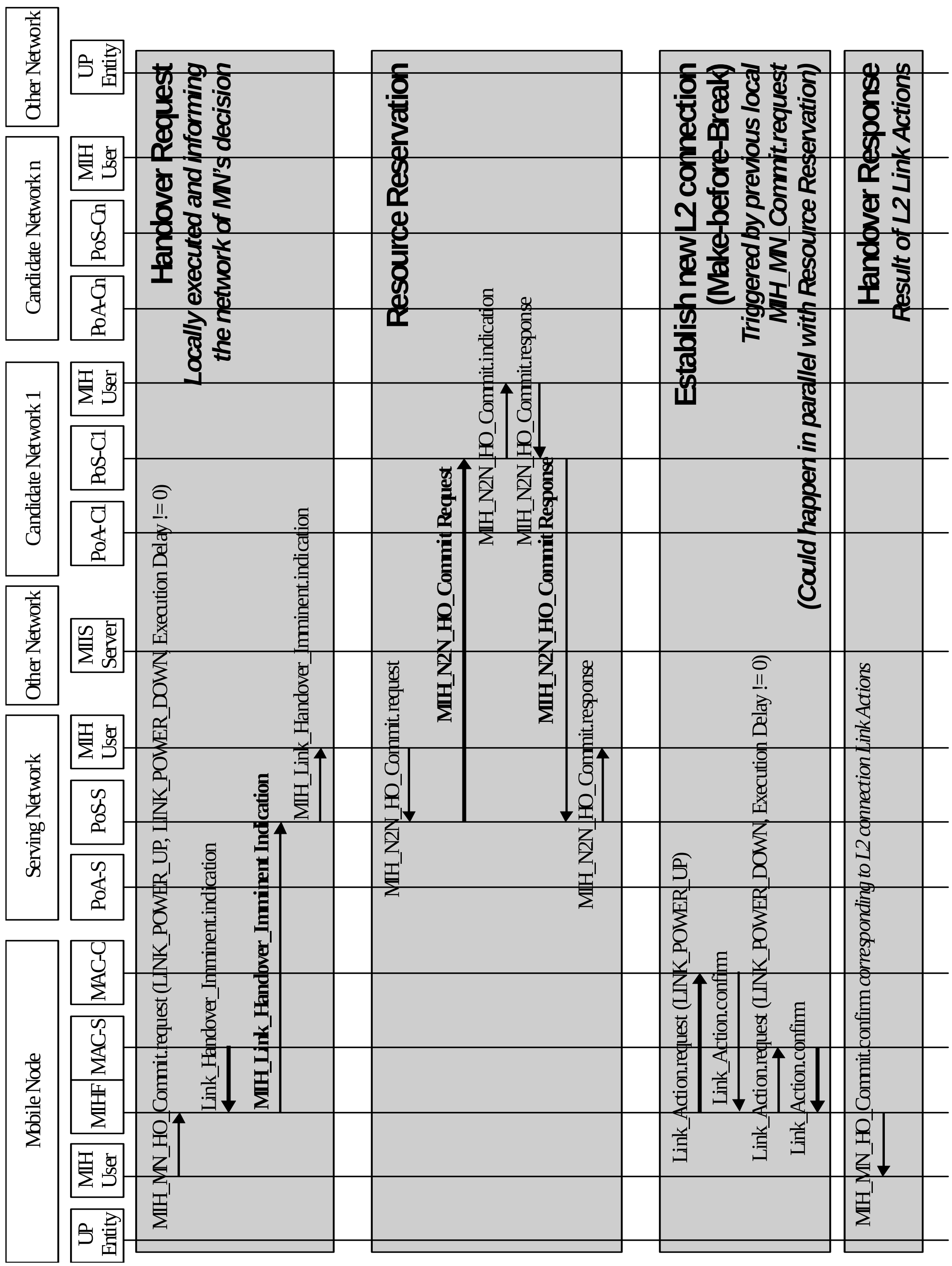


Figure H-6b—Mobile-initiated handover from 3G to WLAN (cont.)

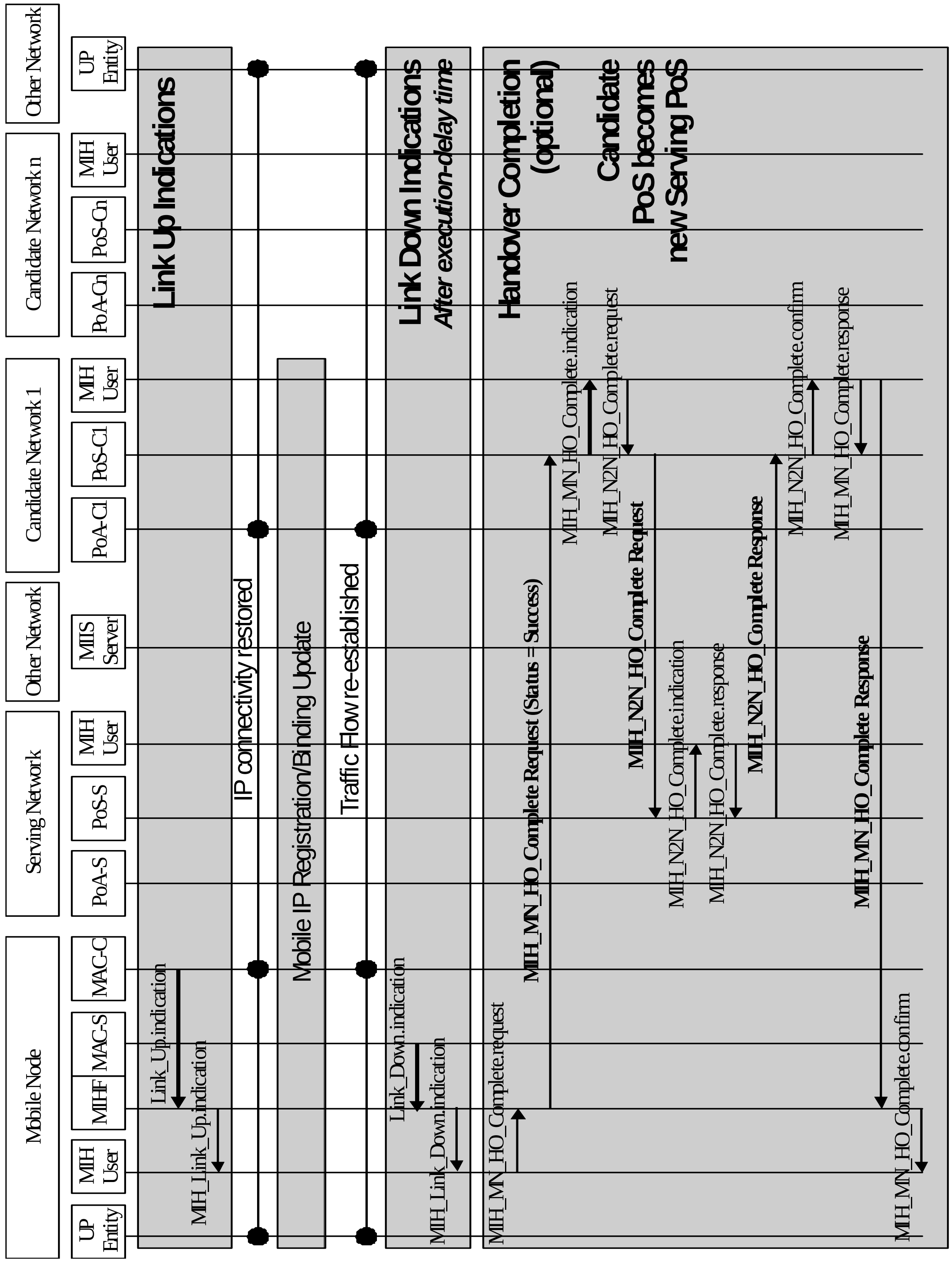


Figure H-6c—Mobile-initiated handover from 3G to WLAN (cont.)

H.5 Example handover flow chart between 802.11 and 802.16

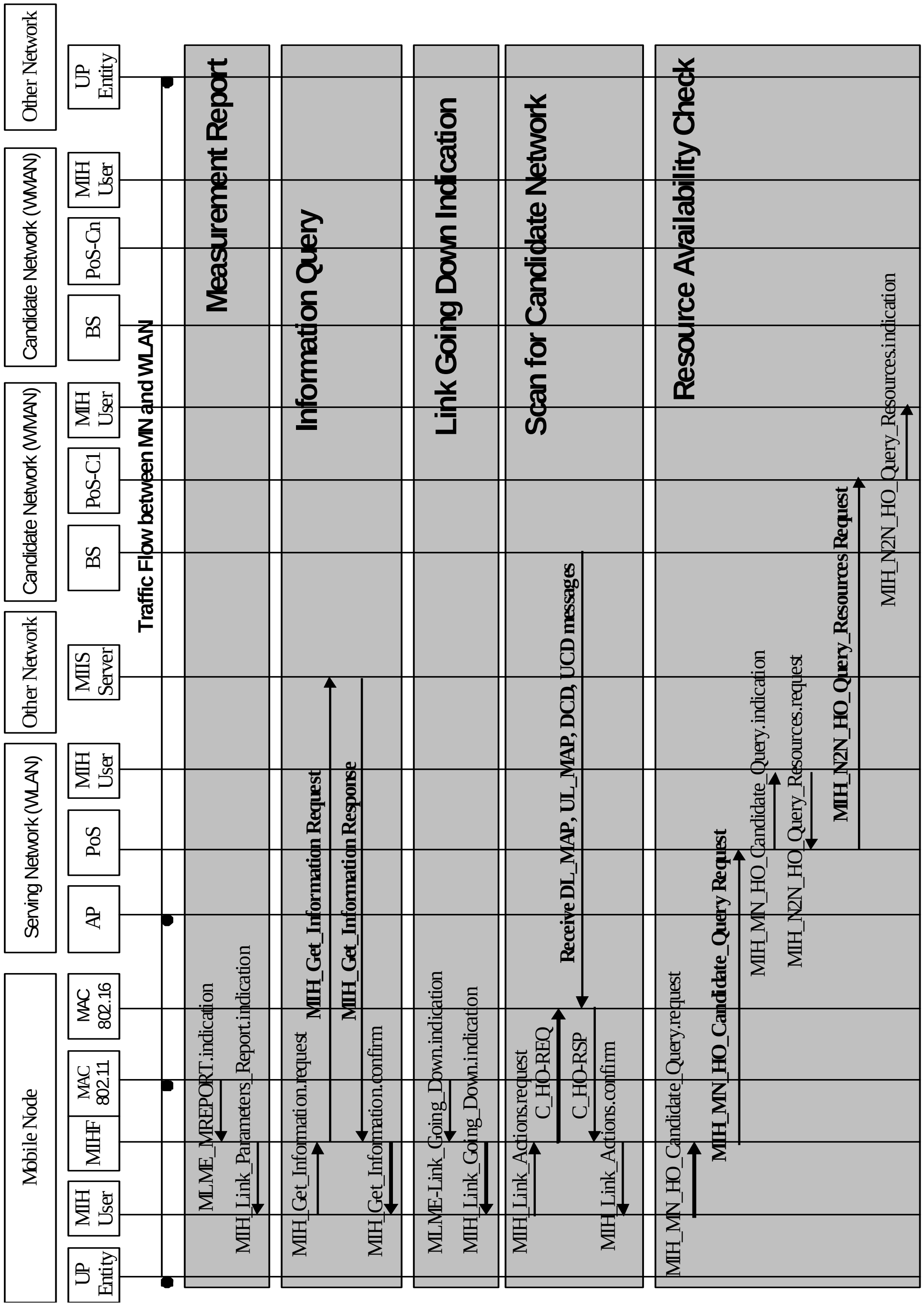


Figure H-7—Example handover flow chart between 802.11 and 802.16

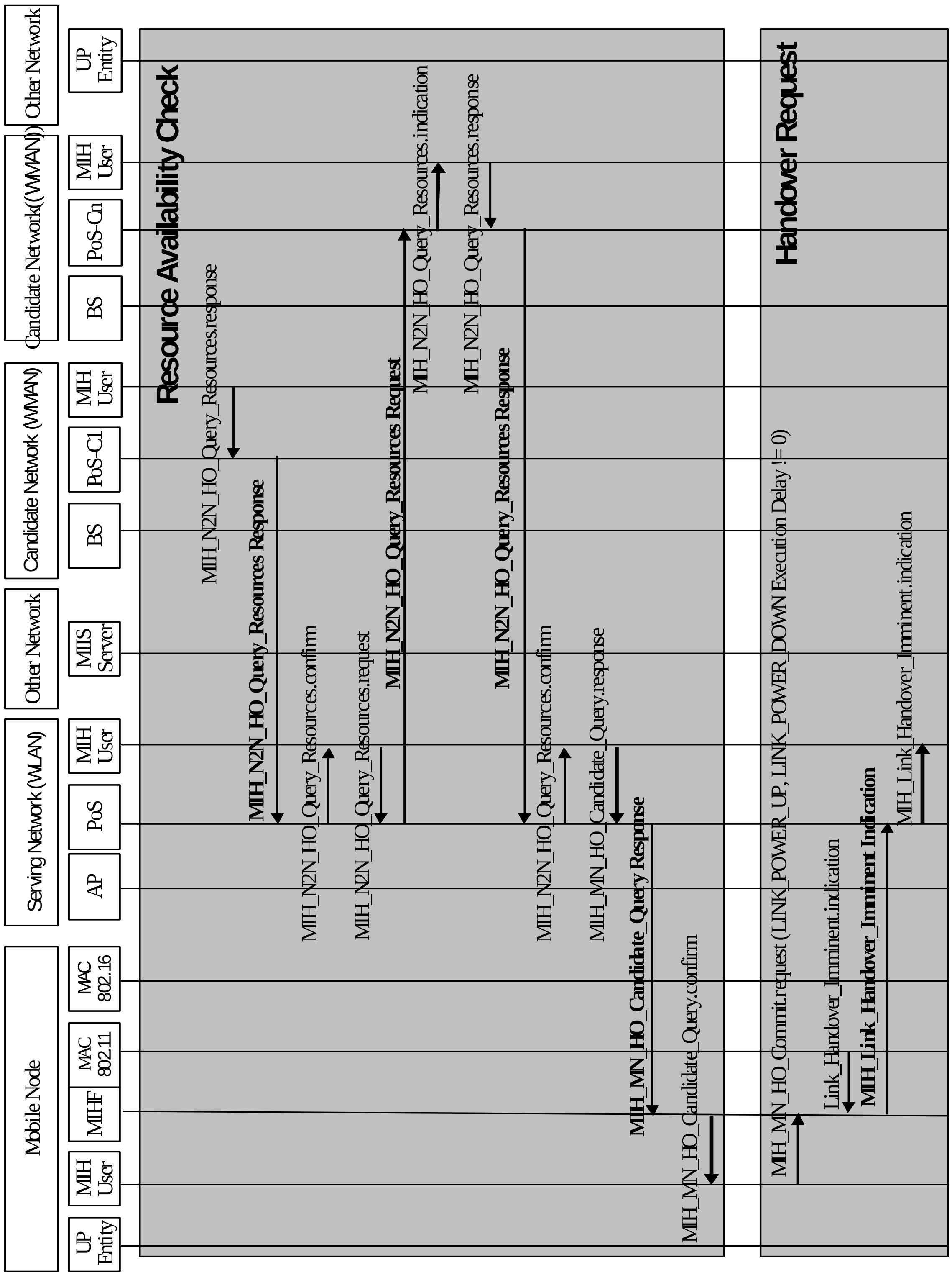


Figure H-7a—Example handover flow chart between 802.11 and 802.16 (cont.)

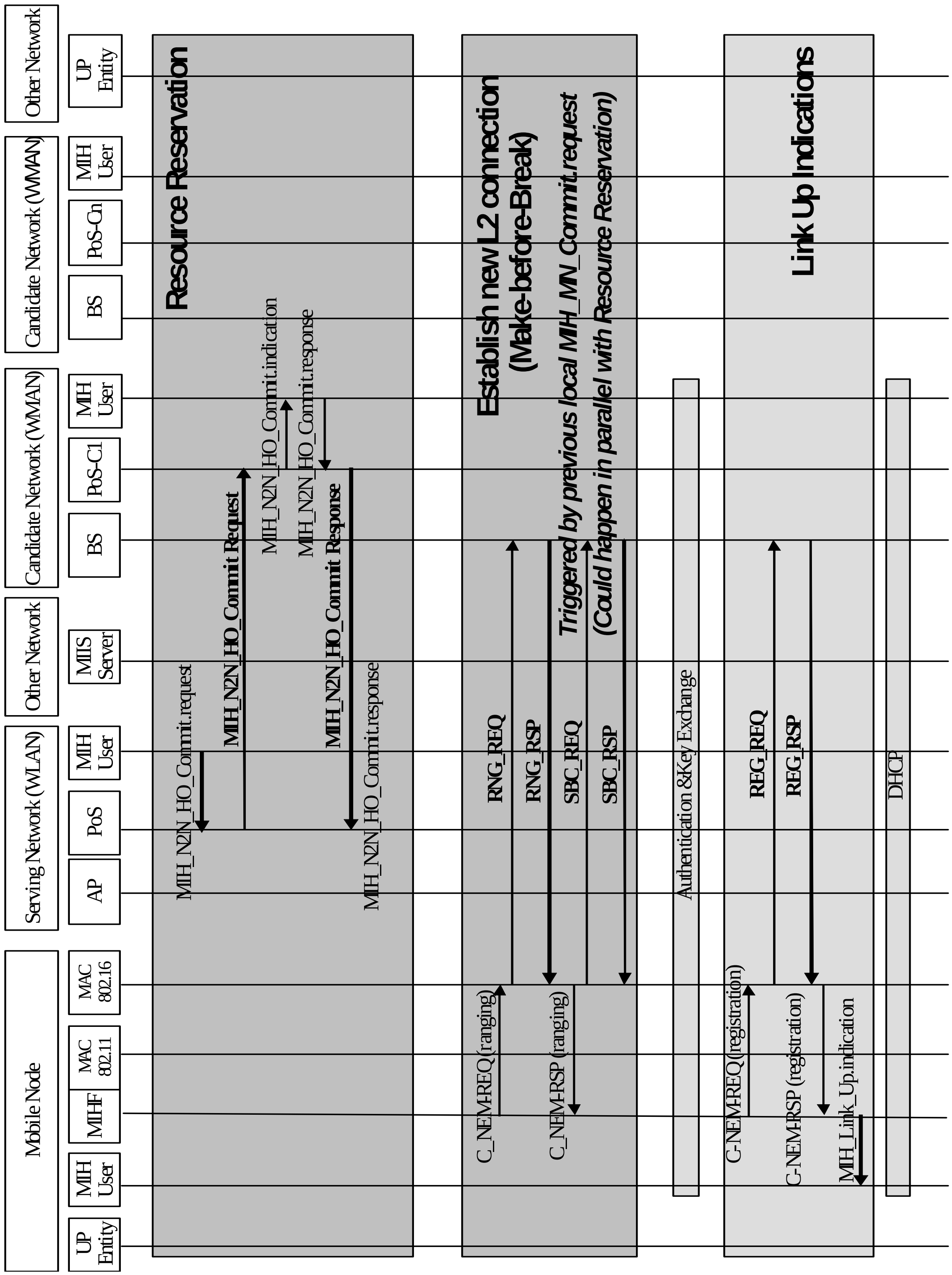


Figure H-7b—Example handover flow chart between 802.11 and 802.16 (cont.)

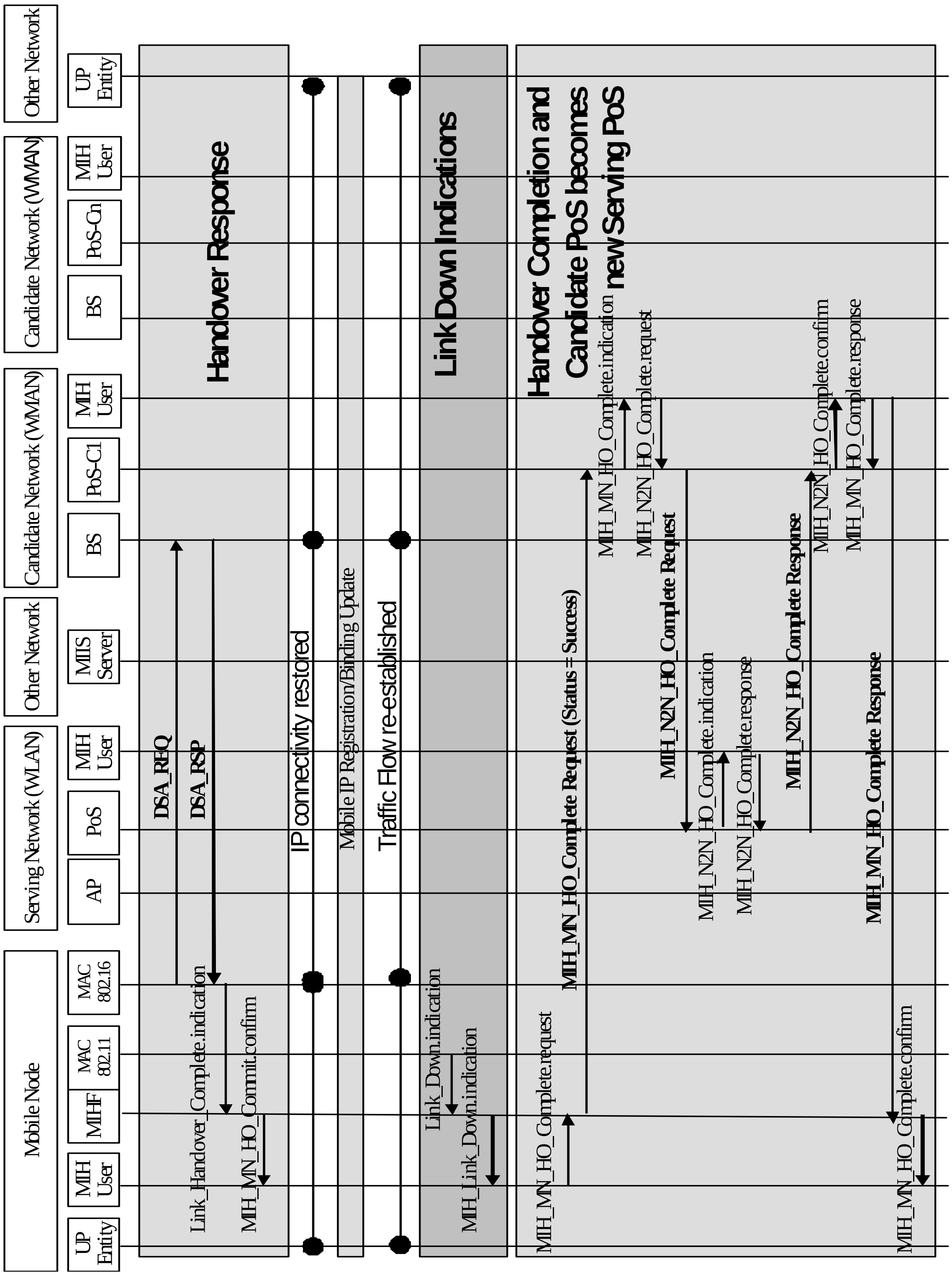


Figure H-7c—Example handover flow chart between 802.11 and 802.16 (cont.)

H.6 Example handover flow charts for Mobile IP

H.6.1 Mobile-initiated handover procedure

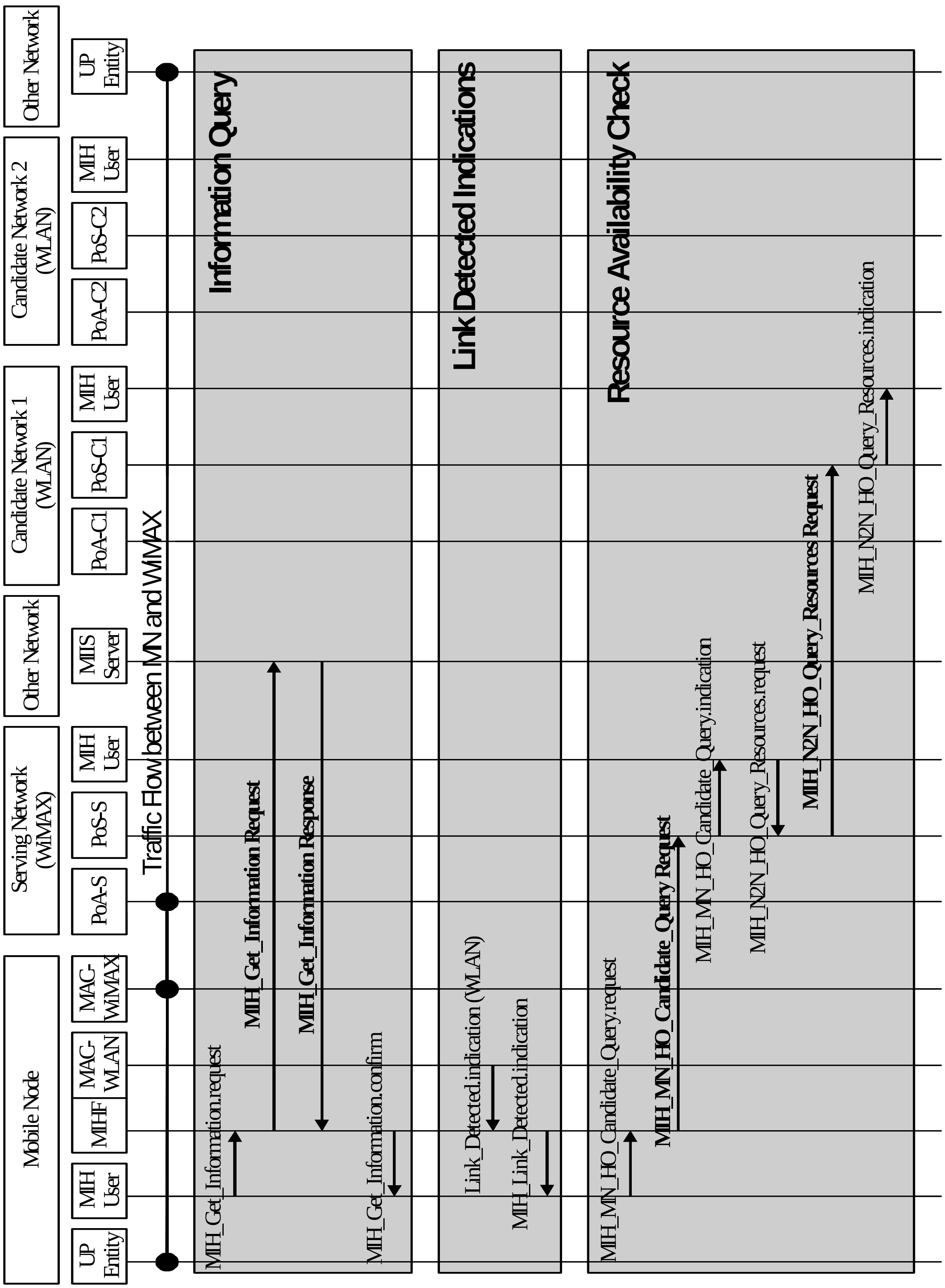


Figure H-8—Mobile-initiated handover procedure

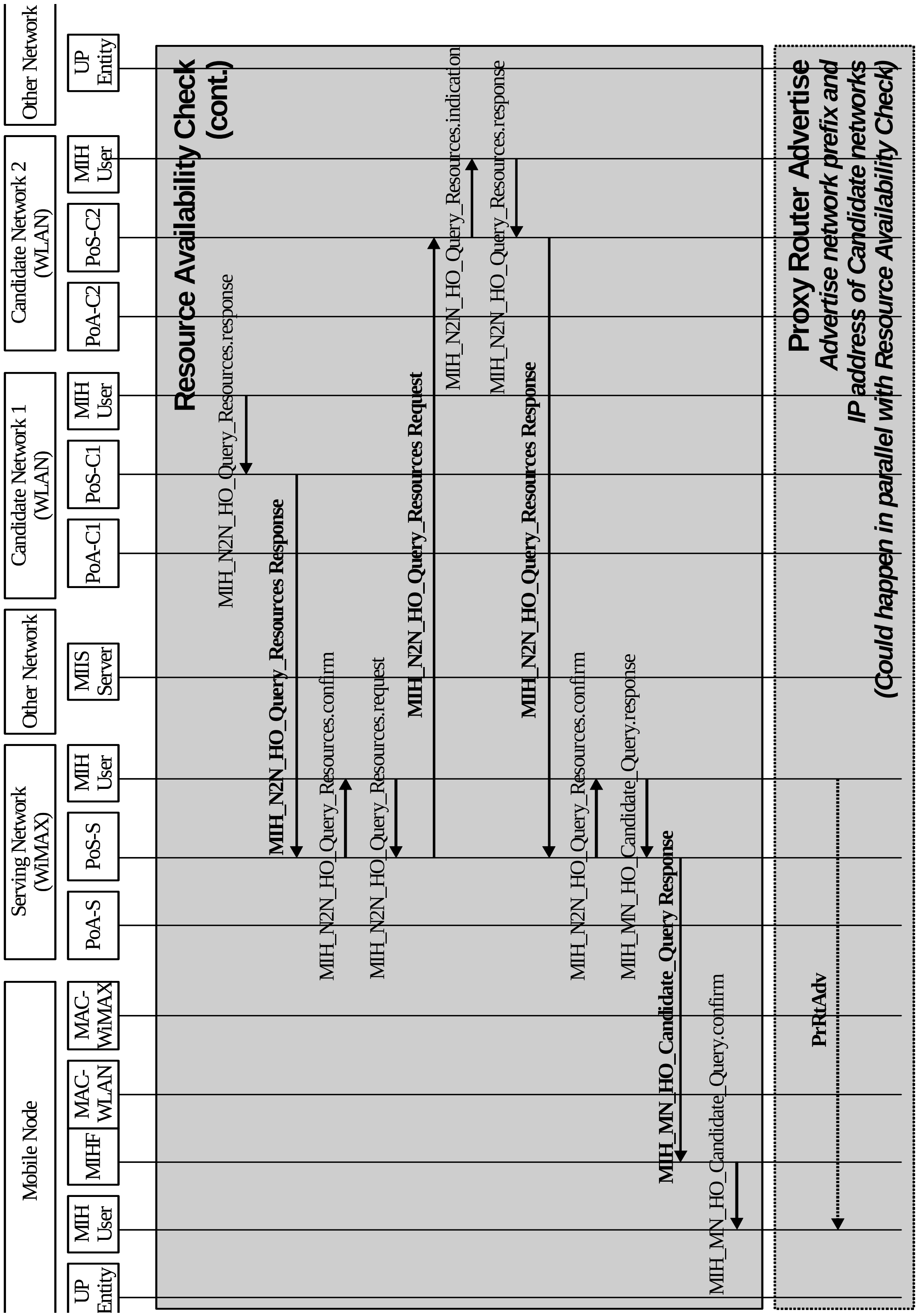


Figure H-8a—Mobile-initiated handover procedure (cont.)

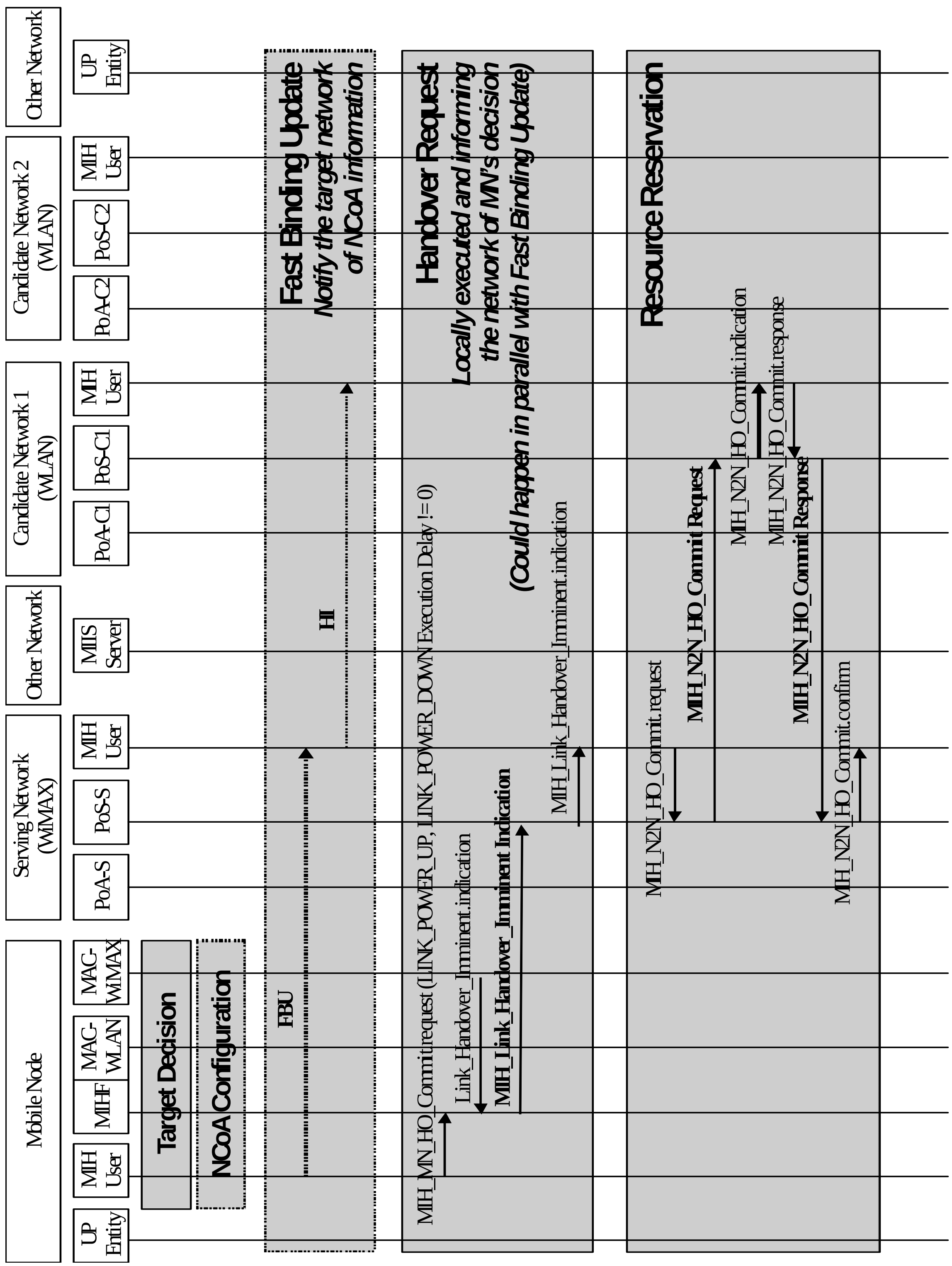


Figure H-8b—Mobile-initiated handover procedure (cont.)

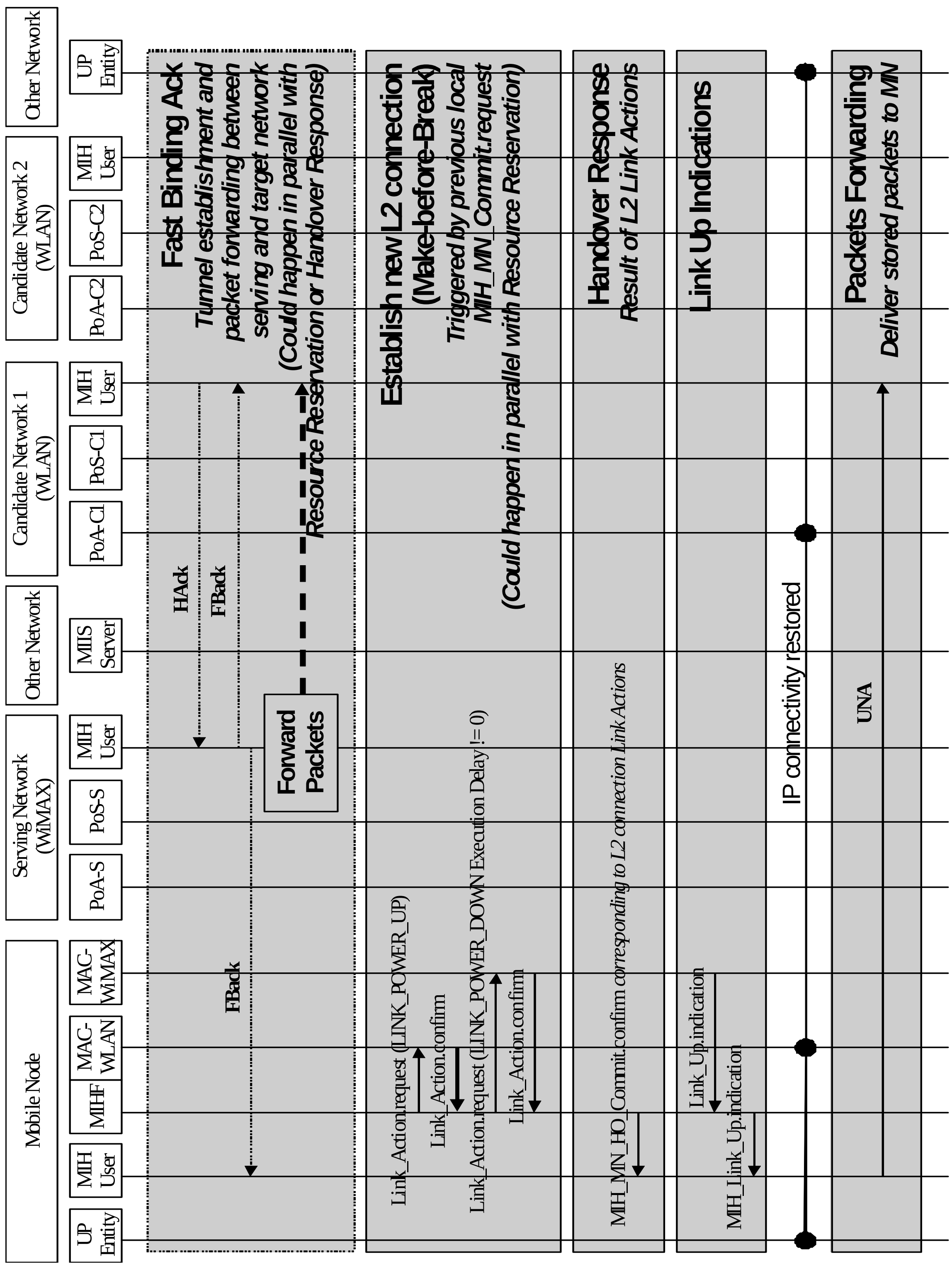


Figure H-8c—Mobile-initiated handover procedure (cont.)

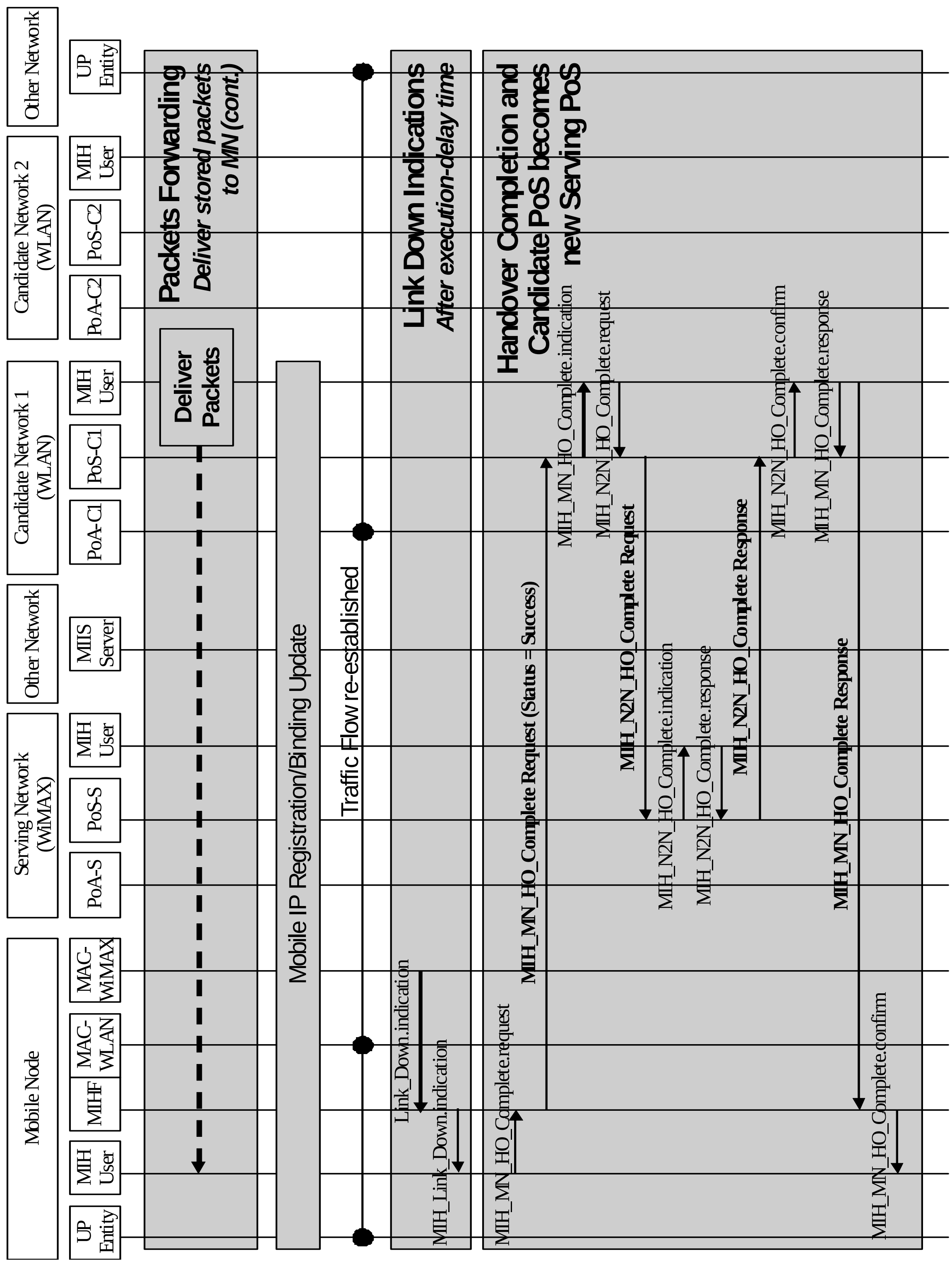


Figure H-8d—Mobile-initiated handover procedure (cont.)

H.6.2 Network-initiated handover procedure

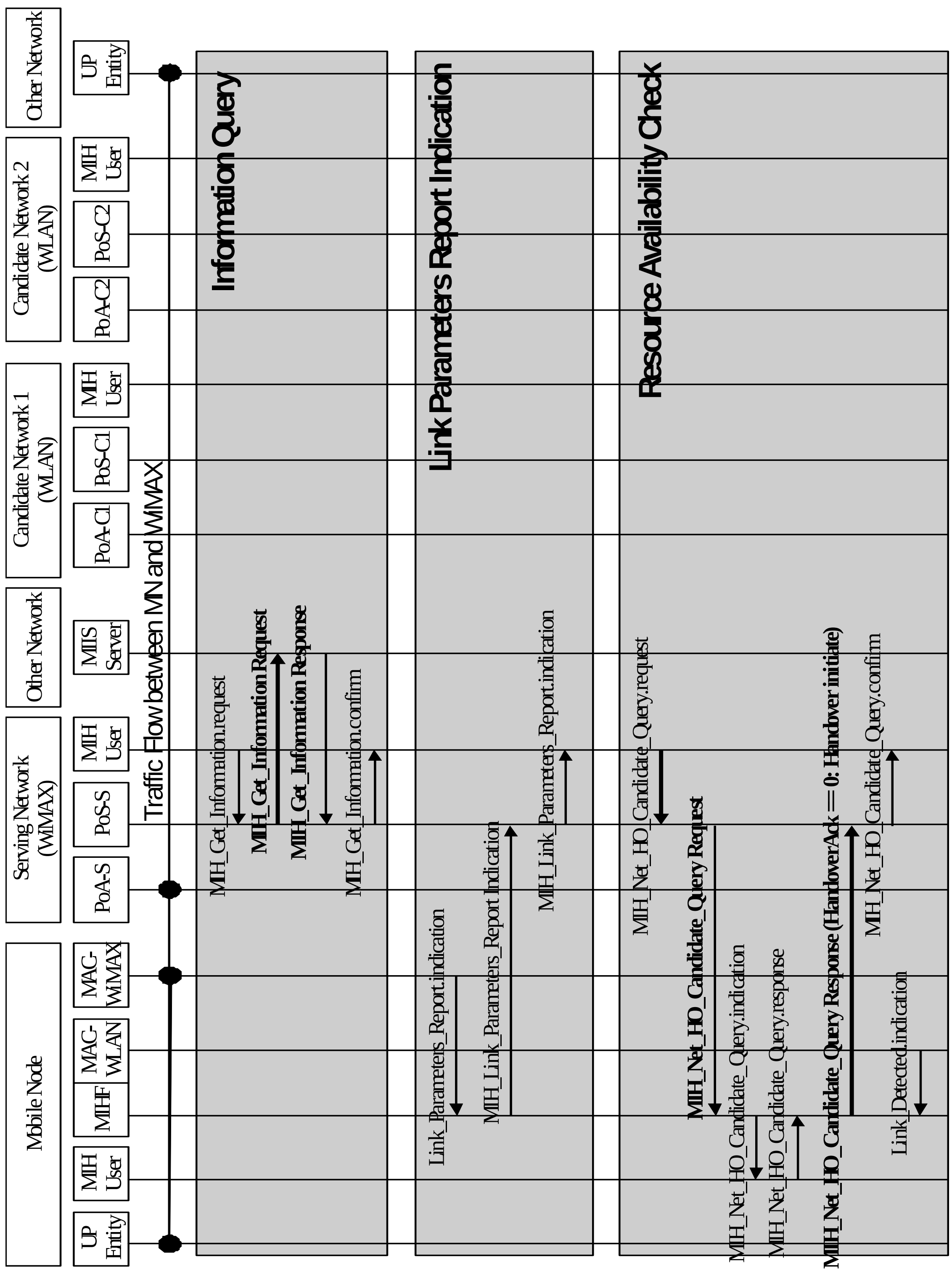


Figure H-9—Network-initiated handover procedure

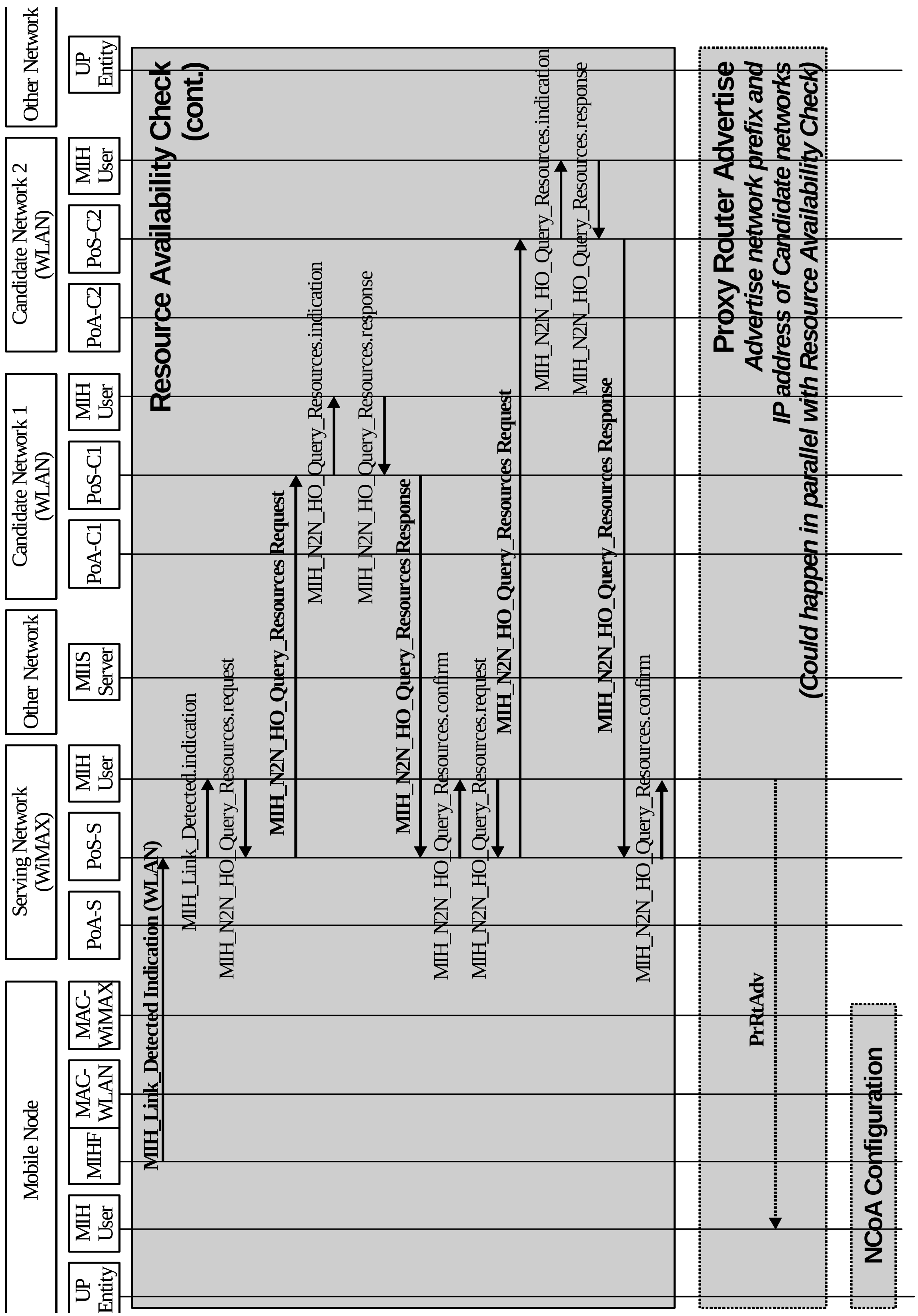


Figure H-9a—Network-initiated handover procedure (cont.)

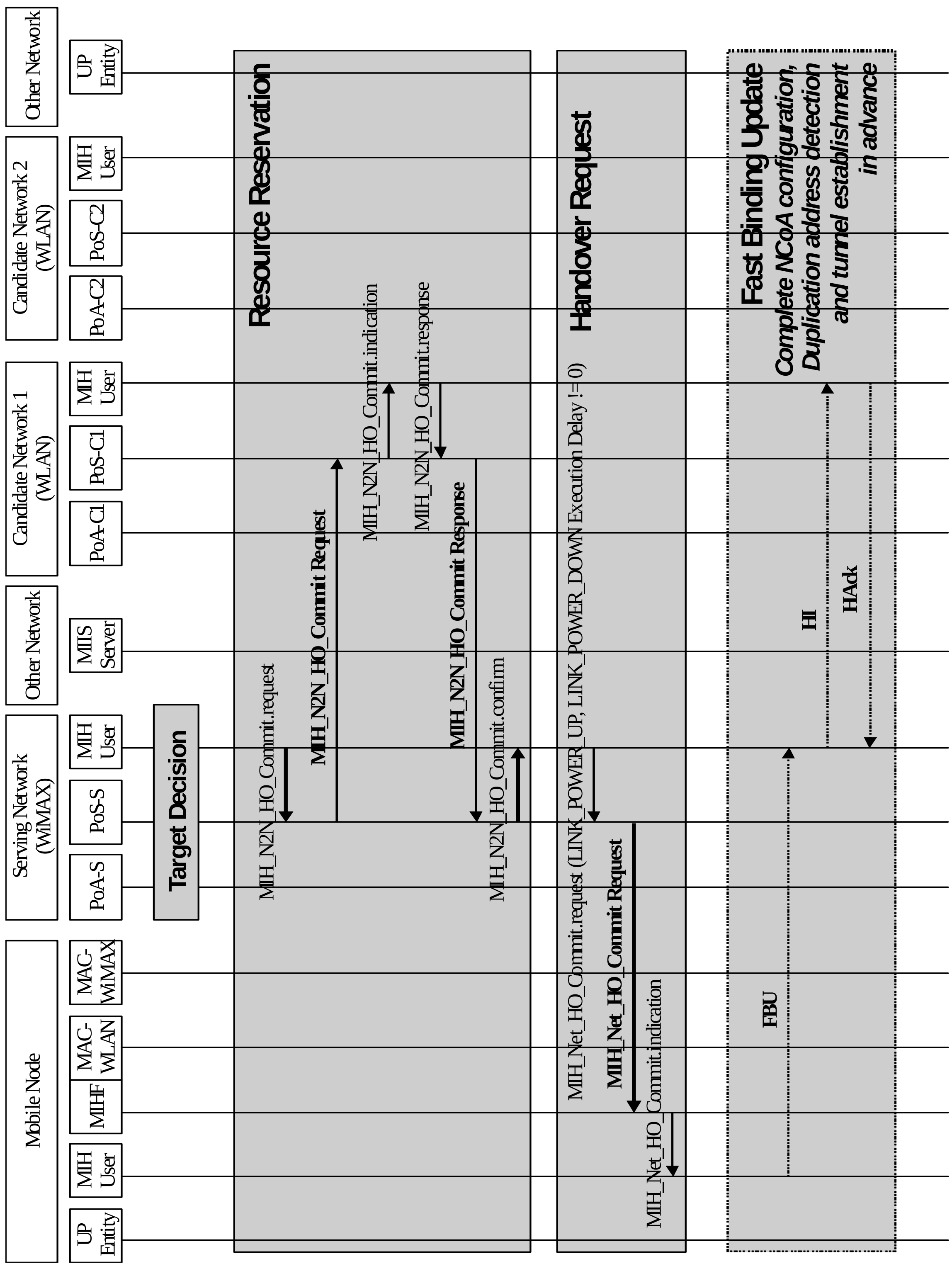


Figure H-9b—Network-initiated handover procedure (cont.)

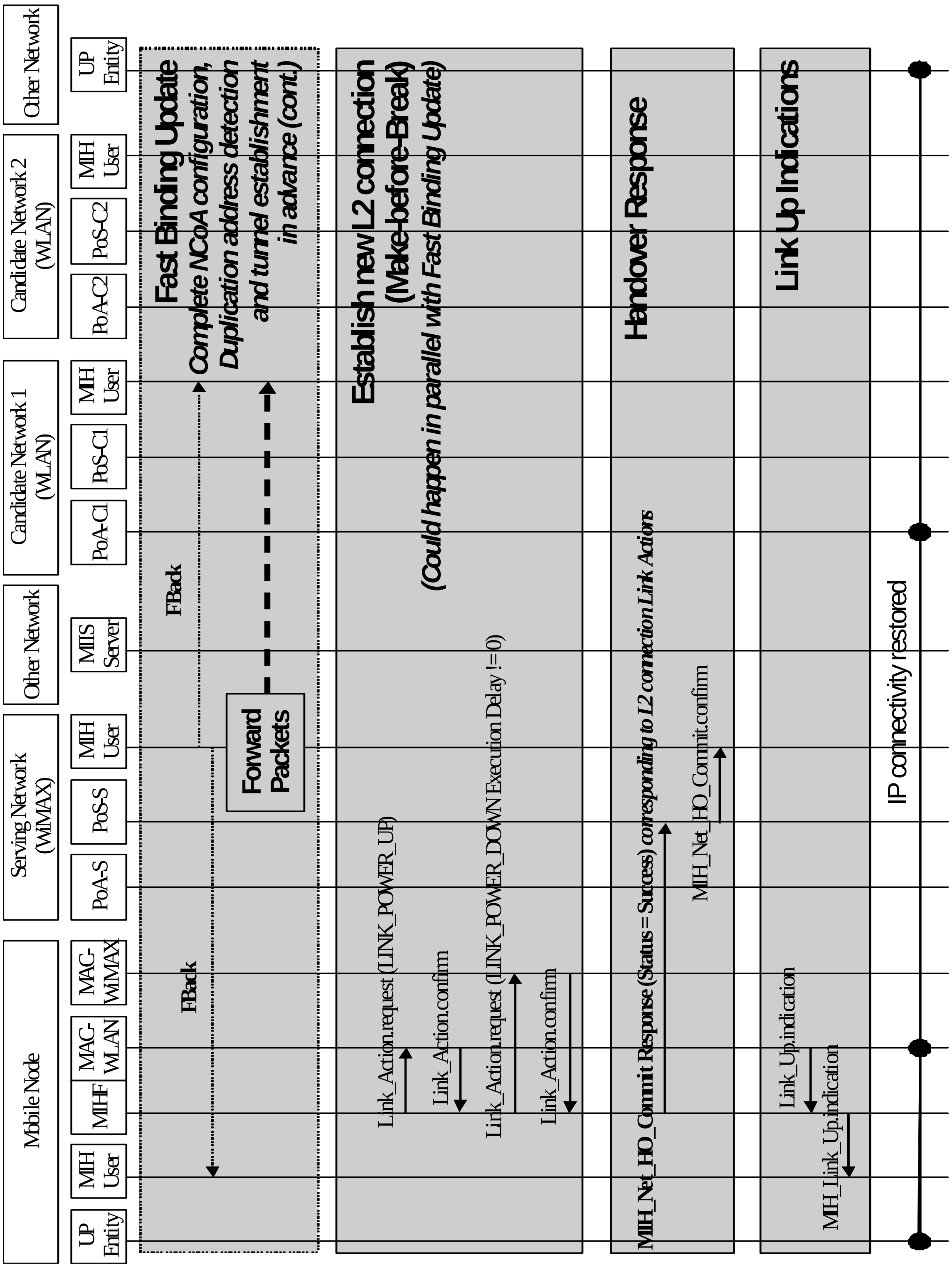


Figure H-9c—Network-initiated handover procedure (cont.)

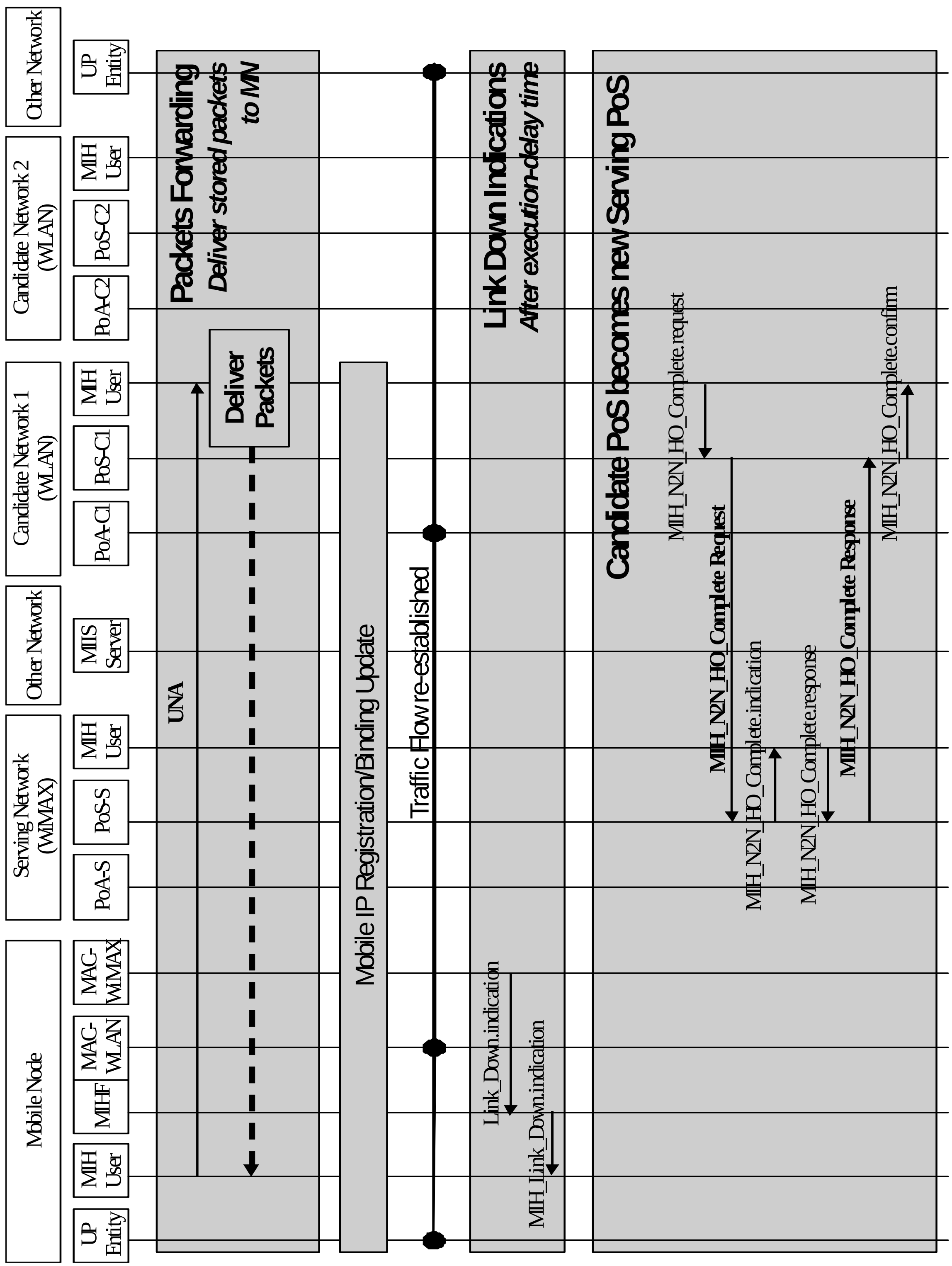


Figure H-9d—Network-initiated handover procedure (cont.)

H.7 Example handover flow chart for Proxy Mobile IPv6

H.7.1 Network-initiated handover procedures

Figure H-10 shows a network-initiated handover flow chart for Proxy Mobile IPv6 (PMIPv6), which is currently under standardization for supporting a local mobility in IETF NetLMM Working Group (Although the Proxy Mobile IP is under standardization, its overall flow is already defined. The following handover flow refers to the overall flow). The handover flow operates as follows:

1) MN receives packets through both Mobile Access Gateway (MAG) 1 located in the serving network and Local Mobility Anchor (LMA), which are primary components of the PMIPv6.

2) The Serving PoS queries the Information Server to get information about available neighboring networks.

3) The Serving PoS triggers a network-initiated handover by sending the MIH_Net_HO_Candidate_Query Request message to the MN. The MN responds through the MIH_Net_HO_Candidate_Query Response message, which contains MN's acknowledgement about the handover initiation and its preferred link and PoS lists.

4) The Serving PoS sends the MIH_N2N_HO_Query_Resource Request messages to different Candidate PoSs (can be more than one) to query the availability of the resource at candidate networks. The Candidate PoSs respond by sending the MIH_N2N_HO_Query_Resource Response message to the Serving PoS. The Serving PoS decides the handover target based on the resource availability information of candidate networks informed by the MIH_N2N_HO_Query_Resource Response message.

5) The Serving PoS informs the decided Target PoS (i.e. Candidate Network 1 in the Figure H-10, where MAG2 is located) of the handover commitment and requests the Target PoS to prepare resources for the incoming MN through sending the MIH_N2N_HO_Commit Request message. The Target PoS replies the result of the handover commitment and resource preparation by sending MIH_N2N_HO_Commit Response. Upon receiving the MIH_N2N_HO_Commit Request message, PMIPv6 client in the Target PoS queries the incoming MN's profile to a policy store such as an AAA server. As a result, the Target PoS obtains information for PMIP processes corresponding to the MN.

6) The Serving PoS requests MN to perform handover to the decided Target PoS by sending the MIH_Net_HO_Commit Request message. The MN replies the result of the handover commitment by sending MIH_Net_HO_Commit Response message.

7) Once the MN establishes Layer 2 connection to the Target PoS, PMIPv6 client in the Target PoS registers the current MN's location to LMA by sending Proxy Binding Update message. LMA updates its Binding Cache Entry with the Proxy Binding Update message and then replies with Proxy Binding Acknowledgement message.

8) After receiving the Proxy Binding Acknowledgement message, PMIPv6 clients sends Router Advertisement message to the MN. The Router Advertisement is constructed with the MN's information obtained from the policy server and LMA. It can be solicited by Router Solicitation message from the MN and periodically transmitted. MN configures IP addresses on its interface, which is currently used to connect to the Target PoS, with the received Router Advertisement message. Once the PMIPv6 procedures are completed, MN receives packets through both MAG 2 and LMA.

9) After the PMIPv6 execution, the Target PoS sends the MIH_N2N_HO_Complete Request message to the previous Serving PoS. The previous Serving PoS responds the message with MIH_N2N_HO_Complete Response.

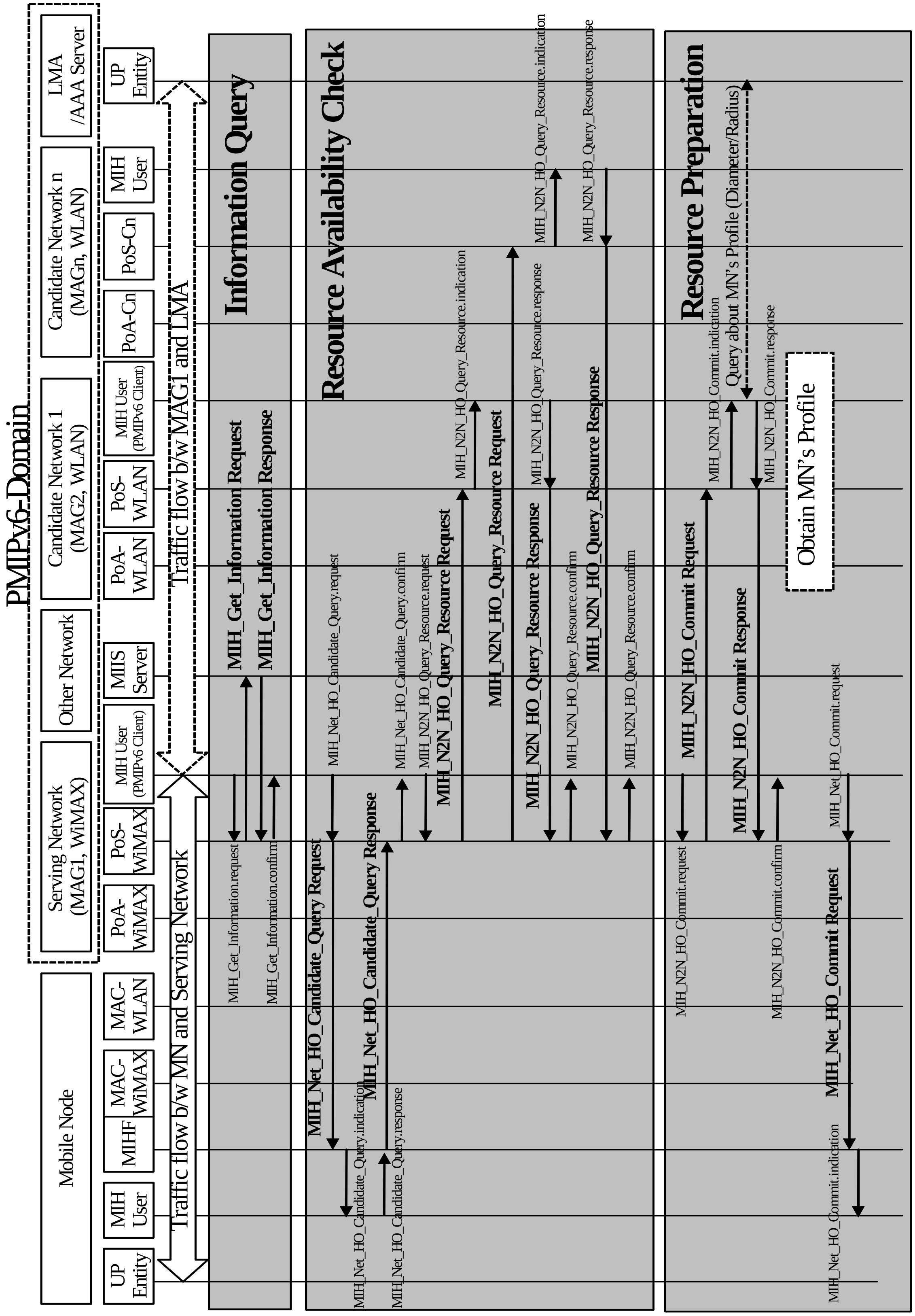


Figure H-10—Network-initiated handover procedures

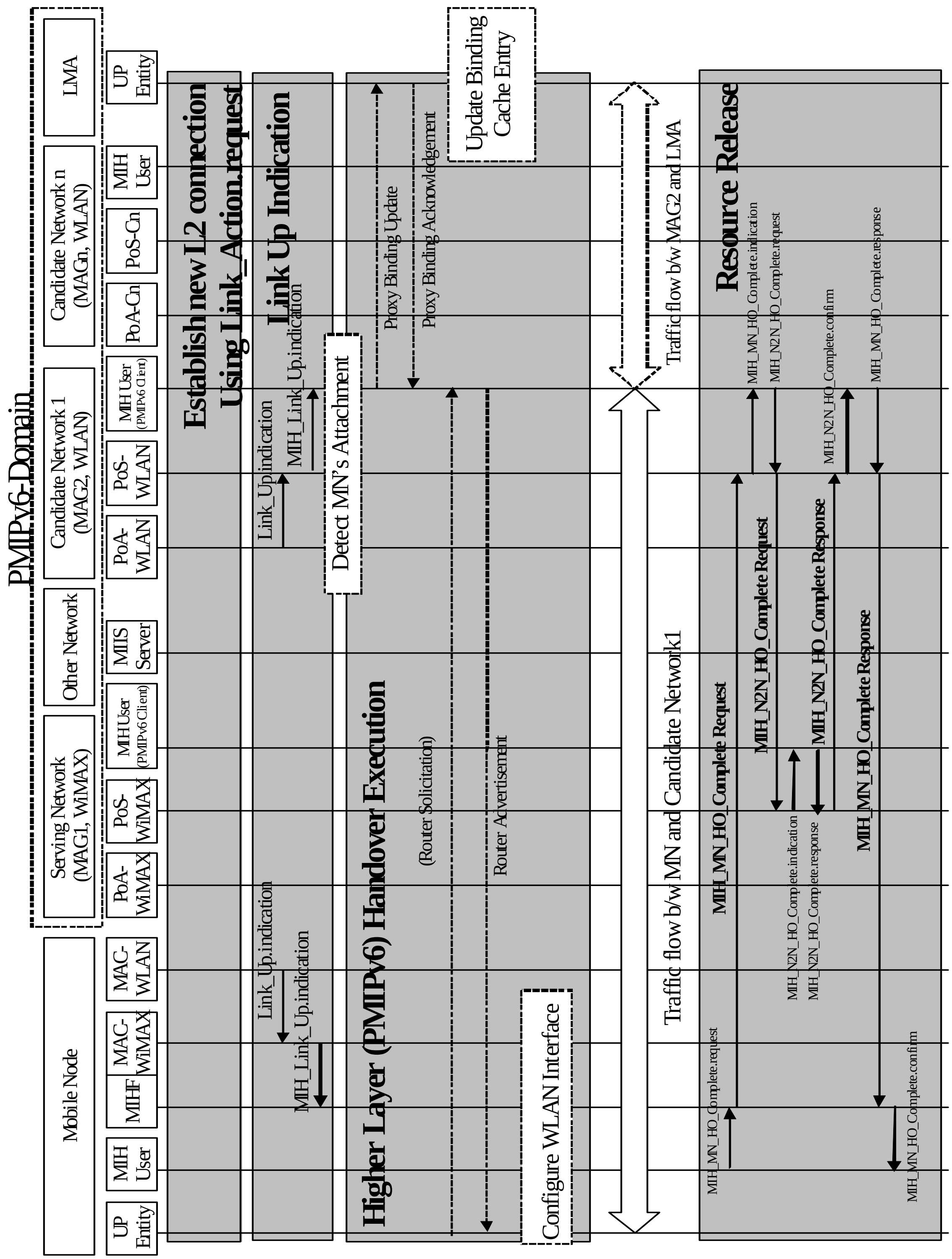


Figure H-10a—Network-initiated handover procedures (cont.)

H.7.2 Mobile-initiated handover procedures

Figure H-11 shows a mobile-initiated handover flow chart for Proxy Mobile IPv6 (PMIPv6), which is currently under standardization for supporting a local mobility in IETF NetLMM Working Group (Although the Proxy Mobile IP is under standardization, its overall flow is already defined. Following handover flow refers to the overall flow). The handover flow operates as follows:

1) MN receives packets through both Mobile Access Gateway (MAG) 1 located in the serving network and Local Mobility Anchor (LMA), which are primary components of the PMIPv6.

2) The MN queries the Information Server to get information about available neighboring networks. This information query may be attempted as soon as the MN attaches to a new serving network or periodically for refreshing the information.

3) MN sends the MIH_MN_HO_Candidate_Query Request message to the Serving PoS for triggering a mobile-initiated handover. This message contains requirements for potential candidate networks.

4) The Serving PoS sends the MIH_N2N_HO_Query_Resource Request messages to the informed Candidate PoSs (can be more than one) in order to query the availability of the resource at the candidate networks. The Candidate PoS responds by sending the MIH_N2N_HO_Query_Resource Response message to the Serving PoS. The Serving PoS in turn sends MIH_MN_HO_Candidate_Query Response message to the MN. Finally, the MN decides the handover target based on the result of query about resource availability at the candidate networks.

5) The MN performs handover to the specified network type and PoA by the MIH_MN_HO_Commit.request primitive. The Serving PoS, in turn, informs the decided Target PoS (i.e. Candidate Network 1 in the Figure H-11, where MAG2 is located) of the MN's handover commitment through sending the MIH_N2N_HO_Commit Request message. Upon receiving the MIH_N2N_HO_Commit Request message, PMIPv6 client as MIH User in the target PoS queries the incoming MN's profile to a policy store such as AAA server. As a result, the Target PoS obtains MN's information for PMIP processes in advance.

6) The Target PoS replies the Serving PoS with the result of the resource preparation by sending MIH_N2N_HO_Commit Response.

7) Once the MN establishes Layer 2 connection to the Target PoS, PMIPv6 client as MIH User in the Target PoS registers the current MN's location to LMA by sending Proxy Binding Update message. LMA updates its Binding Cache Entry with the Proxy Binding Update message and then replies with Proxy Binding Acknowledgement message.

8) After receiving the Proxy Binding Acknowledgement message, PMIPv6 clients sends Router Advertisement message to the MN. The Router Advertisement is constructed with the MN's information obtained from the policy server and LMA. It can be solicited by Router Solicitation message from the MN and periodically transmitted. MN configures IP addresses on its interface, which is currently used to connect to the Target PoS, with the received Router Advertisement message. Once the PMIPv6 procedures are completed, MN receives packets through both MAG 2 and LMA.

9) After the PMIPv6 execution, the Target PoS sends the MIH_N2N_HO_Complete Request message to the previous Serving PoS. The previous Serving PoS responds the message with MIH_N2N_HO_Complete Response.

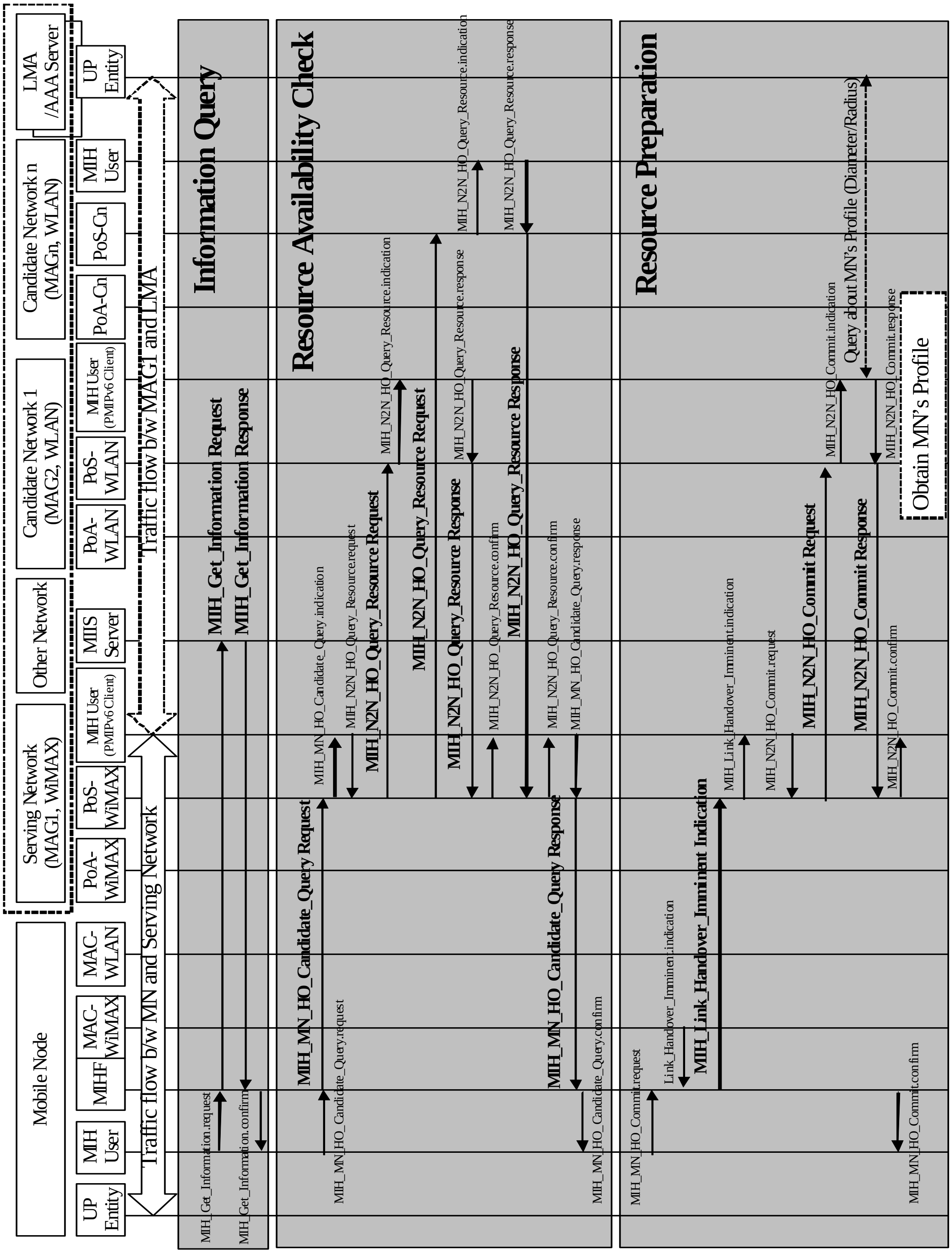


Figure H-11—Mobile-initiated handover procedures

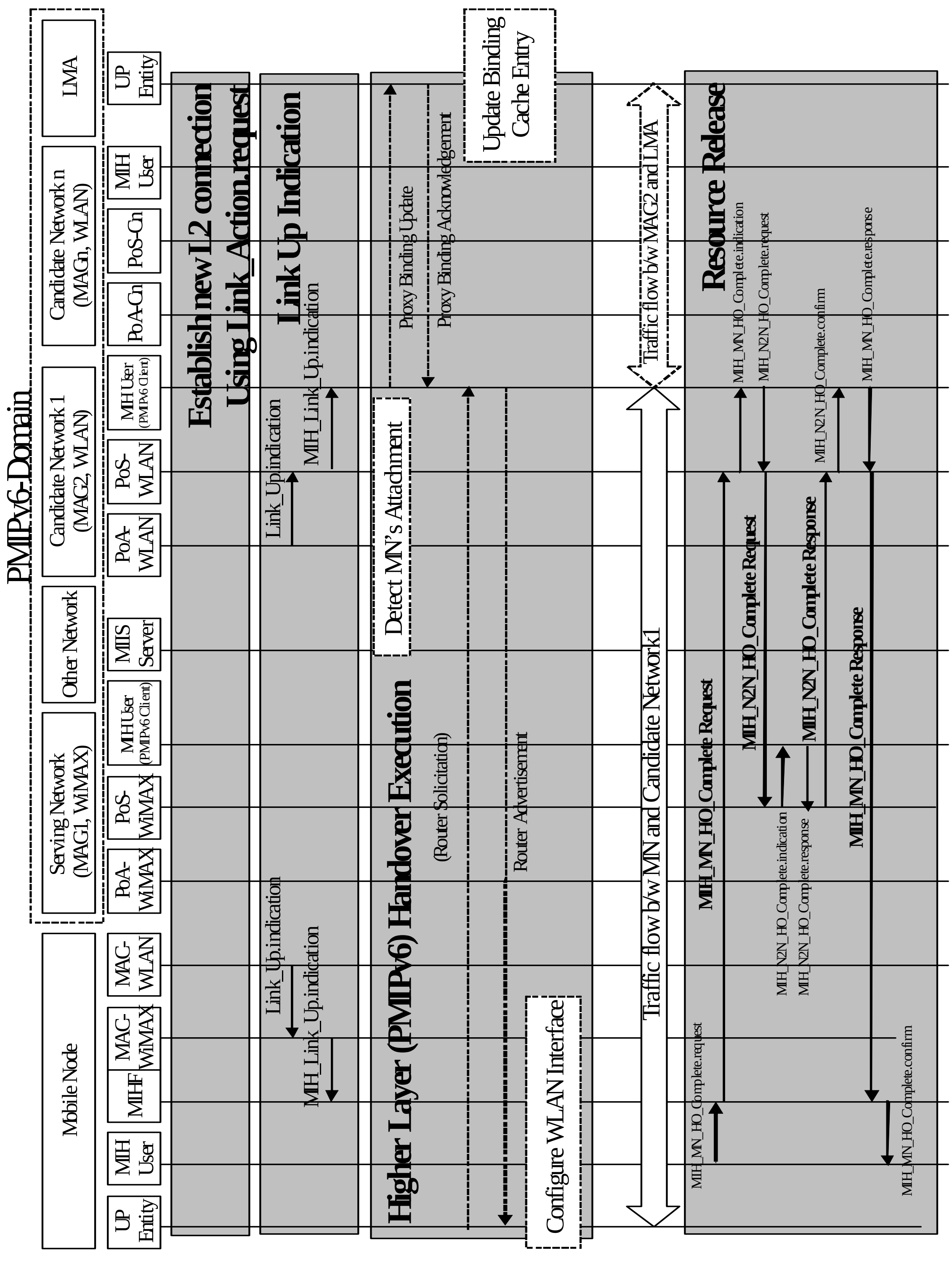


Figure H-11a—Mobile-initiated handover procedures (cont.)

H.8 Network Selection in 802.11 (WLAN) using 802.21

Figure H-12 shows the general topology of an 802.11 (WLAN) network operating with an 802.21 MIIS.

The steps in network selection as shown in Figure H-13 are as follows:

1) Pre-configuration: The AP is pre-configured with advertising protocol identifier (APID) of choice and is pre-configured to use 802.21 MIIS. The AP may discover the MIIS through a variety of different mechanisms that are outside the scope of specification. The maximum length of response messages from MIIS is also set. The AP may communicate with MIIS at L2 or at L3 using a protocol defined elsewhere.

2) Discover AP/Access Network Capabilities: The AP sends out a beacon with Inter-working IE set and APID set to GAS (Generic Advertisement Service). The STA may discover access network capabilities by listening to beacons or it could also send a probe request and discover access network capabilities through the probe response.

3) Query list of SSPNs: The STA sends out a query asking for list of available SSPNs. The query is defined using an 802.21 specific MIH frame. The MIH frame is then relayed by the AP to the MIIS. Meanwhile the AP sends out the initial GAS response to the STA with initial delay (comeback delay).

4) GAS response: The MIIS interprets the query and retrieves the response either from local or remote repository. It then packs the response in an appropriate MIH frame and sends it to the AP. Subsequently when the STA sends the GAS comeback request to the AP, the AP responds with the available information in the MIH frame. The STA then retrieves the information out of the MIH frame and obtains the answer to the query.

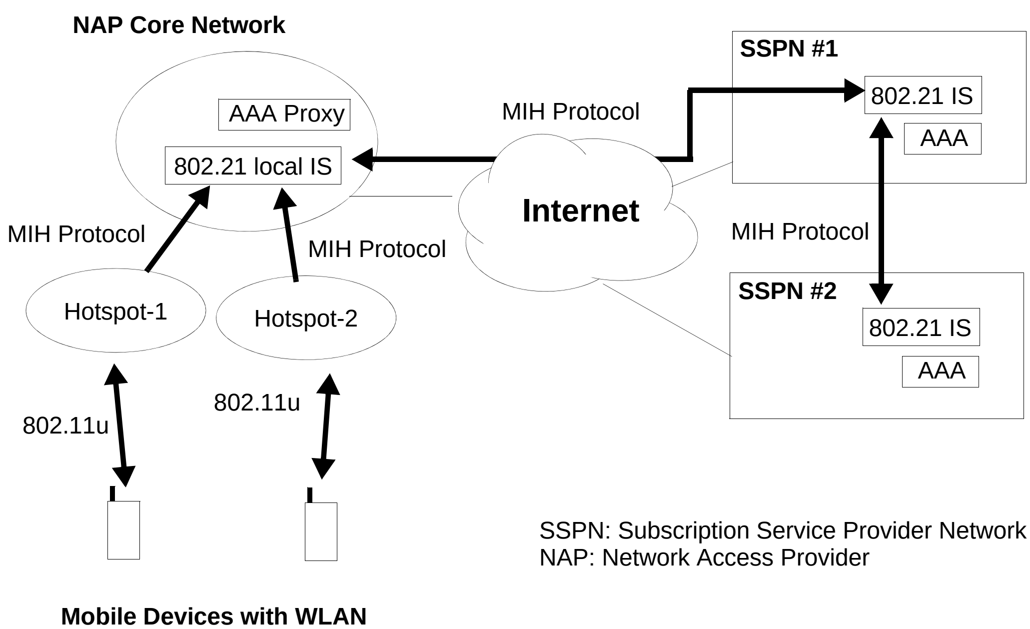


Figure H-12—Network selection in WLAN with 802.11 and 802.21

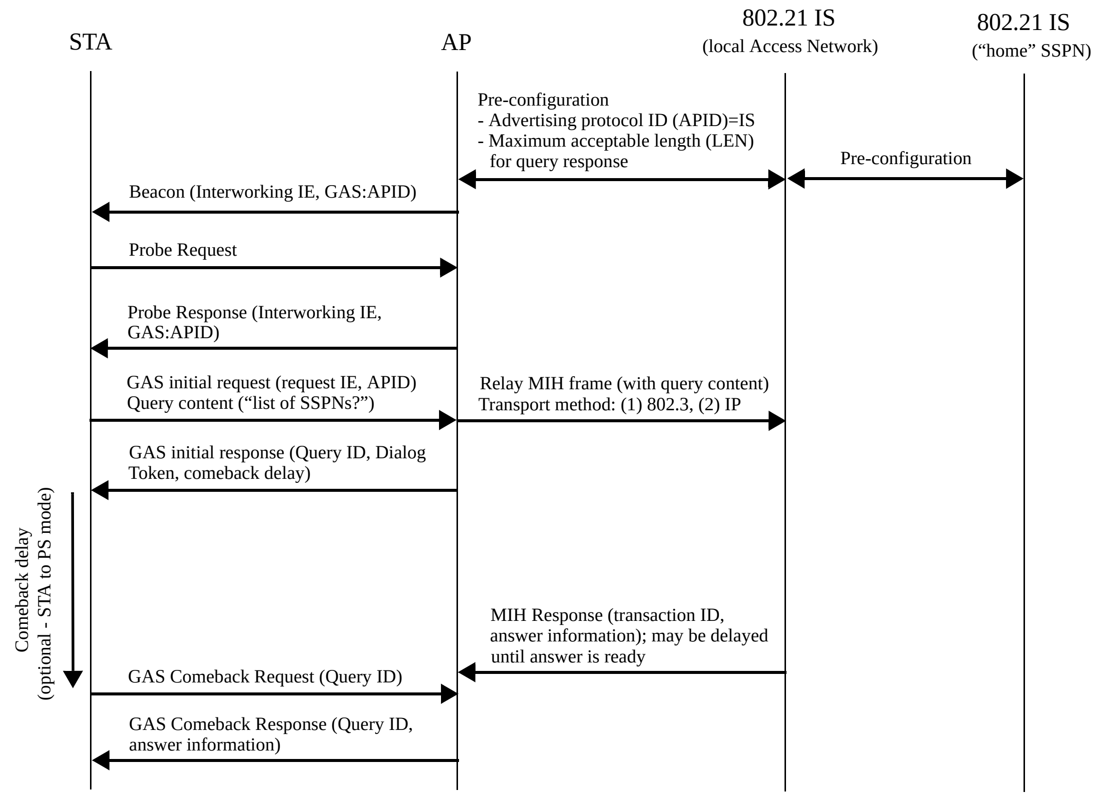


Figure H-13—Use case: query SSPN list

H.9 Handover flow diagram for abnormal cases

H.9.1 Abnormal case at the stage of resource availability check

When MIH_ MN_HO_Candidate_Query Response message indicates the handover initiator to abort handover, the MN will initiate handover process to another suitable target PoA. The following is the handover flow diagram for this case:

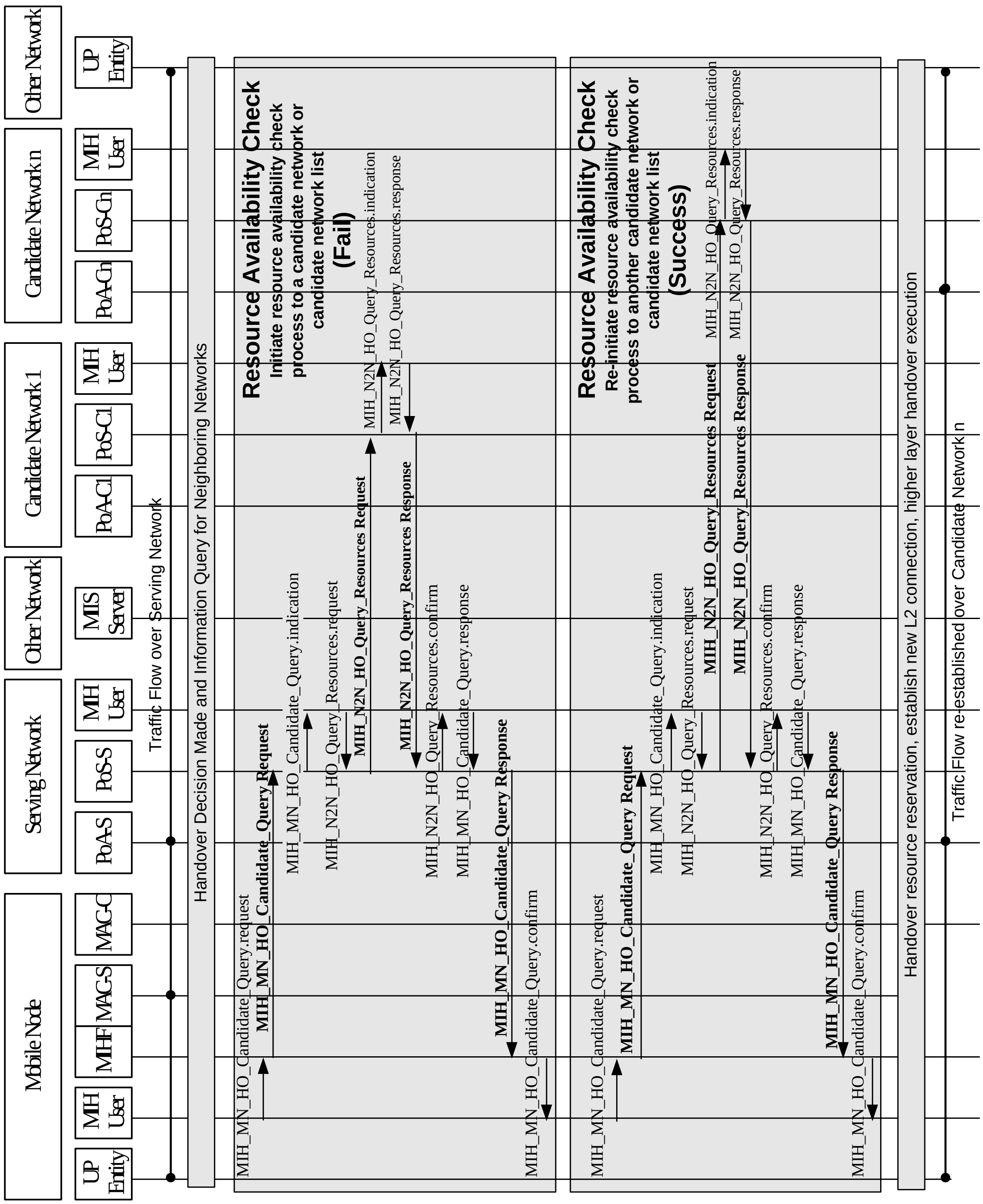


Figure H-14—Abnormal case at the stage of resource availability check

H.9.2 Abnormal case at the stage of handover resource reservation

When MIH_MN_HO_Commit Response indicates the handover initiator to abort handover (Status=Unsuccessful) the MN will issue MIH_MN_HO_Commit Request to another suitable target PoA, i.e. MN reuses the previous Resource Availability Query result and needs not repeat this Query process again. The following is the handover flow diagram for this case:

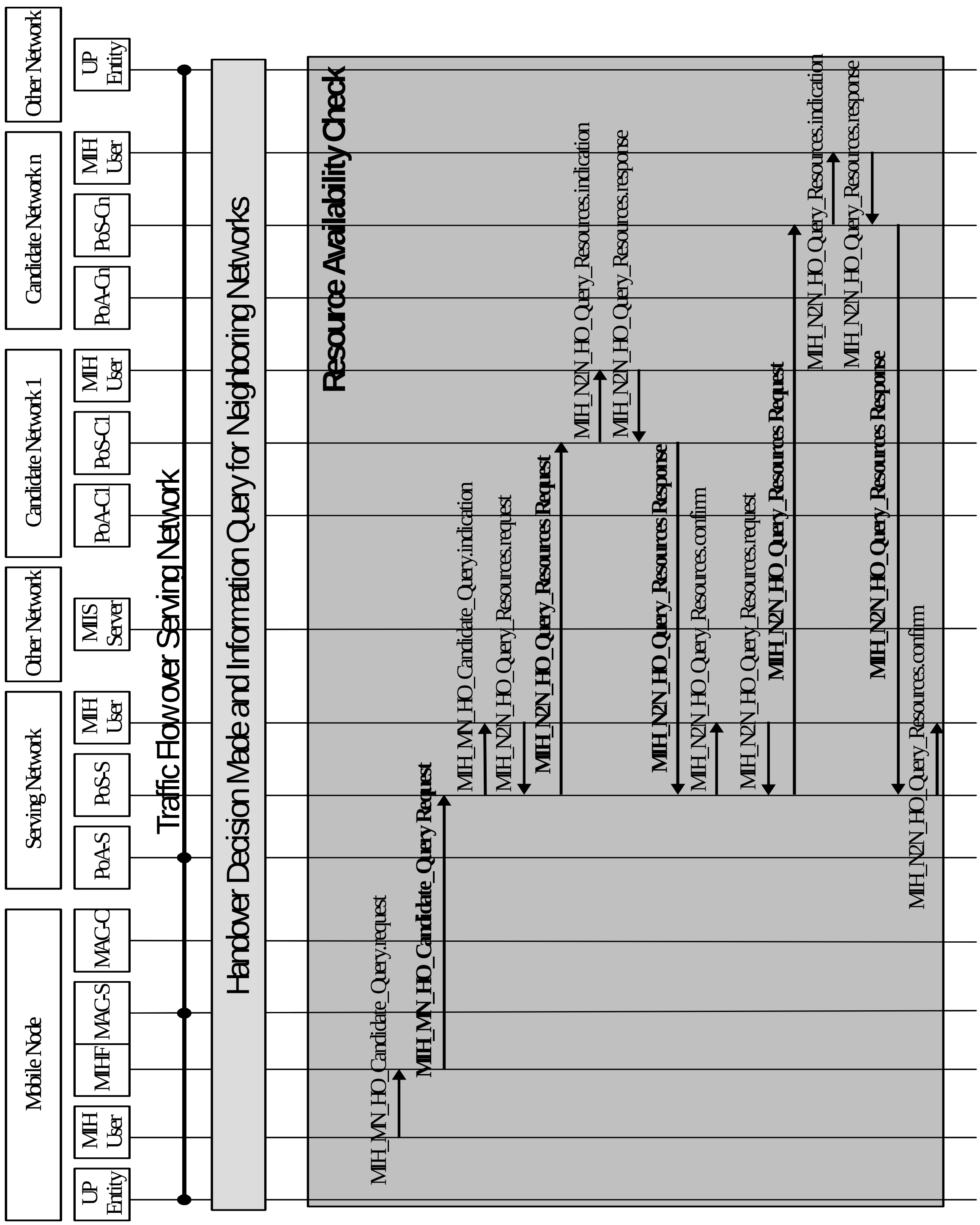


Figure H-15—Abnormal case at the stage of handover resource reservation

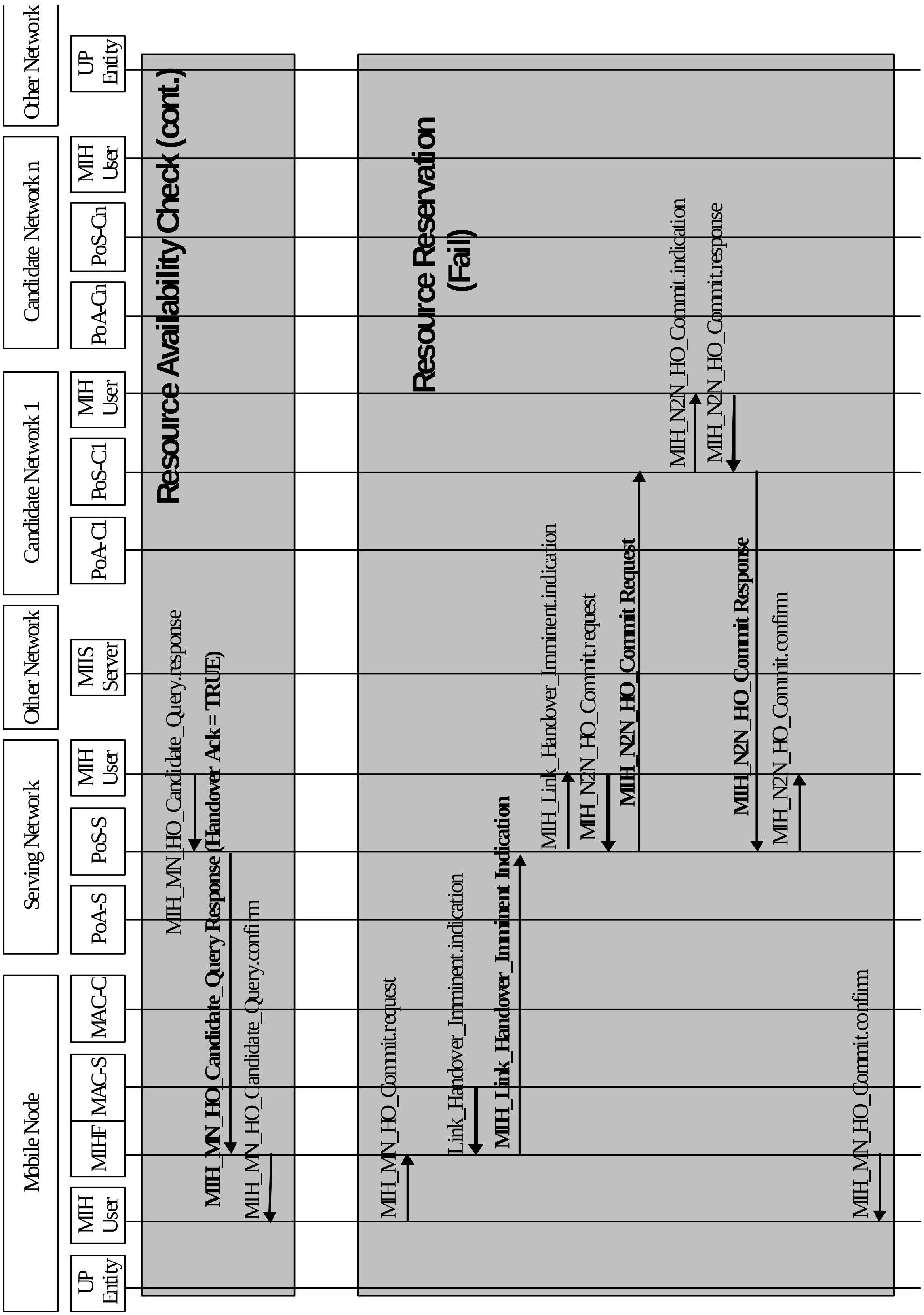


Figure H-15a—Abnormal case at the stage of handover resource reservation (cont.)

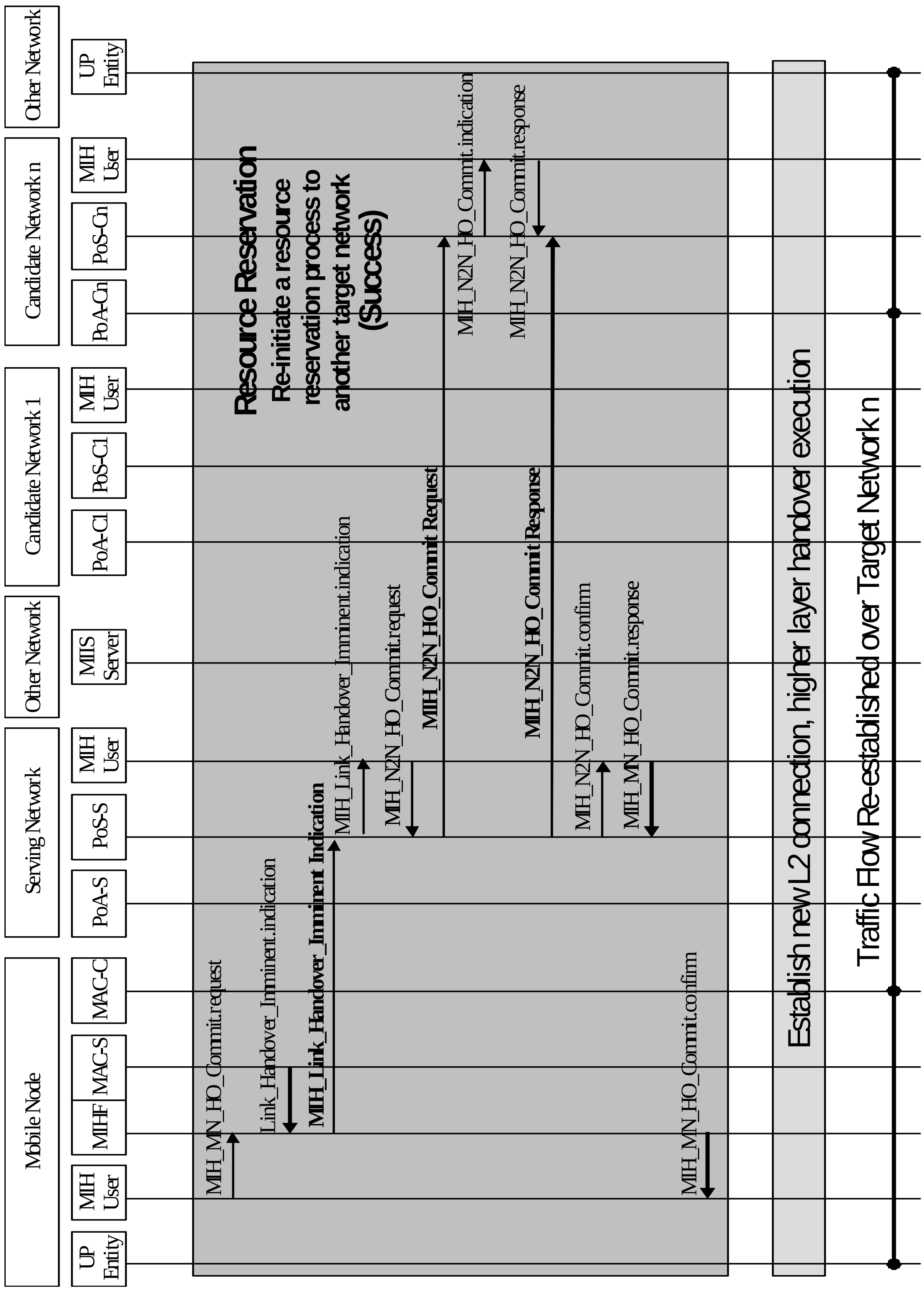


Figure H-15b—Abnormal case at the stage of handover resource reservation (cont.)

Annex I Mapping MIH messages to reference points

(normative)

Table I-1—Mapping MIH messages to reference points

MIH message name	Reference point
MIH_Capability_Discover	R1, R2, R3, R4, R5
MIH_Event_Subscribe	R1, R3
MIH_Event_Unsubscribe	R1, R3
MIH_Register	R1, R3, R5
MIH_DeRegister	R1, R3, R5
MIH_Link_Up	R1, R3, R5
MIH_Link_Down	R1, R3, R5
MIH_Link_Going_Down	R1, R3, R2
MIH_Link_Detected	R1, R3
MIH_Link_Parameters_Report	R1, R3, R5
MIH_Link_Event_Rollback	R1, R3
MIH_Link_Handover_Imminent	R1, R3, R2
MIH_Link_Handover_Complete	R1, R3
MIH_Get_Link_Parameters	R1, R3, R2
MIH_Link_Configure_Thresholds	R1, R3
MIH_Link_Actions	R1, R3
MIH_Net_HO_Candidate_Query	R1, R3
MIH_MN_HO_Candidate_Query	R1, R3
MIH_N2N_HO_Query_Resources	R5
MIH_Net_HO_Commit	R1, R3
MIH_N2N_HO_Commit	R5
MIH_MN_HO_Complete	R1, R2, R3

Table I-1—Mapping MIH messages to reference points

MIH message name	Reference point
MIH_N2N_HO_Complete	R5
MIH_Get_Information	R1, R2, R3, R4, R5

Annex J Quality of service examples

(normative)

This annex provides some examples for mapping the IEEE 802.21 QoS parameters with various sample technologies. Since most every access technology is designed for a specific need, it is not always designed with the same set of features that could be directly used by the MIHF. For this reason multiple media dependent interfaces exist and gleaning the available information and processing it for use as IEEE 802.21 QoS parameters has not been done in a standard way and is left as an implementation decision. This annex provides some examples of this for information. A flow diagram is provided that shows the setting and reporting of QoS parameters using the standard IEEE 802.21 primitives. Tables J-1 and J-2 below show possible mappings for the standard IEEE 802.21 QoS parameters with the access link technology specific parameters. Additionally an example is provided showing how the access link technology specific parameters may be used to derive the standard IEEE 802.21 QoS parameters.

A transmitted packet over a communication medium can experience the following outcomes:

— Be received with no errors at its intended destination
— Be received with errors at its intended destination
— Not be received in which case it is said that the packet is lost.

A communication medium represents one or multiple point-to-point network segments termed links in this specifications.

The maximum attainable speed of information transfer over a given communication channel can be constant, as it is usually the case with network segments involving only wired links, or it can be time varying at different scales, as is the case for segments involving wireless links. This measure will be called link throughput, for the purposes of this standard.

The ability of the link to provide accurate information transfer can be described via a statistical model characterized by the following parameters:

1. Minimum Packet Transfer Delay: is defined as the minimum delay over a population of interest
2. Average Packet Transfer Delay: is defined as the arithmetic mean of the delay over a population of interest
3. Maximum Packet Transfer Delay: is defined as the maximum delay over a population of interest
4. Jitter: is defined as the standard deviation of the delay over a population of interest
5. Packet Loss Rate: is defined as the ratio between the number of frames that are transmitted but not received and the total number of frames transmitted over a population of interest.
6. Packet Error Rate: is defined as the ratio between the number of packets that have been received with errors and the total number of packets present in a population of interest.

For a link that supports CoS differentiation, per CoS traffic accuracy parameters need to be maintained in order to provide insights on how individual traffic classes are faring.

In summary, the following set of parameters characterizes the speed and accuracy of the information transfer that a multi-CoS traffic link supports:

1. Link Throughput, the number of bits successfully received divided by the time it took to transmit them over the medium.

2. Link Packet Error Rate: representing the ratio between the number of frames received in error and the total number of frames transmitted in a link population of interest.
3. Supported Classes of Service: represents the maximum number of differentiable classes of service supported by this link.
4. Class of Service Parameters List: For each of the supported classes of service the following parameters are defined:
    a) Class Minimum Packet Transfer Delay: is defined as the minimum delay over a class population of interest
    b) Class Average Packet Transfer Delay: is defined as the arithmetic mean of the delay over a class population of interest
    c) Class Maximum Packet Transfer Delay: is defined as the maximum delay over a class population of interest
    d) Class Packet Delay Jitter: is defined as the standard deviation of the delay over a class population of interest
    e) Class Packet Loss Rate: is defined as the ratio between the number of frames that are transmitted but not received and the total number of frames transmitted over a class population of interest.

J.1 Generic IEEE 802.21 QoS flow diagram

Figure J-1 represents an example flow diagram for using the QoS framework defined by the MIHF. The MIH_Link_Configure_Thresholds primitive is used to set the application quality of service requirements and make it available to the MIHF. These parameters are mapped into media-specific measurements at the MIH layer and then used to configure the link parameter thresholds. While this mapping is not defined by these specifications, Table J-1 and Table J-2 provide examples of such mappings. The primitive MIH_Link_Parameters_Report is used to relay link specific measurements back to the MIH User.

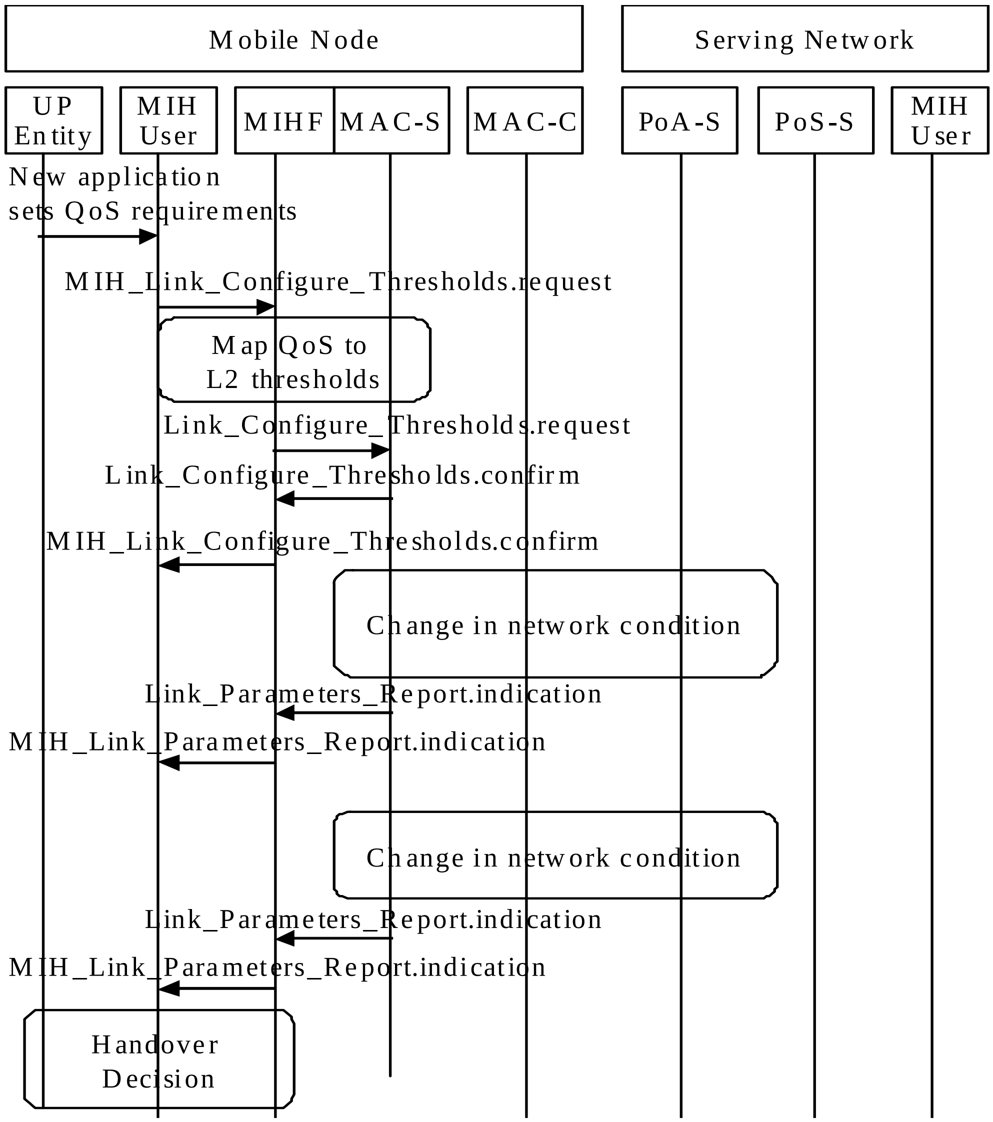


Figure J-1—An example flow for setting application QoS requirements

J.2 Generic IEEE 802.21 QoS parameter mappings

The tables provide example mappings of the standard IEEE 802.21 QoS parameters to the access link technology specific parameters. Table J-1 shows a mapping of IEEE 802.21 Link QoS Parameters and IEEE

802.11 (k) specific parameters. For IEEE 802.11, a collection of the QoS parameters may be on an individual station measurement basis, since this is a media using a distributed (symmetric) access technology.

Table J-1—Example QoS parameter mapping table for IEEE 802.11K

802.21 Link QoS Parameters	Related 802.11k parameters	802.11k IE Name	Note
Throughput	Channel utilization	STA Statistics Report	
Packet Error Rate	TransmittedFragmentCount MulticastTransmittedFrameCount FailedCount ReceivedFragmentCount * MulticastReceivedFrameCount FCSErrorCount * TransmittedFrameCount RetryCount MultipleRetryCount FrameDuplicateCount RTSSuccessCount RTSFailureCount ACKFailureCount	STA Statistics Report	
Supported number of COS	Available Admission Capacity	-	-
CoS Minimum Packet Transfer Delay	Transmit Delay Histogram *	QoS Metric Report	Trigger (Option) (only to specific STA)
CoS Average Packet Transfer Delay	APAverageAccessDelay AverageAccessDelayBestEffort AverageAccessDealyBackGround AverageAccessDelayVideo AverageAccessDealyVoice QoSTransmittedFragmentCount * QoSACKFailureCount *	STA Statistics Report	At an AP for BSS (only access delay)
	Average Queue Delay Average Transmit Delay *	QoS Metric Report	Trigger (Option) (only to specific STA)
	Average Access Delay for BE/Background/Video/Voice	BSS AC Access Delay	At an AP for BSS
CoS Maximum Packet Transfer Delay	Transmit Delay Histogram *	QoS Metric Report	Trigger (Option) (only to specific STA)
CoS Packet Delay Jitter	Transmit Delay Histogram* AverageTransmit Delay*	QoS Metric Report	Trigger(Option) (only to specific STA)

Table J-1—Example QoS parameter mapping table for IEEE 802.11K

802.21 Link QoS Parameters	Related 802.11k parameters	802.11k IE Name	Note
CoS Packet Loss Rate	QoSTransmittedFragmentCount * QoSFailedCount QoSRetryCount QoSMultipleRetryCount QoSFrameDuplicateCount QoSRTSSuccessCount QoSRTSFailureCount QoSACKFailureCount * QoSReceivedFragmentCount QoSTransmittedFrameCount QoSDiscardedFrameCount QoSMPDUsReceivedCount QoSRetriesReceivedCount	STA Statistics Report	
	Transmitted MSDU Count * MSDU Discarded Count MSDU Failed Count* MSDU Multiple Retry Count QoS CF-polls Lost Count	QoS Metric Report	Trigger (Option) (only to specific STA)

Note: (*) indicates that the parameters may be most likely used to directly derive IEEE 802.21 LinkQoSParameters. See E.3 for example derivations.

Table J-2 and Table J-3 show example mappings for IEEE 802.21 QoS link parameters and other link specific parameters for IEEE 802.16, 3GPP, and 3GPP2. For these technologies control is usually by means of a base station, not an individual station, since the media is controlled using asymmetric access.

Table J-2—An example of a QoS parameter mapping table

802.21 Link QoS Parameters	802.16	3GPP2
Throughput	Maximum Sustained Traffic Rate	Peak_Rate
Packet Loss Rate		Max_IP_Packet_Loss_Rate
Packet Error Rate	Packet Error Rate	
CoS Minimum Packet Transfer Delay		
CoS Average Packet Transfer Delay		
CoS Maximum Packet Transfer Delay	Maximum Latency	Max_Latency
CoS Packet Delay Jitter	Tolerated Jitter	Delay_Var_Sensitive

Table J-3—Example QoS parameter mapping table for 3GPP

802.21 Link QoS Parameters	Related 3GPP Parameters			
Supported number of CoS	4			
	Conversational	**Streaming**	**Interactive**	**Background**
Throughput	Peak throughput			
	Mean throughput			
	Maximum bit rate for uplink/downlink			
	Guaranteed bit rate for uplink/downlink			
Link Packet Error Rate	SDU Error Ratio			
	Residual Bit Error Rate			
CoS Minimum Packet Transfer Delay	Transfer delay			
CoS Average Packet Transfer Delay	Transfer delay			
CoS Maximum Packet Transfer Delay	Maximum Transfer delay			
CoS Packet Transfer Delay Jitter		Delay Variation		
CoS Packet Loss Rate	Residual Bit Error Rate			
	SDU Error Ratio			

J.3 Example of deriving generic IEEE 802.21 QoS parameters

This gives an example of generic IEEE 802.21 QoS parameter derivation from IEEE 802.11 link measurement parameters. Examples provided in this section rely on incremental values for the counters specified in the IEEE 802.11 specifications. Incremental counter values are the changes of the counter values over a time interval.

J.3.1 Packet loss rate

To calculate the packet loss rate (PLR), one uses the following equation.

$$\mathrm{PLR} = \frac{\text{the number of lost packets}}{\text{the number of transmitted packet (successful + failed)}}$$

Two kinds of PLRs exist based on the following IEEE 802.11 interpretation of packets.

i) packet = MPDU (MAC sublayer fragmented frame)

ii) packet = MSDU (MAC user packet)

The PLR_{MPDU} can be derived from the **STA Statistics Report information element** using the following equation.

$$PLR_{MPDU} = \frac{\text{ACK failure MPDUs}}{\text{acknowledged MPDUs} + \text{ACK failure MPDUs}}$$
$$= \frac{\text{ACKFailureCount}}{\text{TransmittedFragmentCount} + \text{ACKFailureCount}}$$

The PLR_{MSDU} can be derived from the **QoS Metric information element** using the following equation.

$$PLR_{MSDU} = \frac{\text{failed MSDUs}}{\text{Transmitted MSDUs} + \text{Failed MSDUs}}$$
$$= \frac{\text{MSDUFailedCount}}{\text{TransmittedMSDUCount} + \text{MSDUFailedCount}}$$

If link layer specific information is needed, then PLR_{MPDU} would be used as the IEEE 802.21 PLR. If information about the interface is needed, then PLR_{MSDU} would be used as the IEEE 802.21 PLR.

J.3.2 Packet error rate

The packet error rate (PER) can be calculated using the following equation.

$$PER = \frac{\text{the number of packets that are received with errors}}{\text{the number of packets in a population of interest}}$$

Unlike for PLR, this parameter is only defined for the IEEE 802.11 MPDU. The PER can be derived from the **STA Statistics Report information element** using the following equation.

$$PER = \frac{\text{FCSErrorCount}}{\text{ReceivedFragmentCount} + \text{FCSErrorCount}}$$

J.3.3 Average transfer delay

There exist three possible definitions for delay in IEEE 802.11 for what could be used to map to the IEEE 802.21 meaning.

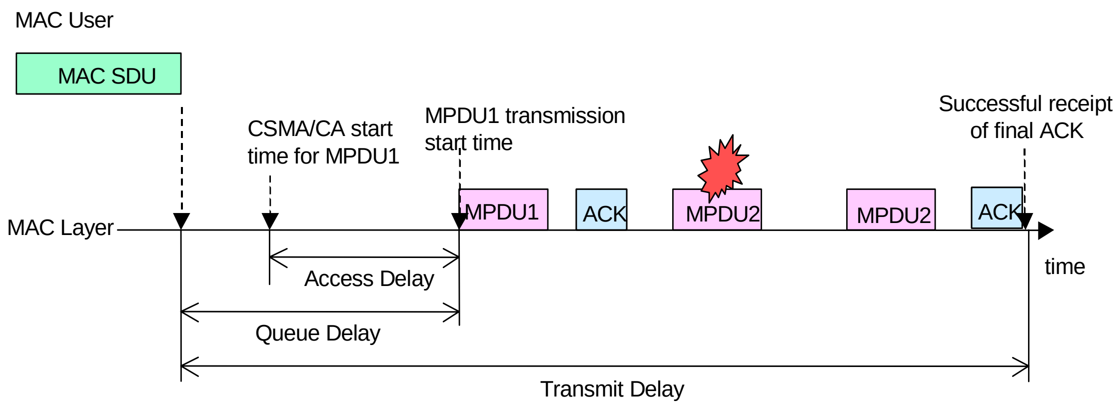
i) Transmit delay (MSDU delay): The delay shall be measured from the time the MSDU is passed to the MAC sublayer until the point at which the entire MSDU has been successfully transmitted including receipt of the final ACK.

ii) Access delay (MPDU delay)

The delay measured from the time the DCF- or EDCA-MPDU is ready for transmission (i.e., begins CSMA/CA access) until the actual frame transmission start time.

iii) Queue delay (MSDU delay)

Queue delay shall be measured from the time the MSDU is passed to the MAC until the point at which the first or only fragment is ready for transmission (i.e., until the transmission starts).



If the average MSDU transmit delay is used for the IEEE 802.21 average transfer delay, it can be derived from **QoS Metric information element**.

$$\begin{aligned} ATD_{MSDU} &= \text{Average MSDU Transmit Delay} \\ &= \text{Average Transmit Delay} \end{aligned}$$

If the MIHF wants to get the average MPDU packet transmit delay, then the following equation can be used (parameters from **STA Statistics Report information element** and **QoS Metric information element**.

$$\begin{aligned} ATD_{MPDU} &= \text{Average MPDU Transmit Delay} \\ &= \frac{\sum \text{Transmit Delay}}{\text{TransmittedFragmentCount} + \text{ACKFailureCount}} \end{aligned}$$

If link layer specific information is needed, then ATD_{MPDU} would be used as the IEEE 802.21 ATD. If information about the interface is needed, then ATD_{MSDU} would be used as the IEEE 802.21 ATD.

J.3.4 Packet transfer delay jitter

Using the IEEE 802.21 definition of "the standard deviation of the delay over a population of interest," the IEEE 802.11 MAC sublayer provides the **QoS Metric information element** and measurement parameters to calculate the standard deviation of delay.

- QoS Metric information element includes:
  - Transmit Delay Histogram and
  - Average Transmit Delay parameters

Variance calculation using discrete density function is given as

$$VAR(X) = \sum_{=1}^{N} P_i \left(x_i - \bar{X}\right)^2$$

Therefore, the packet transfer delay jitter for MSDU level is

$$\text{Packet Transfer Delay Jitter} = \text{MSDU Packet Transmit Delay Jitter}$$

$$= \sqrt{\sum_{=1}^{N} P_i \left(x_i - \text{AverageTransmitDelay}\right)^2}$$

where,

N= the number of bins of Transmit Delay Histogram,

P_i= the value (measured percentile) of i-th bin of Transmit Delay Histogram, and

x_i=the mean value of the delay range of i-th bin.

Annex K Media specific mapping for SAPs

(normative)

The MIHF aggregates disparate interfaces with respective media dependent lower-layer instances (media dependent service access points) into a single interface with the MIH Users (the MIH SAP), reducing the inter-media differences to the extent possible.

The MIHF features media dependent interfaces with IEEE 802 link-layer technologies (IEEE 802.2, IEEE 802.3, IEEE 802.11, and IEEE 802.16) and cellular technologies (3GPP and 3GPP2). The MIHF for the most part uses existing primitives and functionality provided by different access technology standards. Amendments to existing standards are recommended only when deemed necessary to fulfill the MIHF capabilities.

The following subclauses list general amendments recommended to different underlying access technology standards due to the enhanced heterogeneous handover capability provided by MIHF.

K.1 MIH_LINK_SAP mapping to specific technologies

Table K-1—MIH_Link_SAP/802.16 primitives mapping

MIH_LINK_SAP Primitive	IEEE 802.16 C_SAP	IEEE 802.16 M_SAP
Link_Event_Subscribe	N/A	N/A
Link_Event_Unsubscribe	N/A	N/A
Link_Configure_Thresholds	C-HO-REQ/RSP (HO-Scan)	N/A
Link_Up	C-NEM-RSP (Registration)	N/A
Link_Down	N/A	C-NEM-RSP (Deregistration)
Link_Going_Down	N/A	N/A
Link_Event_Rollback	N/A	N/A
Link_Detected	C-HO-RSP (HO-Scan)	N/A
Link_Parameters_Report	C-HO-IND (HO-Scan) C-HO-RSP (HO-Scan) C-RRM-RSP C-SFM-RSP	N/A
Link_PDU_Transmit_Status	N/A	N/A
Link_Handover_Imminent	C-HO-IND (HO-Start)	N/A
Link_Handover_Complete	C-HO-IND (HO-CMPLT)	N/A
Link_Capability_Discover	N/A	N/A
Link_Get_Parameters	C-SFM-REQ/RSP C-HO-REQ/RSP/IND (HO-Scan) C-RRM-REQ/RSP	N/A

Table K-1—MIH_Link_SAP/802.16 primitives mapping

MIH_LINK_SAP Primitive		IEEE 802.16 C_SAP	IEEE 802.16 M_SAP
Link_Action	LINK_DISCONNECT	C-NEM-REQ/RSP (Deregistration)	N/A
	LINK_LOW_POWER	C-IMM-REQ/RSP (Idle_Mobile_Initiation)	
	LINK_POWER_DOWN	N/A	M-SSM-REQ/RSP (Power down)
	LINK_POWER_UP	N/A	M-SSM-REQ/RSP (Power on)

Table K-2—MIH_LINK_SAP/802.11/802.3ah/802.1ag primitives mapping

Primitives	IEEE 802.11	IEEE 802.3ah	IEEE 802.1ag
Link_Event_Subscribe	N/A	N/A	N/A
Link_Event_Unsubscribe	N/A	N/A	N/A
Link_Configure_Thresholds	MLME-MEASURE.request MLME-MREQUEST.request*	N/A	N/A
Link_Up	MLME-LinkUp.indication	Link Up	Dot1agCfmCCheckRestoredEvent
Link_Down	MLME-LinkDown.indication	Link fault	Dot1agCfgCCheckLossEvent
Link_Going_Down	MLME-LinkGoingDown.indication	Dying Gasp	N/A
Link_Event_Rollback	MLME-EventRollback.indication	N/A	N/A
Link_Detected	N/A	N/A	N/A
Link_Parameters_Report	MLME-MEASURE.confirm MLME-MREPORT.indication†	N/A	N/A
Link_PDU_Transmit_Status	MA-UNIDATA-STATUS.indication	N/A	N/A
Link_Handover_Imminent	MLME-LinkHandoverImminent.indication	N/A	N/A
Link_Handover_Complete	MLME-LinkHandoverComplete.indication	N/A	N/A
Link_Capability_Discover	N/A	N/A	N/A
Link_Get_Parameters	N/A	N/A	N/A
Link_Action	N/A	N/A	N/A

[*] It is used to configure threshold values for Link_Parameters_Report. Thresholds are used for triggering reports. IEEE 802.11k primitives, MLME-MEASURE.request(local) and MLME-MREQUEST.request(remote), can be used for that purpose. Only Beacon Request IE and QoS Metric IE can be used for setting thresholds and triggering reports. MLME-MEASURE primitive does not support confirmation to confirm the threshold setting results. It means that MLME-MEASURE primitive does not have the corresponding primitive to Link_Configure_Threshold.confirm. MLME-MEASURE.confirm is used to deliver the measurement results not to confirm the threshold setting.

[†] IEEE 802.11k MLME-MEASURE.confirm and MLME-MREPORT.indication can be used. If MLME-MEASURE.request or MLME-MREQUEST.request includes Beacon Request IE or QoS Metric IE, then MLME-MEASURE.confirm or MLME-MREPORT.indication is delivered to the MIHF when one of the reporting conditions (thresholds) is satisfied. Link_Parameter_Report.indication can be also generated at a predefined regular interval determined by a user configurable time. This is also performed by MLME-MEASURE.request and MLME-MEASURE.confirm (local) or MLME-MREQUEST.request and MLME-MREPORT.indication (remote) with measurement duration setting.

Table K-3—MIH_LINK_SAP/3GPP/3GPP2 primitives mapping

Primitives	3GPP	3GPP2
Link_Event_Subscribe	N/A	N/A
Link_Event_Unsubscribe	N/A	N/A
Link_Configure_Thresholds	SMREG-PDP-MODIFY	L2.Supervision.Request
Link_Up	SMSM-ACTIVE RABMSM-ACTIVATE	L2.Condition.Notification LCP-Link-Open LCP-Link-Up IPCP-Link-Open
Link_Down	SMSM-DEACTIVEATE SMSM-STATUS RABMSM-DEACTIVATE RABMSM-STATUS RABMAS-RAB-RELEASE	LCP-Carrier-Failure LCP-Link-Quality-Failure LCP-Timeout IPCP-Link-Closed IPCP-Config-Failure IPCP-Timeout
Link_Going_Down	N/A	LCP-Closing
Link_Event_Rollback	N/A	N/A
Link_Detected	N/A	N/A
Link_Parameters_Report	SMSM-MODIFY RABMSM-MODIFY	N/A
Link_PDU_Transmit_Status	N/A	N/A
Link_Handover_Imminent	N/A	N/A
Link_Handover_Complete	RABMAS-RAB-ESTABLISH RABMSM-MODIFY	L2.Data.Confirm
Link_Capability_Discover	N/A	N/A
Link_Get_Parameters	N/A	N/A
Link_Action	N/A	N/A

K.2 C_SAP primitives referenced by IEEE 802.21

Table K-4 shows the description of C_SAP primitives that are relevant to IEEE 802.21.

Table K-4—C_SAP primitives referenced by 802.21

Primitives	Description
C-NEM-REQ (Action Type = Ranging) C-NEM-RSP (Action Type = Ranging)	Upper layers can control ranging procedure with these primitives. Upper layers shall commence IEEE 802.16 link setup procedure by sending C-NEM-REQ (Action Type = Ranging) primitive through NCMS. **Note:** C-NEM-REQ (Action Type = Ranging) primitive with ranging type: “initial” is an IEEE 802.16 Link Switch Link Command (link setup) which corresponds to Link Switch MIH command.
M-NEM-REQ (Action Type = Registration)	Upper layers can control registration procedure with these primitives. Upper layers are notified of link setup by M-NEM-RSP (Action Type = Registration). **Note:** M-NEM-RSP (Action Type = Registration) primitive is an IEEE 802.16 Link_Up Link Event.
C-NEM-NOTIFY (Action Type = Neighbor)	When an IEEE 802.16 MAC receives neighbor advertisement (MOB_NBR-ADV), this primitive is used to deliver the information to the upper layers.
C-HO-REQ (HO-Scan), C-HO-RSP (HO-Scan)	Upper layers can command autonomous scanning with these primitives. Upper layers can also schedule scanning period with a BS. During scanning period BS may buffer downlink traffic to the mobile terminal. **Note:** These primitives are Scan Link Command for MIH_Link_Actions command in IEEE 802.21.
C-HO-REQ (HO-Scan), C-HO-RSP (HO-Scan) C-HO-NOTIRY (HO-Scan)	Delivery of the primitives shall be based on the pre-registration procedure between upper layer management entities and NCMS. Scan report can be made remotely to the BS or locally to the upper layer entity depending on the report target value in C-HO-NOTIFY (HO-Scan)
C-HO-REQ, C-HO-RSP, C-HO-NOTIFY	Upper layers can control handover procedure by using these primitives. **Note:** These primitives can be used as link commands for MIH_Net_HO_Candidate_Query.request/response. Currently only parameters relevant to IEEE 802.16 handover are included. Parameters for Media Independent Handover shall be identified and added.
C-HO-NOTIFY	An MS transmits a MOB_HO-IND message for final indication that is about performing a HO. **Note:** This primitive is a Link Command of the MIH_Net_HO_Commit.request/response MIH Command. For remote service, IEEE 802.16 MAC management message shall be used. (MOB_HO-IND). Currently only parameters relevant to IEEE 802.16 handover are included. Parameters for Media Independent Handover shall be identified and added.
M-MTM-REQ M-MTM-RSP	These primitives are used to manage the status of mobile terminal. Upper layer can change the status of mobile terminal into power on/down/hold/de-register, etc. **Note:** These can be mapped to MIH_Configure MIH Command Service. For remote service, IEEE 802.16 MAC management messages can be used. (RES-CMD/ DREG-CMD)

Annex L Information service examples

(informative)

L.1 Example query for RDF_DATA

An example query request and response when RDF_DATA is specified as InfoQueryType to obtain a list of IEEE 802.11 point of attachments (i.e., basic service set identifiers (BSSIDs)) around a specific location where the location is represented as an IEEE 802.11 point of attachment is shown below.

```
MIH_Get_Information.request (RDF_DATA, "PREFIX mihbasic: <URL_TO_BE_ASSIGNED>
SELECT ?poa_address
WHERE {?x1 mihbasic:neighboring-poa ?x2 .
?x2 mihbasic:link-type 19 .
?x2 mihbasic:poa_address ?x3 .
?x3 mihbasic:address ?poa_address .
?x1 mihbasic:poa_address ?x4 .
?x4 mihbasic:address "001122334455" }")
MIH_Information.response(RDF_DATA, "<?xml version="1.0"?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
 <head>
  <variable name="poa_address"/>
 </head>
 <results>
  <result>
<binding name="poa-address"><literal datatype="http://www.w3.org/2001/XMLSchema#hexBinary">aabbccdde-
eff</literal></binding>
<binding name="poa-address"><literal datatype="http://www.w3.org/2001/XMLSchema#hexBi-
nary">0123456789ab</literal></binding>
  </result>
 </results>
</sparql>", Success)
```

L.2 Example query for RDF_SCHEMA_URL

An example query request and response when XML_SCHEMA_URL is specified as InfoQueryType is shown below.

```
MIH_Get_Information.request(RDF_SCHEMA_URL, "")
MIH_Get_Information.response(RDF_SCHEMA_URL,
"http://www.networkdiscovery.org/2005/04/rdf-extended-schema/",Success)
```

L.3 Example query for RDF_SCHEMA

An example query request and response when RDF_SCHEMA is specified as InfoQueryType to obtain the extended schema is shown below.

```
MIH_Get_Information.request(RDF_SCHEMA, "")
MIH_Get_Information.response(RDF_SCHEMA,
"<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
<!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
```

```
<!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
]>
<rdf:RDF xmlns:rdf="&rdf;" xmlns: :rdfs="&rdfs;">
…
</rdf:RDF>", Success)
```

L.4 SPARQL query example and usage of RDF schema

This document contains an example of SARQL query and usage of RDF basic schema.

L.4.1 RDF example

The format of the RDF is Turtle. Turtle is an extension of Notation 3 (N3). Turtle is a language which is a compact and readable syntax describing only RDF graphs.

```
@prefix : <http://www.mih.org/2006/09/rdf-basic-schema#> .

  <http://informationserver/network1>     :cost  [
        :currency "USD";
        :service-info "1";
        :unit "2";
        :value "5" ];
     :network-type "23";
     :operator-identifier  [
        :operator-name "red-mobile";
        :operator-namespace "0" ];
     :poa  [
        :poa-address  [
           :address "aaaaaaaaaaaa";
           :address-type "6" ];
        :poa-capabilities "004a";
        :poa-channel-range  [
           :high-channel-range "470";
           :low-channel-range "450" ];
        :poa-data-rate "50";
        :poa-location  [
           :geospatial-location-information  [
              :altitude "0f00";
              :altitude-resolution "30";
              :altitude-type "1";
              :datum "1";
              :latitude "04dcc1fc8";
              :latitude-resolution "21";
              :longitude "f65ecf031";
              :longitude-resolution "20" ];
           :method "0" ];
        :poa-mac-type "CSMA-CA";
        :poa-subnet-information  [
           :prefix-length "16";
           :subnet-address  [
              :address "c0230000";
              :address-type "1" ] ] ],
           [
        :poa-address  [
           :address "bbbbbbbbbbbb";
           :address-type "6" ];
```

```
        :poa-capabilities "004c";
        :poa-channel-range  [
            :high-channel-range "430";
            :low-channel-range "410" ];
        :poa-data-rate "50";
        :poa-location  [
            :geospatial-location-information  [
                :altitude "0f00";
                :altitude-resolution "30";
                :altitude-type "1";
                :datum "1";
                :latitude "04dcc1fcf";
                :latitude-resolution "21";
                :longitude "f65ecf030";
                :longitude-resolution "20" ];
            :method "0" ];
        :poa-mac-type "";
        :poa-subnet-information  [
            :prefix-length "16";
            :subnet-address  [
                :address "c01f0000";
                :address-type "1" ] ] ];
    :qos "class1/class2/class3";
    :roaming-partner  [
        :operator-name "red-partner1";
        :operator-namespace "0" ],
            [
        :operator-name "red-partner2";
        :operator-namespace "0" ];
    :security "strong-cipher" .

<http://informationserver/network2>     :cost  [
        :currency "USD";
        :service-info "1";
        :unit "2";
        :value "3" ];
    :network-type "27";
    :operator-identifier  [
        :operator-name "red-mobile";
        :operator-namespace "0" ];
    :poa  [
        :poa-address  [
            :address "cccccccccccc";
            :address-type "6" ];
        :poa-capabilities "00b7";
        :poa-channel-range  [
            :high-channel-range "470";
            :low-channel-range "450" ];
        :poa-data-rate "80";
        :poa-location  [
            :geospatial-location-information  [
                :altitude "6700";
                :altitude-resolution "30";
                :altitude-type "2";
                :datum "1";
                :latitude "053c1f751";
                :latitude-resolution "21";
                :longitude "f50ba5b97";
                :longitude-resolution "20" ];
```

```
            :method "0" ];
        :poa-mac-type "IEEE 802.16";
        :poa-subnet-information  [
            :prefix-length "16";
            :subnet-address  [
                :address "c0c90000";
                :address-type "1" ] ] ];
    :qos "class1";
    :security "strong-cipher" .

<http://informationserver/network3>     :cost  [
        :currency "USD";
        :service-info "1";
        :unit "2";
        :value "5" ];
    :network-type "27";
    :operator-identifier  [
        :operator-name "blue-mobile";
        :operator-namespace "0" ];
    :poa  [
        :poa-address  [
            :address "dddddddddddd";
            :address-type "6" ];
        :poa-capabilities "00b1";
        :poa-channel-range  [
            :high-channel-range "470";
            :low-channel-range "450" ];
        :poa-data-rate "95";
        :poa-location  [
            :geospatial-location-information  [
                :altitude "6800";
                :altitude-resolution "30";
                :altitude-type "2";
                :datum "1";
                :latitude "053c1f751";
                :latitude-resolution "21";
                :longitude "f50ba5b97";
                :longitude-resolution "20" ];
            :method "0" ];
        :poa-mac-type "IEEE 802.16";
        :poa-subnet-information  [
            :prefix-length "48";
            :subnet-address  [
                :address "2002522d94d10000000000000000000000";
                :address-type "2" ] ] ];
    :qos "class1/class2";
    :roaming-partner  [
        :operator-name "blue-partner1";
        :operator-namespace "0" ];
    :security "open" .
```

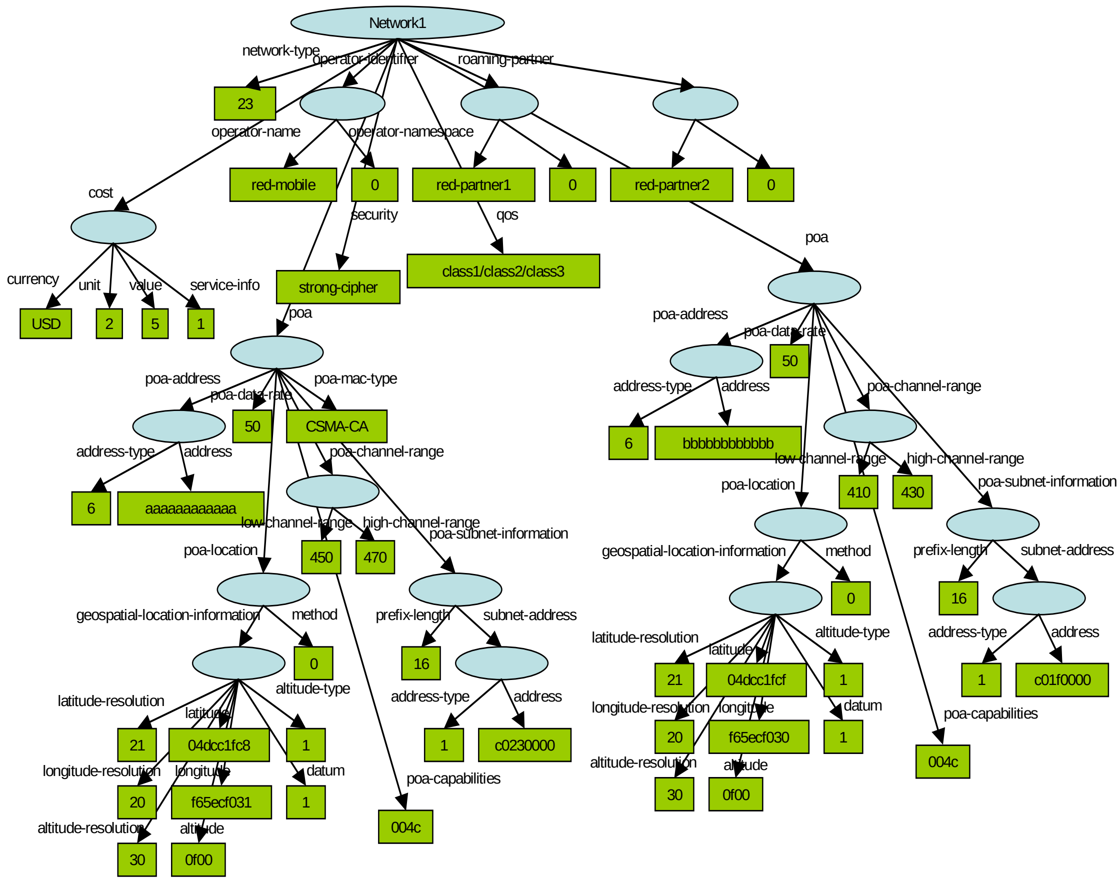


Figure L-1—-Graphic representation of example network (network 1)

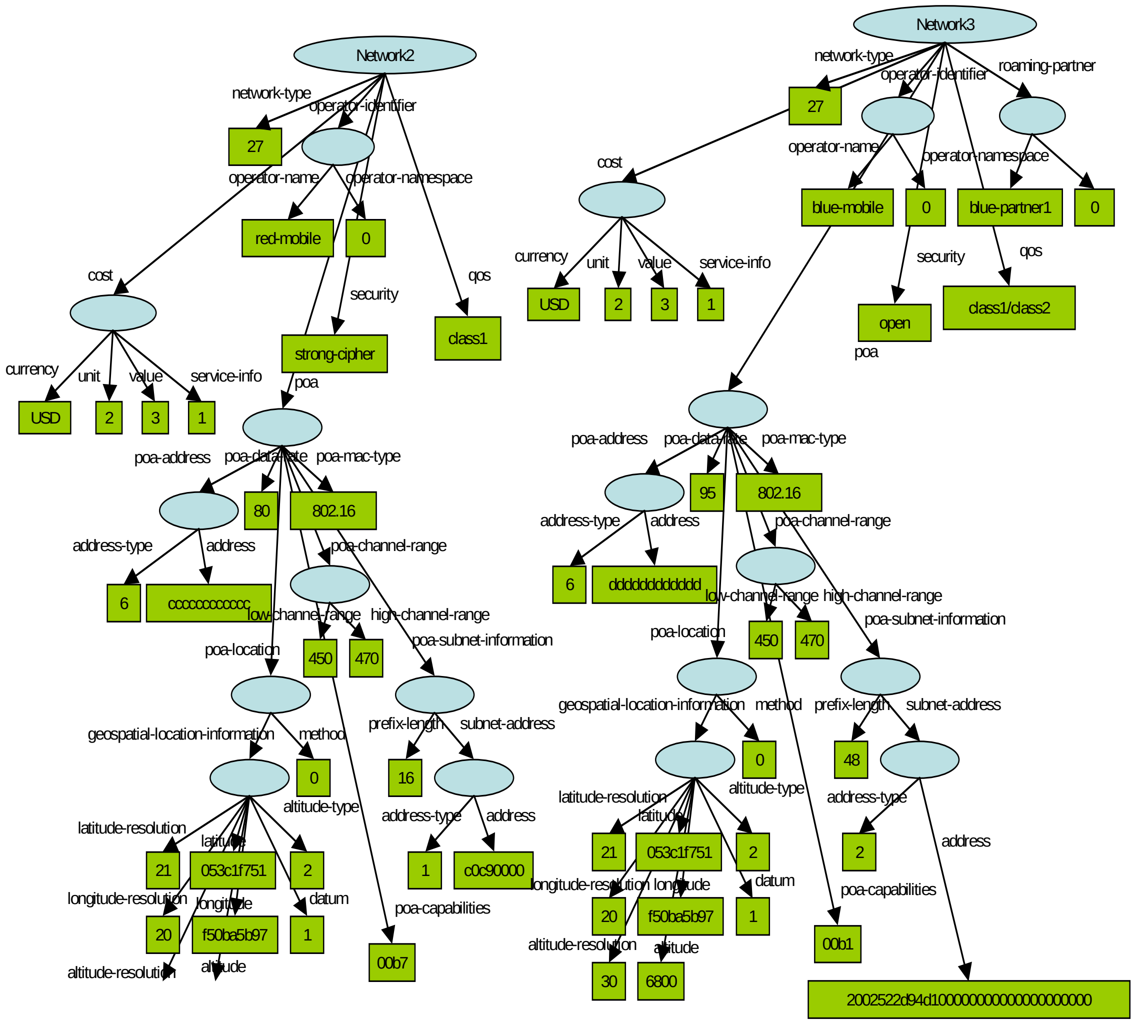


Figure L-2—Graphic representation of example network (network 2 and 3)

L.4.2 SPARQL query example

There are 5 example queries which are same queries as Appendix of 21-06-0674-02-0000-IE_TLV_repsesentation.

L.4.2.1 Example Query 1

MIIS client is asking for all neighboring network information.

The MIIS client sends an MIIS Query Parameters TLV which includes SPARQL query as an Information Request.The actual SPARQL query is below.

DESCRIBE ?x
WHERE {?x ?y ?z}

MIIS server sends back an MIIS Query Response TLV which also includes SPARQL query response as an Information Response. The SPARQL query response is below. The data format is Turtle.

```
@prefix : <http://www.mih.org/2006/09/rdf-basic-schema#> .

  <http://informationserver/network1>     :cost  [
          :currency "USD";
          :service-info "1";
          :unit "2";
          :value "5" ];
      :network-type "23";
      :operator-identifier  [
          :operator-name "red-mobile";
          :operator-namespace "0" ];
      :poa  [
          :poa-address  [
              :address "aaaaaaaaaaaa";
              :address-type "6" ];
          :poa-capabilities "004a";
          :poa-channel-range  [
              :high-channel-range "470";
              :low-channel-range "450" ];
          :poa-data-rate "50";
          :poa-location  [
              :geospatial-location-information  [
                  :altitude "0f00";
                  :altitude-resolution "30";
                  :altitude-type "1";
                  :datum "1";
                  :latitude "04dcc1fc8";
                  :latitude-resolution "21";
                  :longitude "f65ecf031";
                  :longitude-resolution "20" ];
              :method "0" ];
          :poa-mac-type "CSMA-CA";
          :poa-subnet-information  [
              :prefix-length "16";
              :subnet-address  [
                  :address "c0230000";
                  :address-type "1" ] ] ],
              [
          :poa-address  [
              :address "bbbbbbbbbbbb";
              :address-type "6" ];
          :poa-capabilities "004c";
          :poa-channel-range  [
              :high-channel-range "430";
              :low-channel-range "410" ];
          :poa-data-rate "50";
          :poa-location  [
              :geospatial-location-information  [
                  :altitude "0f00";
                  :altitude-resolution "30";
                  :altitude-type "1";
```

```
                :datum "1";
                :latitude "04dcc1fcf";
                :latitude-resolution "21";
                :longitude "f65ecf030";
                :longitude-resolution "20" ];
            :method "0" ];
        :poa-mac-type "";
        :poa-subnet-information  [
            :prefix-length "16";
            :subnet-address  [
                :address "c01f0000";
                :address-type "1" ] ] ];
    :qos "class1/class2/class3";
    :roaming-partner  [
        :operator-name "red-partner1";
        :operator-namespace "0" ],
            [
        :operator-name "red-partner2";
        :operator-namespace "0" ];
    :security "strong-cipher" .

<http://informationserver/network2>     :cost  [
        :currency "USD";
        :service-info "1";
        :unit "2";
        :value "3" ];
    :network-type "27";
    :operator-identifier  [
        :operator-name "red-mobile";
        :operator-namespace "0" ];
    :poa  [
        :poa-address  [
            :address "cccccccccccc";
            :address-type "6" ];
        :poa-capabilities "00b7";
        :poa-channel-range  [
            :high-channel-range "470";
            :low-channel-range "450" ];
        :poa-data-rate "80";
        :poa-location  [
            :geospatial-location-information  [
                :altitude "6700";
                :altitude-resolution "30";
                :altitude-type "2";
                :datum "1";
                :latitude "053c1f751";
                :latitude-resolution "21";
                :longitude "f50ba5b97";
                :longitude-resolution "20" ];
            :method "0" ];
        :poa-mac-type "IEEE 802.16";
        :poa-subnet-information  [
            :prefix-length "16";
            :subnet-address  [
```

```
            :address "c0c90000";
            :address-type "1" ] ] ];
      :qos "class1";
      :security "strong-cipher" .

   <http://informationserver/network3>    :cost  [
         :currency "USD";
         :service-info "1";
         :unit "2";
         :value "5" ];
      :network-type "27";
      :operator-identifier  [
         :operator-name "blue-mobile";
         :operator-namespace "0" ];
      :poa  [
         :poa-address  [
            :address "ddddddddddd";
            :address-type "6" ];
         :poa-capabilities "00b1";
         :poa-channel-range  [
            :high-channel-range "470";
            :low-channel-range "450" ];
         :poa-data-rate "95";
         :poa-location  [
            :geospatial-location-information  [
               :altitude "6800";
               :altitude-resolution "30";
               :altitude-type "2";
               :datum "1";
               :latitude "053c1f751";
               :latitude-resolution "21";
               :longitude "f50ba5b97";
               :longitude-resolution "20" ];
            :method "0" ];
         :poa-mac-type "IEEE 802.16";
         :poa-subnet-information  [
            :prefix-length "48";
            :subnet-address  [
               :address "2002522d94d100000000000000000000";
               :address-type "2" ] ] ];
      :qos "class1/class2";
      :roaming-partner  [
         :operator-name "blue-partner1";
         :operator-namespace "0" ];
      :security "open" .
```

L.4.2.2 Example Query 2

MIIS client is asking for only UMTS neighboring network information.

SPARQL query:

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
DESCRIBE ?x
```

WHERE {?x mihbasic:network-type "23"}

SPARQL query response (Turtle format):

```
@prefix : <http://www.mih.org/2006/09/rdf-basic-schema#> .
  <http://informationserver/network1>     :cost  [
        :currency "USD";
        :service-info "1";
        :unit "2";
        :value "5" ];
      :network-type "23";
      :operator-identifier  [
        :operator-name "red-mobile";
        :operator-namespace "0" ];
      :poa  [
        :poa-address  [
          :address "aaaaaaaaaaaa";
          :address-type "6" ];
        :poa-capabilities "004a";
        :poa-channel-range  [
          :high-channel-range "470";
          :low-channel-range "450" ];
        :poa-data-rate "50";
        :poa-location  [
          :geospatial-location-information  [
            :altitude "0f00";
            :altitude-resolution "30";
            :altitude-type "1";
            :datum "1";
            :latitude "04dcc1fc8";
            :latitude-resolution "21";
            :longitude "f65ecf031";
            :longitude-resolution "20" ];
          :method "0" ];
        :poa-mac-type "CSMA-CA";
        :poa-subnet-information  [
          :prefix-length "16";
          :subnet-address  [
            :address "c0230000";
            :address-type "1" ] ] ],
          [
        :poa-address  [
          :address "bbbbbbbbbbbb";
          :address-type "6" ];
        :poa-capabilities "004c";
        :poa-channel-range  [
          :high-channel-range "430";
          :low-channel-range "410" ];
        :poa-data-rate "50";
        :poa-location  [
          :geospatial-location-information  [
            :altitude "0f00";
            :altitude-resolution "30";
            :altitude-type "1";
```

```
            :datum "1";
            :latitude "04dcc1fcf";
            :latitude-resolution "21";
            :longitude "f65ecf030";
            :longitude-resolution "20" ];
          :method "0" ];
        :poa-mac-type "";
        :poa-subnet-information  [
          :prefix-length "16";
          :subnet-address  [
            :address "c01f0000";
            :address-type "1" ] ] ];
      :qos "class1/class2/class3";
      :roaming-partner  [
        :operator-name "red-partner1";
        :operator-namespace "0" ],
          [
        :operator-name "red-partner2";
        :operator-namespace "0" ];
      :security "strong-cipher" .
```

L.4.2.3 Example Query 3

MIIS client is asking for only UMTS neighboring network information and only wanting the return of “General Network” information.

SPARQL query:

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
SELECT ?opid ?partners ?currency ?unit ?value ?servinfo ?security ?qos
WHERE {?x1 mihbasic:network-type "23" .
    ?x1 mihbasic:operator-identifier ?x2 .
    ?x2 mihbasic:operator-name ?opid .
    ?x1 mihbasic:roaming-partner ?x3 .
    ?x3 mihbasic:operator-name ?partners .
    ?x1 mihbasic:cost ?x4 .
    ?x4 mihbasic:currency ?currency .
    ?x4 mihbasic:unit ?unit .
    ?x4 mihbasic:value ?value .
    ?x4 mihbasic:service-info ?servinfo .
    ?x1 mihbasic:security ?security .
    ?x1 mihbasic:qos ?qos }
```

SPARQL query response (XML format):


```
<?xml version="1.0"?>
<sparql
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xs="http://www.w3.org/2001/XMLSchema#"
  xmlns="http://www.w3.org/2005/sparql-results#" >
 <head>
  <variable name="opid"/>
  <variable name="partners"/>
  <variable name="currency"/>
```

```
    <variable name="unit"/>
    <variable name="value"/>
    <variable name="servinfo"/>
    <variable name="security"/>
    <variable name="qos"/>
  </head>
  <results ordered="false" distinct="false">
    <result>
      <binding name="opid">
        <literal>red-mobile</literal>
      </binding>
      <binding name="partners">
        <literal>red-partner1</literal>
      </binding>
      <binding name="currency">
        <literal>USD</literal>
      </binding>
      <binding name="unit">
        <literal>2</literal>
      </binding>
      <binding name="value">
        <literal>5</literal>
      </binding>
      <binding name="servinfo">
        <literal>1</literal>
      </binding>
      <binding name="security">
        <literal>strong-cipher</literal>
      </binding>
      <binding name="qos">
        <literal>class1/class2/class3</literal>
      </binding>
    </result>
    <result>
      <binding name="opid">
        <literal>red-mobile</literal>
      </binding>
      <binding name="partners">
        <literal>red-partner2</literal>
      </binding>
      <binding name="currency">
        <literal>USD</literal>
      </binding>
      <binding name="unit">
        <literal>2</literal>
      </binding>
      <binding name="value">
        <literal>5</literal>
      </binding>
      <binding name="servinfo">
        <literal>1</literal>
      </binding>
      <binding name="security">
        <literal>strong-cipher</literal>
```

```
    </binding>
    <binding name="qos">
      <literal>class1/class2/class3</literal>
    </binding>
   </result>
  </results>
</sparql>
```

opid	partners	currency	unit	value	servinfo	security	qos
red-mobile	red-partner1	USD	2	5	1	strong-cipher	class1/class2/class3
red-mobile	red-partner2	USD	2	5	1	strong-cipher	class1/class2/class3

L.4.2.4 Example Query 4

MIIS client is asking for PoA information owned by operator “blue-mobile”.

SPARQL query:

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
DESCRIBE ?y
WHERE {?x mihbasic:poa ?y .
      ?x mihbasic:operator-identifier ?z .
      ?z mihbasic:operator-name "blue-mobile"}
```

SPARQL query response (Turtle format):

```
  @prefix : <http://www.mih.org/2006/09/rdf-basic-schema#> .

  [     :poa-address  [
           :address "ddddddddddd";
           :address-type "6" ];
        :poa-capabilities "00b1";
        :poa-channel-range  [
           :high-channel-range "470";
           :low-channel-range "450" ];
        :poa-data-rate "95";
        :poa-location  [
           :geospatial-location-information  [
              :altitude "6800";
              :altitude-resolution "30";
              :altitude-type "2";
              :datum "1";
              :latitude "053c1f751";
              :latitude-resolution "21";
              :longitude "f50ba5b97";
              :longitude-resolution "20" ];
           :method "0" ];
        :poa-mac-type "IEEE 802.16";
        :poa-subnet-information  [
           :prefix-length "48";
           :subnet-address  [
              :address "2002522d94d10000000000000000000";
```

:address-type "2"]]].

L.4.2.5 Example Query 5

MIIS client is asking for only subnet information of PoAs of operator “red-mobile”.

SPARQL query:

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
SELECT ?prefix_length ?type ?address
WHERE {?x1 mihbasic:operator-identifier ?x2 .
    ?x2 mihbasic:operator-name "red-mobile" .
    ?x1 mihbasic:poa ?x3 .
    ?x3 mihbasic:poa-subnet-information ?x4 .
    ?x4 mihbasic:prefix-length ?prefix_length .
    ?x4 mihbasic:subnet-address ?x5 .
    ?x5 mihbasic:address-type ?type .
    ?x5 mihbasic:address ?address }
```

SPARQL query response (XML format):


```
<?xml version="1.0"?>
<sparql
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:xs="http://www.w3.org/2001/XMLSchema#"
   xmlns="http://www.w3.org/2005/sparql-results#" >
  <head>
   <variable name="prefix_length"/>
   <variable name="type"/>
   <variable name="address"/>
  </head>
  <results ordered="false" distinct="false">
   <result>
     <binding name="prefix_length">
       <literal>16</literal>
     </binding>
     <binding name="type">
       <literal>1</literal>
     </binding>
     <binding name="address">
       <literal>c0c90000</literal>
     </binding>
   </result>
   <result>
     <binding name="prefix_length">
       <literal>16</literal>
     </binding>
     <binding name="type">
       <literal>1</literal>
     </binding>
     <binding name="address">
       <literal>c01f0000</literal>
     </binding>
   </result>
   <result>
     <binding name="prefix_length">
       <literal>16</literal>
     </binding>
```

```
      <binding name="type">
        <literal>1</literal>
      </binding>
      <binding name="address">
        <literal>c0230000</literal>
      </binding>
    </result>
  </results>
</sparql>
```

prefix_length	type	address
16	16	16
1	1	1
c0c90000	c01f0000	c0230000

L.4.2.6 Example Query 6

MIIS client is asking for PoAs, excluding those with DATA_RATE larger or equial 75.

SPARQL query:

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
DESCRIBE ?y
WHERE {?x mihbasic:poa ?y .
        ?y mihbasic:poa-data-rate ?z .
        FILTER ( xsd:unsignedInt(?z) >= 75 ) }
```

SPARQL query response (Turtle format):

```
@prefix : <http://www.mih.org/2006/09/rdf-basic-schema#> .
[    :poa-address  [
         :address "cccccccccccc";
         :address-type "6" ];
       :poa-capabilities "00b7";
       :poa-channel-range  [
         :high-channel-range "470";
         :low-channel-range "450" ];
       :poa-data-rate "80";
       :poa-location  [
         :geospatial-location-information  [
           :altitude "6700";
           :altitude-resolution "30";
           :altitude-type "2";
           :datum "1";
           :latitude "053c1f751";
           :latitude-resolution "21";
           :longitude "f50ba5b97";
           :longitude-resolution "20" ];
         :method "0" ];
       :poa-mac-type "IEEE 802.16";
       :poa-subnet-information  [
         :prefix-length "16";
```

```
            :subnet-address  [
               :address "c0c90000";
               :address-type "1" ] ] ].

[      :poa-address  [
            :address "dddddddddddd";
            :address-type "6" ];
         :poa-capabilities "00b1";
         :poa-channel-range  [
            :high-channel-range "470";
            :low-channel-range "450" ];
         :poa-data-rate "95";
         :poa-location  [
            :geospatial-location-information  [
               :altitude "6800";
               :altitude-resolution "30";
               :altitude-type "2";
               :datum "1";
               :latitude "053c1f751";
               :latitude-resolution "21";
               :longitude "f50ba5b97";
               :longitude-resolution "20" ];
            :method "0" ];
         :poa-mac-type "IEEE 802.16";
         :poa-subnet-information  [
            :prefix-length "48";
            :subnet-address  [
               :address "2002522d94d1000000000000000000000";
               :address-type "2" ] ] ].
```

L.4.3 SPARQL query

An example of Mobile Node’s current location based query. A Mobile node knows the current location is (N38.89868 (0x04dcc1fc8) , W77.03723 (0xf65ecf031)) using GPS and so on. An approximately 100m square, of which the Mobile node is in center is shown using the change of 19th bit of latitude and longitude. The 19th bit is a resolution of almost 100m.

SPARQL query for retrieving IEEE 802.11 AP BSSID in 100m square of current location of the Mobile is below:

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
PREFIX xsd:      <http://www.w3.org/2001/XMLSchema#>
SELECT ?address
WHERE {?x1 mihbasic:poa-address ?x2 .
     ?x2 mihbasic:address ?address .
     ?x1 mihbasic:poa-location ?x3 .
     ?x3 mihbasic:geospatial-location-information ?x4 .
     ?x4 mihbasic:latitude ?latitude .
     FILTER ((xsd:unsignedLong(?latitude) < 4dcc9fc8 ) &&
            (xsd:unsignedLong(?latitude) > 4dcb9fc8 ) ) .
     ?x4 mihbasic:longitude ?longitude .
     FILTER ((xsd:unsignedLong(?longitude) < f65ed7031 ) &&
            (xsd:unsignedLong(?longitude) > f65ec7031 ) ) .
     }
```

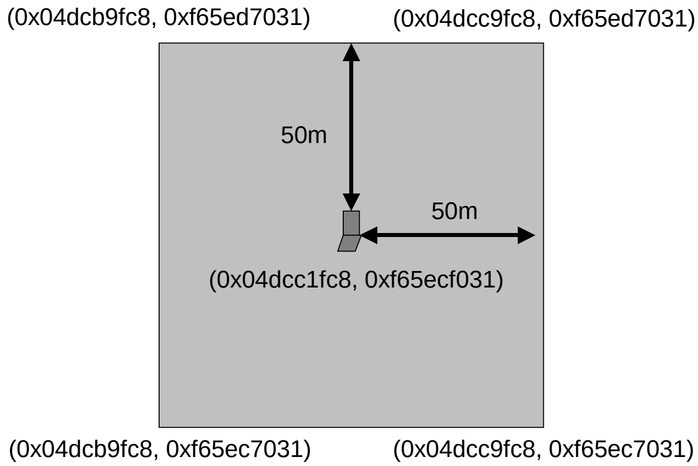


Figure L-3—The serching area of the location based query

L.4.3.1 An example of asking query

If a Mobile node needs yes or no question, SPARQL has ASK form. Example below is for asking whether the current network's one of the roaming partner is "red-partner" or not.

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
PREFIX xsd:      <http://www.w3.org/2001/XMLSchema#>
ASK {?x1 mihbasic:poa ?x2.
?x2 mihbasic:poa-address ?x3 .
       ?x3 mihbasic:address "aabbccddeeff" .
       ?x1 mihbasic:roaming-partner ?x4 .
       ?x4 mihbasic:operator-name "red-partner"
      }
```

L.4.3.2 An example of a query using LIMIT option

If a mobile node can assume that query response might be large data size, LIMIT is useful. If used, limit number of answer is only returned. Example below is that a Mobile node needs only 5 of address of IEEE 802.11 AP around 100m square from Mibile node's current location.

```
PREFIX mihbasic: <http://www.mih.org/2006/09/rdf-basic-schema#>
PREFIX xsd:      <http://www.w3.org/2001/XMLSchema#>
SELECT ?address
WHERE {?x1 mihbasic:poa-address ?x2 .
      ?x2 mihbasic:address ?address .
      ?x1 mihbasic:poa-location ?x3 .
      ?x3 mihbasic:geospatial-location-information ?x4 .
      ?x4 mihbasic:latitude ?latitude .
      FILTER ((xsd:unsignedLong(?latitude) < 4dcc9fc8 ) &&
            (xsd:unsignedLong(?latitude) > 4dcb9fc8 ) ) .
      ?x4 mihbasic:longitude ?longitude .
      FILTER ((xsd:unsignedLong(?longitude) < f65ed7031 ) &&
            (xsd:unsignedLong(?longitude) > f65ec7031 ) ) .
      }
LIMIT 5
```

Annex M Mappings from MIH_LINK_SAP to media-specific SAPs

(normative)

M.1 802.3

LSAP, defined in the IEEE 802.2 standard, provides the interface between the MIHF and the Logical Link Control sublayer in IEEE 802.3 network. This SAP is used for local MIH exchanges between the MIHF and the lower layers of the IEEE 802.3 interface (as the IEEE 802.3 instantiation of the MIH_LINK_SAP) and for the L2 transport of MIH messages across IEEE 802.3 access links.

M.2 802.11

The MIHF uses MLME_SAP for interfacing with the link layer of IEEE 802.11 networks. The MIH_LINK_SAP defines additional primitives that map to MLME_SAP. These primitives are recommended as enhancements to IEEE 802.11 link layer SAPs. MLME_SAP is defined by IEEE 802.11 and it may include, but is not limited to primitives related to:

— System configuration

— Link state change notifications/triggers

— MIH frame transport through control or management frames

LSAP, defined in the IEEE 802.2 standard, provides the interface between the MIHF and the Logical Link Control sublayer in IEEE 802.11. This SAP is used for the L2 transport of MIH messages across IEEE 802.11access links. The MIH messages are carried in 802.11 data frames.

M.3 802.16

The MIHF uses C_SAP and M_SAP for interfacing with the Control and Management planes of the IEEE 802.16 network.

C_SAP is defined by IEEE 802.16g and it may include, but is not limited to primitives related to:

— Handovers (e.g., notification of HO request from mobile station (MS))

— Idle mode mobility management (e.g., Mobile entering idle mode)

— Subscriber and session management (e.g., Mobile requesting session setup)

— Radio resource management

— Authentication, Authorization, and Accounting (AAA) server signaling (e.g., EAP payloads)

— Media Independent Function Services

The primitives defined as part of C_SAP are described in Annex K.2.

M_SAP is defined by IEEE 802.16g and it may include, but is not limited to primitives related to:

— System configuration

— Monitoring Statistics

— Notifications Triggers

— Multi-mode interface management

CS_SAP, defined in the IEEE 802.16 standard, provides the interface between the MIHF and the service-specific Convergence Sublayer in IEEE 802.16 networks. This SAP is used for the L2 transport of MIH messages through data frames across IEEE 802.16 access links.

M.4 3GPP and 3GPP2

This SAP defines MIH_3GLINK_SAP interface between the MIHF and the different protocol elements of the 3G system.

3GPP and 3GPP2 service primitives for GERAN, UMTS, LTE, cdma2000, cdma2000-HRPD and UMB may be used to access MIH services. This is done by establishing a relationship between the 3GPP/3GPP2 primitives and MIH primitives.

Table K-3 lists this mapping. Note that a 3GPP primitive group may be mapped to more than one MIH primitive, as shown in Table K-3.