

2006-10-16

IEEE 802.16i-06/001r4, October 2006

This a NetMan Task Group P802.16i Baseline Document.

~~Draft Amendment to IEEE Standard~~ for Local and Metropolitan Area Networks - Part 16: Management Information Base Extensions

Sponsor

LAN MAN Standards Committee
of the
IEEE Computer Society

and the

IEEE Microwave Theory and Techniques Society

Abstract: This document provides updates to IEEE Std 802.16's MIB for the MAC, PHY and associated management procedures in order to accommodate recent extensions to the standard. The project will use protocol-neutral methodologies for network management to develop resource models and related solution sets for the management of devices in a multi-vendor 802.16 network.

Keywords: fixed broadband wireless access network, mobile broadband wireless access network, metropolitan area network, microwave, millimeter wave, WirelessMAN™ standards, WMAN MIB

~~Copyright © 2006 by the Institute of Electrical and Electronics Engineers, Inc.~~
Three Park Avenue
New York, NY 10016-5997, USA
All rights reserved.

This document is an unapproved draft of a proposed IEEE Standard. As such, this document is subject to change. USE AT YOUR OWN RISK! Because this is an unapproved draft, this document must not be utilized for any conformance/compliance purposes. Permission is hereby granted for IEEE Standards Committee participants to reproduce this document for purposes of IEEE standardization activities only. Prior to submitting this document to another standards development organization for standardization activities, permission must first be obtained from the Manager, Standards Licensing and Contracts, IEEE Standards Activities Department. Other entities seeking permission to reproduce this document, in whole or in part, must obtain permission from the Manager, Standards Licensing and Contracts, IEEE Standard Activities Department.

IEEE Standards Activities Department
Standards Licensing and Contracts
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331, USA

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied “**AS IS.**”

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

(This introduction is not part of IEEE Draft P802.16i, Draft Amendment to IEEE Standard for Local and Metropolitan Area Networks - Part 16: Management Information Base Extensions.)

Participants

This document was developed by the IEEE 802.16 Working Group on Broadband Wireless Access, which develops the WirelessMAN™ Standard for Wireless Metropolitan Area Networks.

IEEE 802.16 Working Group Officers

Roger B. Marks, *Chair*

Jose Puthenkulam, *Vice Chair*

Peiying Zhu, *Secretary*

Primary development was carried out by the Working Group's Network Management Task Group.

Phillip Barber, *Chair*

Joey Chou, *Technical Editor, 802.16f*

The following members of the IEEE 802.16 Working Group on Broadband Wireless Access participated in the Working Group Letter Ballot in which the draft of this standard was prepared and finalized for IEEE Ballot:

[to be determined]

The following participated as non-members in the Working Group Letter Ballot:

[to be determined]

The following members of the IEEE Balloting Committee voted on this standard, whether voting for approval or disapproval, or abstaining.

[to be determined]

The following persons, who were not members of the IEEE Balloting Committee, participated (without voting) in the IEEE Sponsor Ballot in which the draft of this standard was approved:

[to be determined]

When the IEEE-SA Standards Board approved this standard on *[date]*, it had the following membership:

[to be determined]

Also included is the following nonvoting IEEE-SA Standards Board liaison:

[to be determined]

This draft is intended for IEEE-SA Sponsor Ballot with individuals as the ballot group members.

Contents

1. Overview.....	1
1.1 Scope.....	1
1.2 Purpose.....	1
1.3 Reference Models	1
1.3.1 Management Reference Models	2
2. References.....	5
4. Abbreviations and Acronyms	6
9. Configuration	8
9.4 Mobile MIB for SNMP.....	8
9.4.1 MIB-II integration.....	8
9.4.2 Usage of MIB-II tables	8
15. IRP Definitions	13
15.1 NRM IRP IS.....	13
15.1.1 Information Object Classes.....	13
15.1.1.1 Information entities imported and local labels:	13
15.1.1.2 Class diagram.....	14
15.1.1.3 Information object classes definition.....	22
15.1.1.4 Information relationships definition	36
15.1.1.5 Notifications.....	37
15.1.1.6 Information attributes definition.....	37
15.2 NRM IRP SNMP Solution Set.....	49
15.2.1 wmanIf2Mib	49
15.2.1.1 wmanIf2BsObjects.....	50
15.2.1.2 wmanIf2SsObjects	57
15.2.1.3 wmanIf2CommonObjects	60
15.2.2 wmanIf2mMib	61
15.2.3 ASN.1 Definitions of 802.16 MIB for SNMP	62
15.2.3.1 WMAN-IF2-MIB.....	62
15.2.3.2 WMAN-IF2M-MIB	326
Annex E.	328
Annex F. Proposal for Adding Mobility Handover and Paging group MIBs	329
1. Introduction.....	329
2. Proposed Text Introduction	329
2.1 wmanIfBsObjects.....	329
2.1.1 wmanIfBsMobility	329
2.1.1.1 wmanIfBsHandoverConfiguration.....	329
2.1.1.2 wmanIfBsPagingGroupTable	329
3. ASN.1 Definitions of 802.16 MIB for SNMP	330

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

List of Figures

Figure 1—Mobile BWA Network Management Reference Model.....	2
Figure 2—Mobile BWA Network Management Architecture - Context A	3
Figure 3—Mobile BWA Network Management Architecture - Context B.....	4
Figure 4—SS / MS Network Entry	10
Figure 5—WmanSubNetwork Containment/Naming and Association Diagram.....	14
Figure 6—WmanBsFunction Containment/Naming and Association Diagram.....	15
Figure 7—BsMacPacketCs Containment/Naming and Association Diagram	15
Figure 8—BsMacCps Containment/Naming and Association Diagram	16
Figure 9—BsSecurity Containment/Naming and Association Diagram	16
Figure 10—BsPhy Containment/Naming and Association Diagram	17
Figure 11—Top Inheritance Hierarchy Diagram.....	18
Figure 12—WmanManagedFunction Inheritance Hierarchy Diagram	18
Figure 13—Bs Inheritance Hierarchy Diagram	19
Figure 14—Bs PacketCs Inheritance Hierarchy Diagram	19
Figure 15—Bs MacCps Inheritance Hierarchy Diagram.....	20
Figure 16—Bs Phy Inheritance Hierarchy Diagram.....	21
Figure 17—Bs Security Inheritance Hierarchy Diagram.....	21
Figure 18—wmanIf2Mib structure	50
Figure 19—wmanIf2BsPacketCs structure	50
Figure 20—wmanIf2BsCps structure	51
Figure 21—wmanIf2BsPkmObjects structure	53
Figure 22—wmanIf2BsNotification structure	54
Figure 23—wmanIf2BsPhy structure	55
Figure 24—wmanIf2SsCps structure.....	57
Figure 25—wmanIf2SsPkmObjects structure	58
Figure 26—wmanIf2SsNotification structure	58
Figure 27—wmanIf2SsPhy structure.....	59
Figure 28—wmanIf2CmnPacketCs structure	60
Figure 29—wmanIf2CmnCps structure.....	60
Figure 30—wmanIf2CmnPkmObjects structure	61

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

List of Tables

Table 1—Example of the Usage of ifTable objects for BS	9
Table 2—Example of the Usage of ifTable objects for SS.....	9
Table 3—Information entities imported and local labels	13
Table 4—Attributes of WmanBsFunction.....	22
Table 5—Attributes of ExternalWmanBsFunction	23
Table 6—Attributes of WmanBsRelation.....	24
Table 7—Attributes of BsPagingGroup	25
Table 8—Attributes of BsOfdmaPowerCtrl	26
Table 9—Attributes of BsSecurity	27
Table 10—Attributes of PkmBase_F.....	28
Table 11—Attributes of PkmTek_F	29
Table 12—Attributes of BsPkmAuth_F	30
Table 13—Attributes of BsOfdmUIChannel	31
Table 14—Attributes of BsOfdmDIChannel	32
Table 15—Attributes of BsOfdmUcdBurstProfile_F	33
Table 16—Attributes of BsOfdmDcdBurstProfile_	34
Table 17—Attributes of BsClassifierRule_F.....	35
Table 18—Attributes of BsClassifierRule_M	36
Table 19—Roles of the relation ExternalNeighbourWmanBsRelation.....	37
Table 20—Information attributes definition	37

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

Baseline document for Draft Amendment to IEEE Standard for Local and metropolitan area networks

Part 16: Management Information Base Extensions

NOTE—The editing instructions contained in this amendment define how to merge the material contained herein into the existing base standard IEEE Std 802.16-2004.

The editing instructions are shown ***bold italic***. Four editing instructions are used: ***change***, ***delete***, ***insert***, and ***replace***. ***Change*** is used to make small corrections in existing text or tables. The editing instruction specifies the location of the change and describes what is being changed by using strike through (to remove old material) and underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material without disturbing the existing material. Insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. ***Replace*** is used to make large changes in existing text, subclauses, tables, or figures by removing existing material and replacing it with new material. Editorial notes will not be carried over into future editions because the changes will be incorporated into the base standard.

1. Overview

1.1 Scope

This document provides mobility enhancements to the IEEE Std 802.16 Management Information Base for the medium access control layer, physical layer, and associated management procedures. It uses protocol-neutral methodologies for network management to specify resource models and related solution sets for the management of devices in a multivendor 802.16 mobile network.

1.2 Purpose

This amendment provides a definition of managed objects to enhance the standards-based management of 802.16 devices.

1.3 Reference Models

1.3.1 Management Reference Models

Figure 1 illustrates the Management Reference Model (see also 3GPP TS 32.101). It shows the Operation System interfacing with other systems. A number of management interfaces are identified in Figure 1, namely:

1. Between the Network Elements (NEs) and the Element Manager (EM)
2. Between the Element Manager (EM) and the Network Manager (NM)
3. Between the Network Managers and the Enterprise Systems
4. Between Network Managers (NMs)
5. Between Enterprise Systems & Network Managers of different Organisations
6. Between Network Elements (NEs).

The resource model defined within this section focuses primarily on serving management interface "2" and to a lesser extent on management interface "1" from the above list.

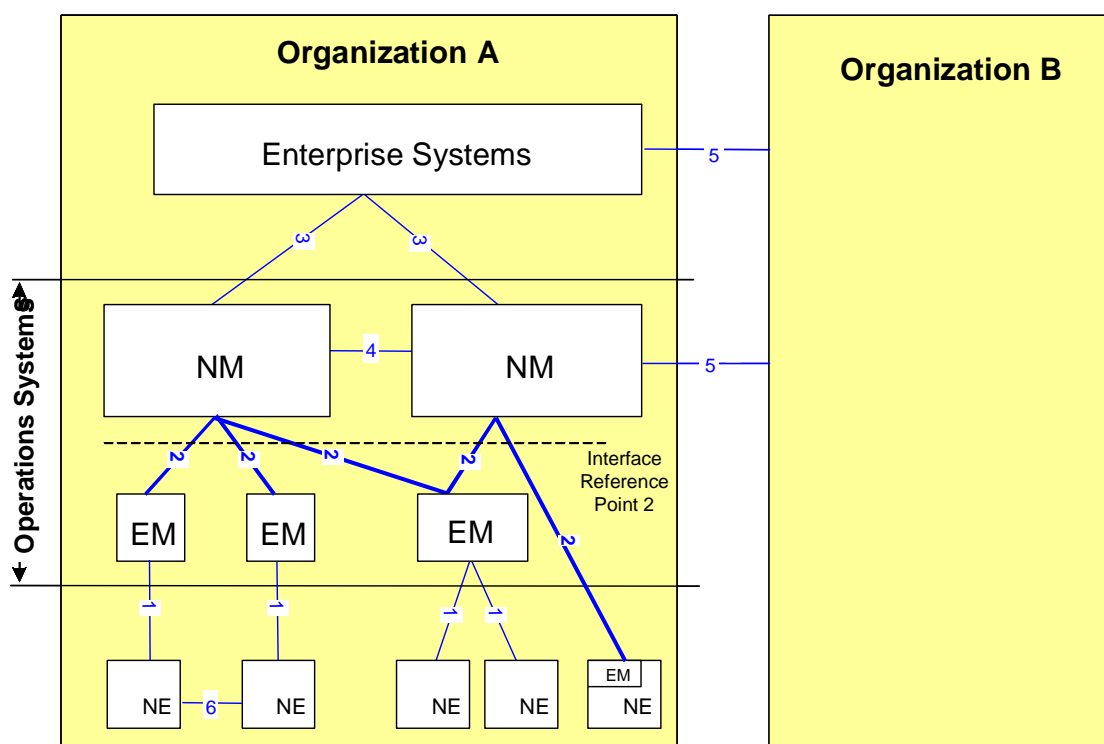


Figure 1—Mobile BWA Network Management Reference Model

Figure 2 and Figure 3 identify system contexts of the Management Interface "2" in terms of its implementation, called IRPAgent, and the user of the IRPAgent, called IRPManager (for a definition of IRPManager and IRPAgent see 3GPP TS 32.102). An NE can be managed either

- via System Context A (IRP Agent is a standalone system) or
- Via System Context B (element management function and IRP Agent embedded within the NE).

The criterion for choosing System Context A or B to manage a particular NE is implementation dependent. An IRP Agent shall support one of the two System Contexts.

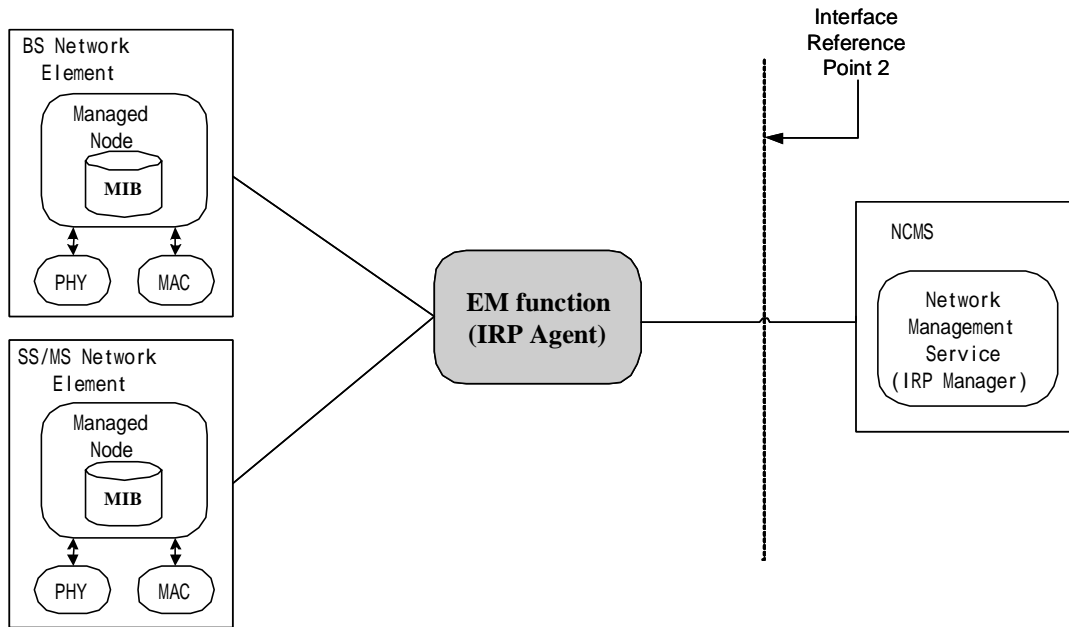


Figure 2—Mobile BWA Network Management Architecture - Context A

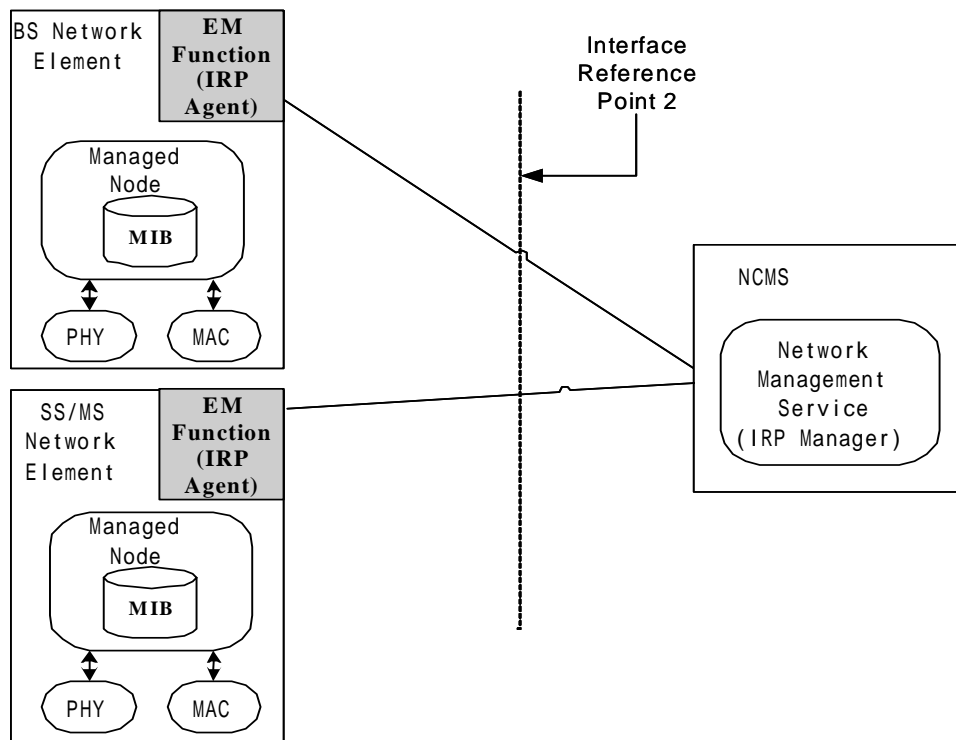


Figure 3—Mobile BWA Network Management Architecture - Context B

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

2. References

This standard shall be used in conjunction with the following publications. When the following specifications are superseded by an approved revision, the revision shall apply.

[Replace the following references]

~~IETF RFC1902, "Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2)", January 1996~~

IETF RFC2578 "Structure of Management Information Version 2 (SMIv2) " April 1999

~~IETF RFC1903, "Textual Convention for Version 2 of the Simple Network Management Protocol (SNMPv2)", January 1996~~

IETF RFC2579 "Textual Conventions for SMIv2 " April 1999

~~IETF RFC2576, "Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework", March 2000~~

IETF RFC3584 "TCoexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework " August 2003

[Insert the following new references]

3GPP TS 32.101, "Principles and High Level Requirements"; Release 6, http://www.3gpp.org/ftp/specs/archive/32_series

3GPP TS 32.150, "Integration Reference Point (IRP) Concept and Definitions", Release 6, http://www.3gpp.org/ftp/specs/archive/32_series

3GPP TS 32.151, "Integration Reference Point (IRP) Information Service (IS) Template", Release 6, http://www.3gpp.org/ftp/specs/archive/32_series

3GPP TS 32.152, "Integration Reference Point (IRP) Information Service (IS) Unified Modelling Language (UML) Repertoire", Release 6, http://www.3gpp.org/ftp/specs/archive/32_series

3GPP TS 32.622, "Configuration Management (CM); Generic Network Resources Integration Reference Point (IRP); Network Resource Model (NRM)"; Release 6, http://www.3gpp.org/ftp/specs/archive/32_series

3GPP2 S.S0028-002-C, "OAM&P for cdma2000 (3GPP2 Generic NRM IRP)" http://www.3gpp2.org/Public_html/specs/index.cfm

3GPP TS 32.102, "Telecommunication management; Architecture", Release 6, <http://www.3gpp.org/ftp/>

1 specs/archive/32_series
2
3
4
5
6
7
8
9
10
11
12
13

14 **4. Abbreviations and Acronyms**

15
16
17 *[Insert a new definition in this sunclause]*
18

19 RDN Relative Distinguished Name
20

21 RP Integration Reference Point
22

23 IS Information Service
24

25 NRM Network Resource Model
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

9. Configuration

[Insert a new subclause 9.4]

9.4 Mobile MIB for SNMP

9.4.1 MIB-II integration

wmanIf2Mib is located under MIB-II subtree. A submission will be sent to the Internet Assigned Numbers Authority (IANA) to assign ieee80216WMAN for wmanIf2Mib.

```

IANAifType ::= TEXTUAL-CONVENTION
SYNTAX INTEGER
{
    ieee80216WMAN (???)  -- IEEE 802.16 WirelessMAN standard to be assigned
                        -- by IANA
}

```

Pending on IETF approval, wmanIf2Mib will be accessed through

```
iso.org.dod.internet.mgmt.mib-2.transmission.ifType (1.3.6.1.2.1.10.???)
```

9.4.2 Usage of MIB-II tables

"Interfaces" group of MIB-II, in RFC2863, has been designed to manage various sub-layers (e.g. MAC and PHY) beneath the internetwork-layer for numerous media-specific interfaces. The implementation of ifTable in SNMP managed BS and SS is mandatory.

The implementation of the ifTable for BS must create one row for each BS sector. Each BS sector may support different standards (e.g. IEEE 802.16-2004, IEEE 802.16e). The following recommendations must be applied to each row defining BS sector:

- ifIndex value is implementation specific
- ifType must be set to ieee80216WMAN
- ifSpeed must be null
- ifPhysAddress must be set to the MAC Address of the BS sector
- All other columnar objects must be initialized as specified in RFC2863

Table 1—Example of the Usage of ifTable objects for BS

ifTable	ifIndex	ifType (IANA)	ifSpeed	ifPhysAddress	ifAdminStatus	ifOperStatus
BS Sector 1	1	ieee80216WMAN	Null	MAC address of BS sector	Administration Status	Operational Status
BS Sector 2	2	ieee80216WMAN	Null	MAC address of BS sector	Administration Status	Operational Status
BS Sector 3	3	ieee80216WMAN	Null	MAC address of BS sector	Administration Status	Operational Status
BS Sector 4	4	ieee80216WMAN	Null	MAC address of BS sector	Administration Status	Operational Status
Ethernet			Null	MAC address	Administration Status	Operational Status

Table 1 shows an example of the usage of ifTable for BS that supports multiple sectors. Each sector may support one of the following MAC / PHY interfaces:

- IEEE 802.16-2004, OFDM 256
- IEEE 802.16-2004, OFDMA 2048
- IEEE 802.16e, OFDM 128
- IEEE 802.16e, OFDM 512
- IEEE 802.16e, OFDM 1024

The implementation of the ifTable for SS must create one row for each SS WirelessMAN interface. Additional rows may be necessary to support other network interfaces, such as Ethernet. The following recommendations must be applied to each row:

- ifIndex value is implementation specific
- ifType must be set to ieee80216WMAN
- ifSpeed must be null
- ifPhysAddress must be set to the SS MAC Address (of the WirelessMAN interface)
- All other columnar objects must be initialized as specified in RFC2863

Table 2—Example of the Usage of ifTable objects for SS

fTable	ifIndex	ifType (IANA)	ifSpeed	ifPhysAddress	ifAdminStatus	ifOperStatus
SS	An ifEntry for SS	ieee80216WMAN	Null	MAC address of SS	Administration Status	Operational Status
Ethernet			Null	MAC address	Administration Status	Operational Status

Table 2 shows an example of the usage of ifTable for SS that may support one of the following MAC / PHY interfaces:

- IEEE 802.16-2004, OFDM 256
- IEEE 802.16-2004, OFDMA 2048

- IEEE 802.16e, OFDMA 128
- IEEE 802.16e, OFDMA 512
- IEEE 802.16e, OFDMA 102

Figure 5 shows a procedure describing how BS can determine the MAC / PHY standard interface and capability a SS / MS can support.

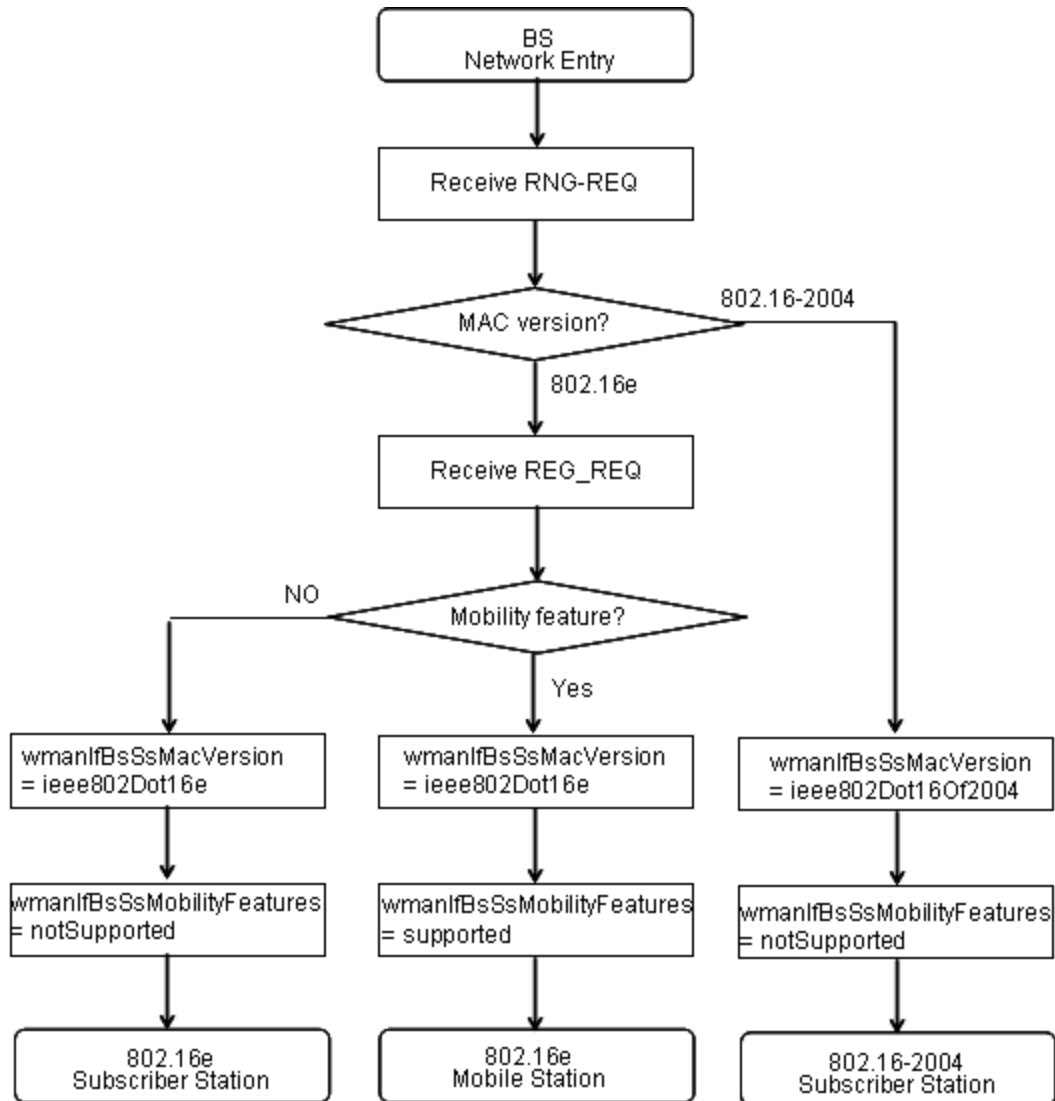


Figure 4—SS / MS Network Entry

1. Receive RNG-REQ from SS / MS
2. If MAC version is 802.16-2004, then
 - a) wmanIfBsSsMacVersion = ieee802Dot16Of2004
 - b) wmanIfBsSsMobilityFeatures = No Supported
 - c) Go to step 5
3. Receive REG-REQ from SS / MS

- 1 4. If Mobility Feature is supported, then
- 2 a) wmanIfBsSsMacVersion = ieee802Dot16e
- 3 b) wmanIfBsSsMobilityFeatures = Supported
- 4 otherwise
- 5 a) wmanIfBsSsMacVersion = ieee802Dot16e
- 6 b) wmanIfBsSsMobilityFeatures = Not Supported
- 7
- 8
- 9
- 10 5. Continue network entry procedure
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45
- 46
- 47
- 48
- 49
- 50
- 51
- 52
- 53
- 54
- 55
- 56
- 57
- 58
- 59
- 60
- 61
- 62
- 63
- 64
- 65

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

[Insert a new subclause 15]

15. IRP Definitions

The IRP concept is borrowed from 3GPP. TS 32.150 states:

"For the purpose of management interface development 3GPP has developed an interface concept known as Integration Reference Point (IRP) to promote the wider adoption of standardized management interfaces in telecommunication networks. The IRP concept and associated methodology employs protocol and technology neutral modelling methods as well as protocol specific solution sets to achieve its goals."

According to TS 32.150:

"The three cornerstones of the IRP concept are:

- Top-down, process-driven modelling approach: The purpose of each IRP is automation of one specific task [...]. This allows taking a "one step at a time" approach with a focus on the most important tasks.
- Technology-independent modelling: To create from the requirements an interface technology independent model. This is specified in the IRP Information Service.
- Standards-based technology-dependent modelling: To create one or more interface technology dependent models from the technology independent model. This is specified in the IRP Solution Set(s)."

This document encompasses phases 2 and 3 only.

15.1 NRM IRP IS

This subclause defines the NRM IRP IS for 802.16 Mobile & Fixed Network, and is based on the IS Template defined in 3GPP TS 32.151 as well as the UML Repertoire defined in 3GPP TS 32.152 - refer to these specifications for details on how to interpret the information defined below.

15.1.1 Information Object Classes

15.1.1.1 Information entities imported and local labels:

Table 3—Information entities imported and local labels

Label reference	Local label
3GPP TS 32.622, information object class, ManagedElement	ManagedElement
3GPP TS 32.622, information object class, ManagedFunction	ManagedFunction

Table 3—Information entities imported and local labels

Label reference	Local label
3GPP TS 32.622, information object class, Sub-Network	SubNetwork
3GPP TS 32.622, information object class, Top	Top
3GPP2 S.S0028-002-C, information object class, ExternalIOCI	ExternalIOCI

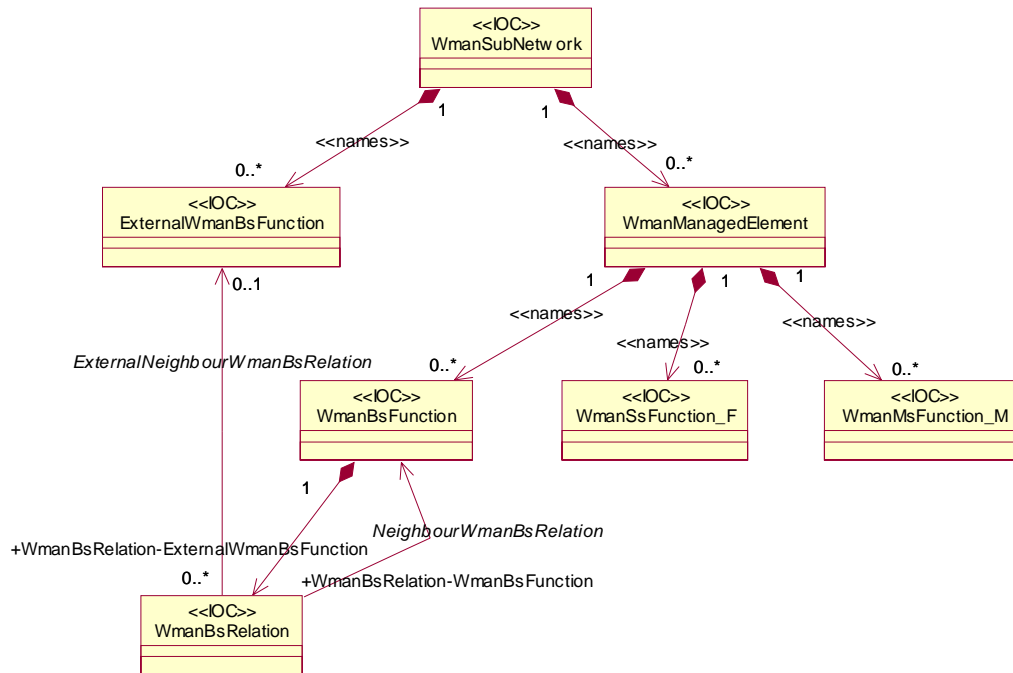
15.1.1.2 Class diagram

15.1.1.2.1 Attributes and relationships

This clause provides the overview of all information object classes in UML. Subsequent clauses provide more detailed specification of various aspects of these information object classes.

The naming and containment for the protocol neutral network management models of the 802.16 standard are shown in the following figures. They are split in several figures only for a readability purpose.

15.1.1.2.1.1 WmanSubNetwork Relationships

**Figure 5—WmanSubNetwork Containment/Naming and Association Diagram**

15.1.1.2.1.2 Bs Object Relationships

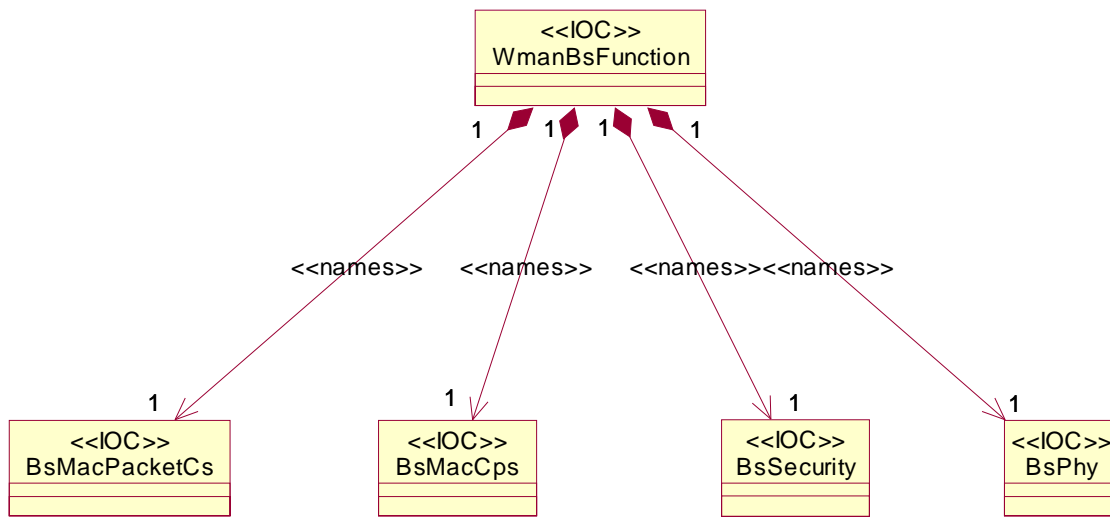


Figure 6—WmanBsFunction Containment/Naming and Association Diagram

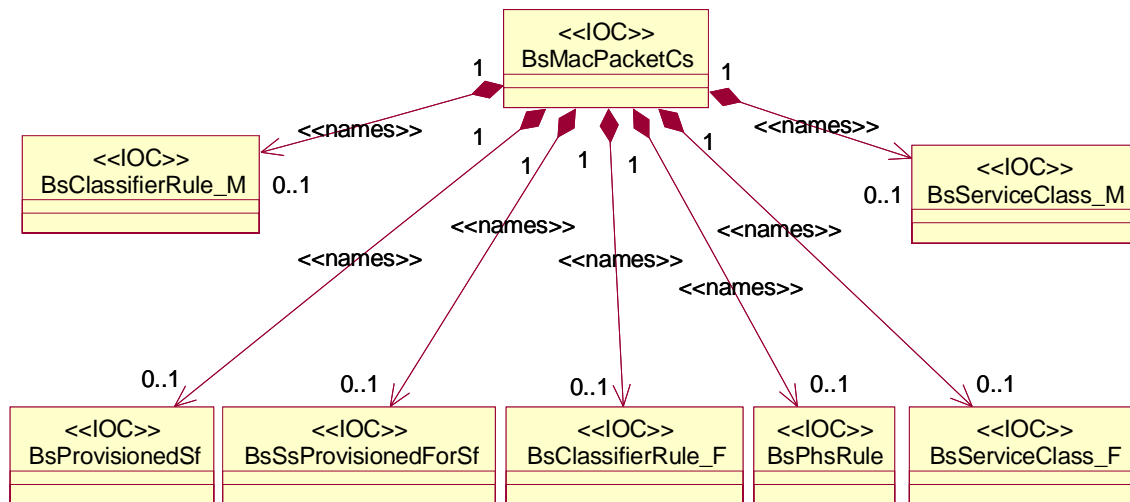


Figure 7—BsMacPacketCs Containment/Naming and Association Diagram

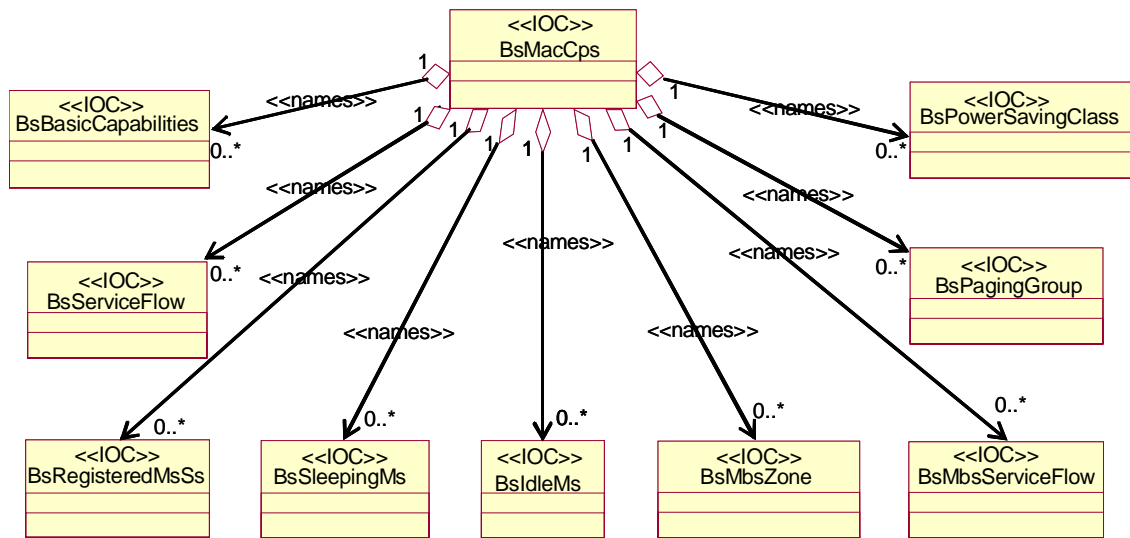


Figure 8—BsMacCps Containment/Naming and Association Diagram

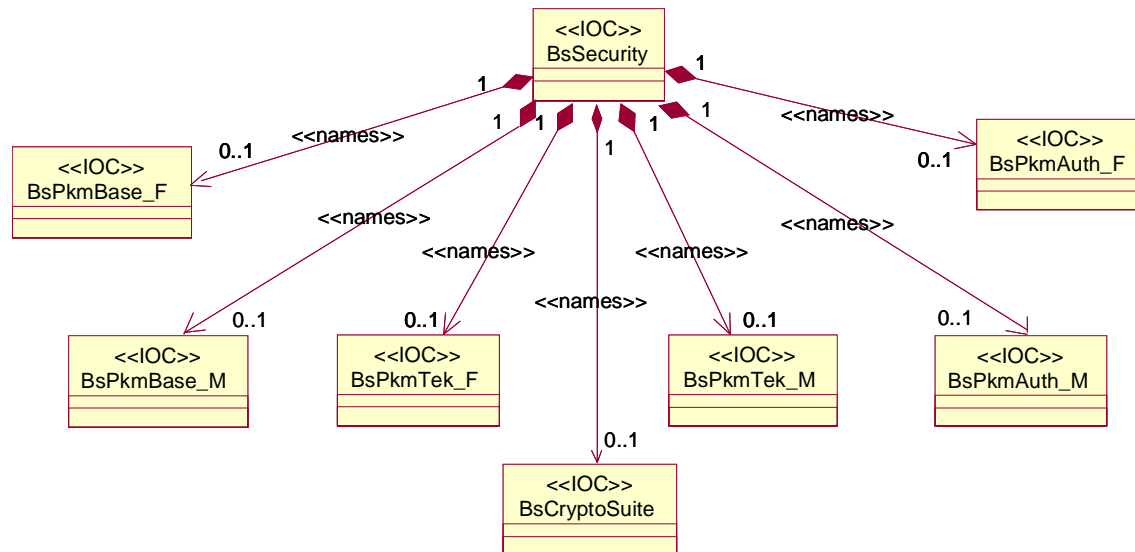


Figure 9—BsSecurity Containment/Naming and Association Diagram

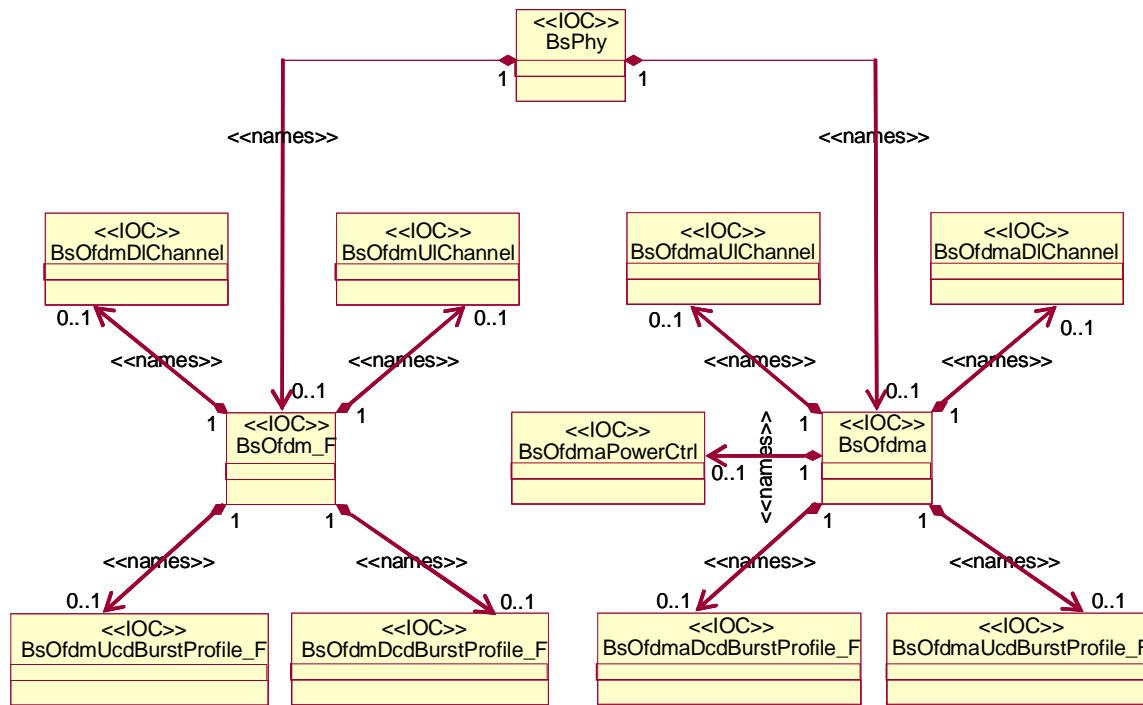


Figure 10—BsPhy Containment/Naming and Association Diagram

15.1.1.2.1.3 Ss Object Relationships

TBD

15.1.1.2.2 Inheritance

This subclause depicts the inheritance relationships that exist between information object classes.

15.1.1.2.2.1 TOP Inheritance

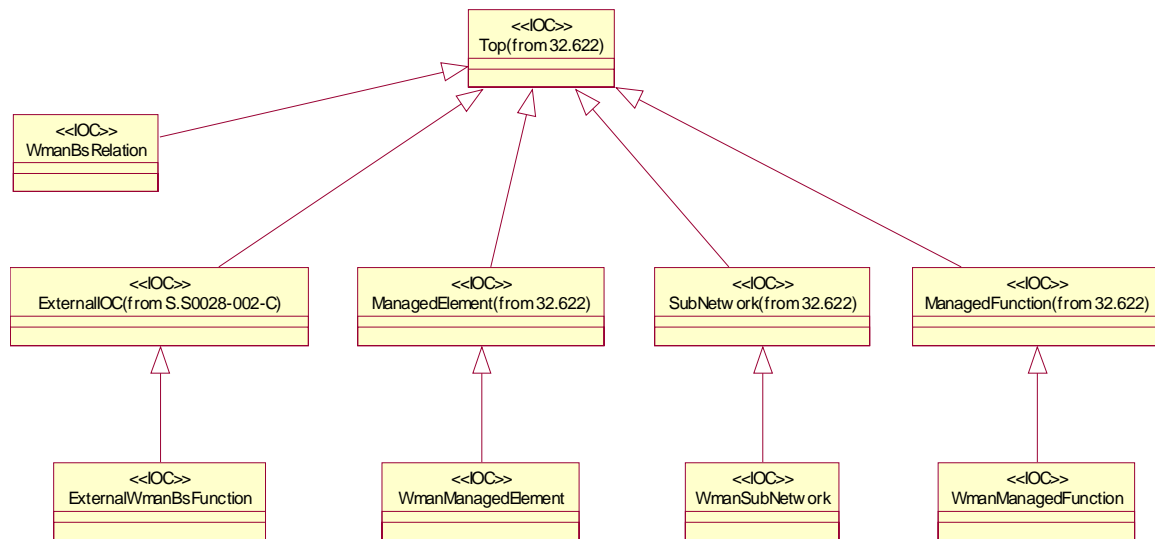


Figure 11—Top Inheritance Hierarchy Diagram

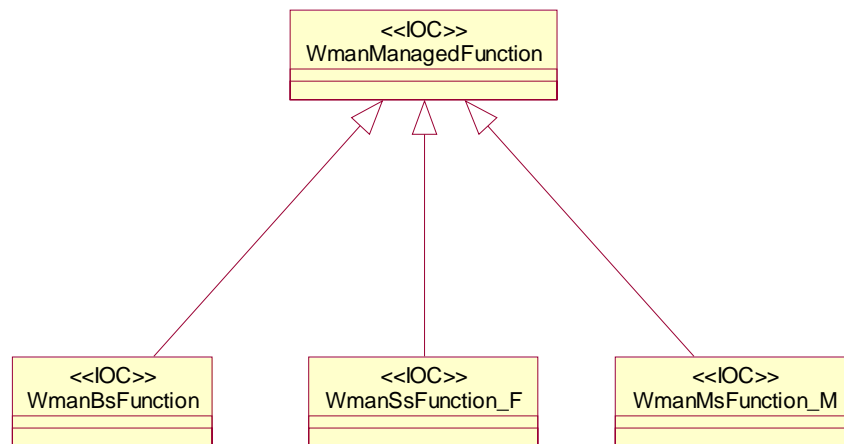


Figure 12—WmanManagedFunction Inheritance Hierarchy Diagram

15.1.1.2.2.2 Bs Object Inheritance

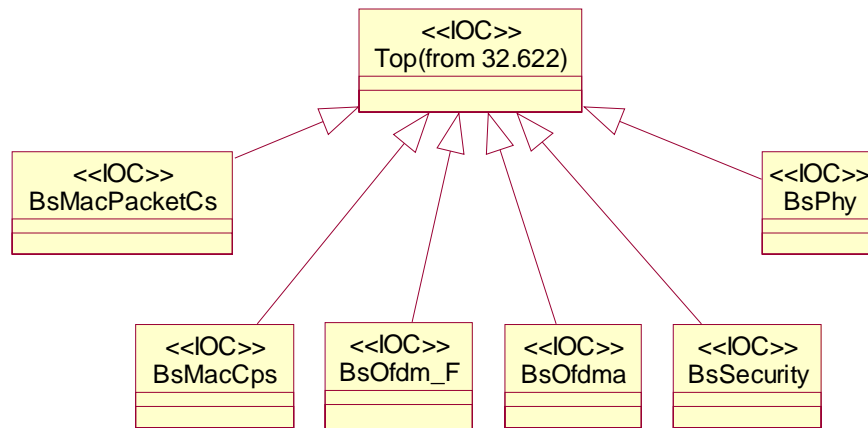


Figure 13—Bs Inheritance Hierarchy Diagram

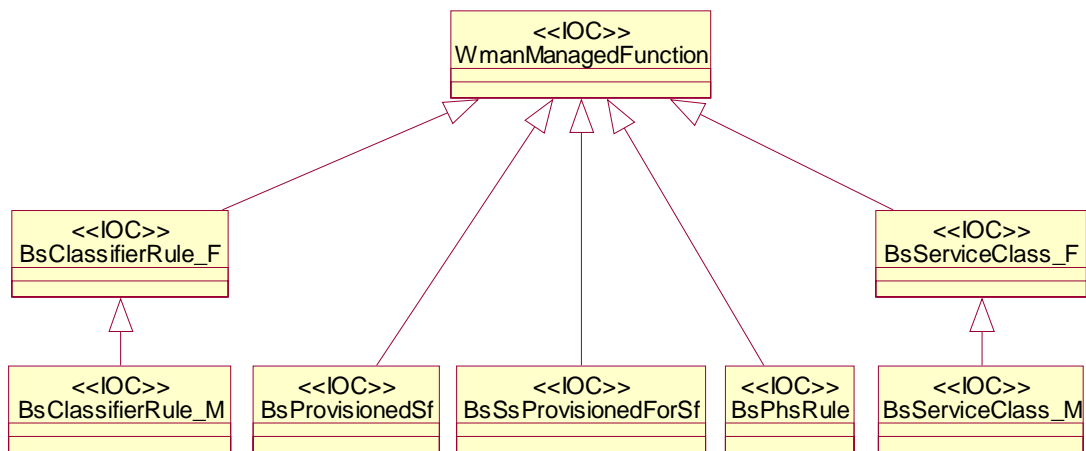


Figure 14—Bs PacketCs Inheritance Hierarchy Diagram

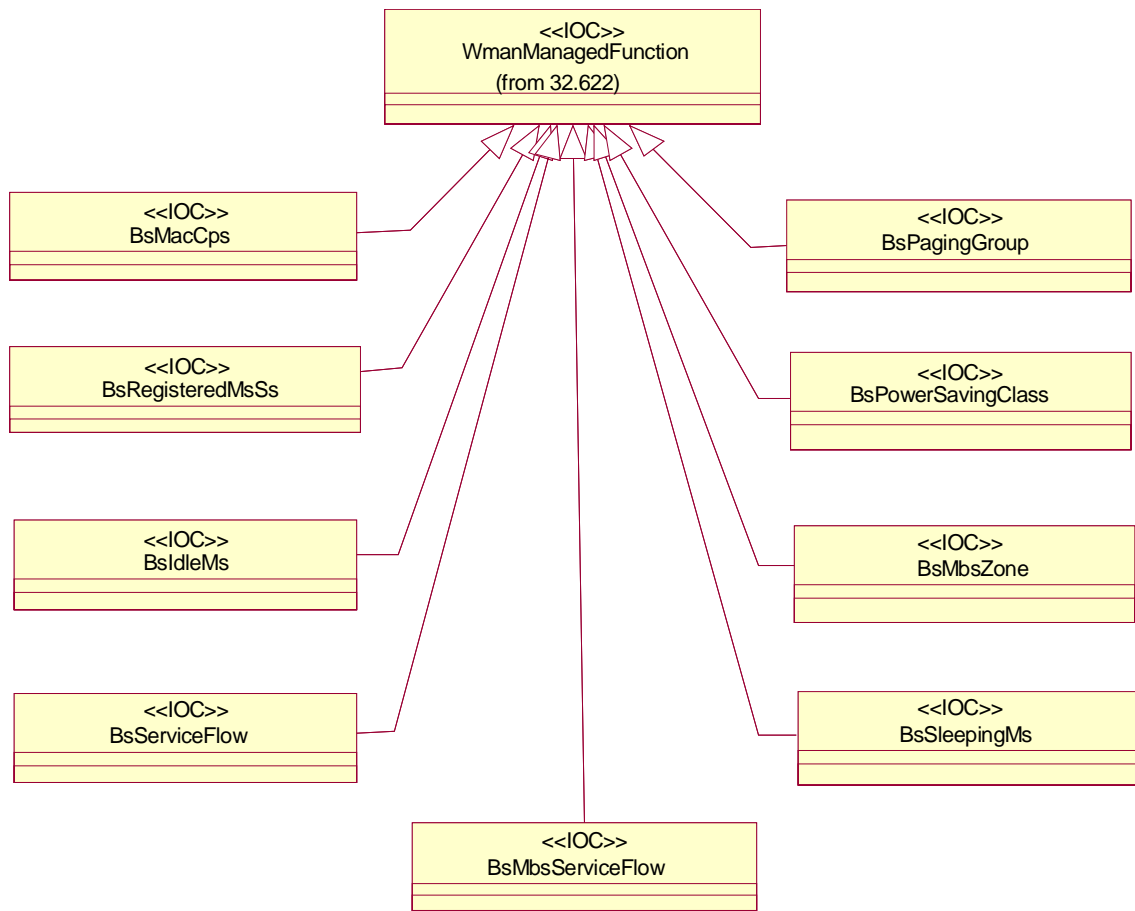


Figure 15—Bs MacCps Inheritance Hierarchy Diagram

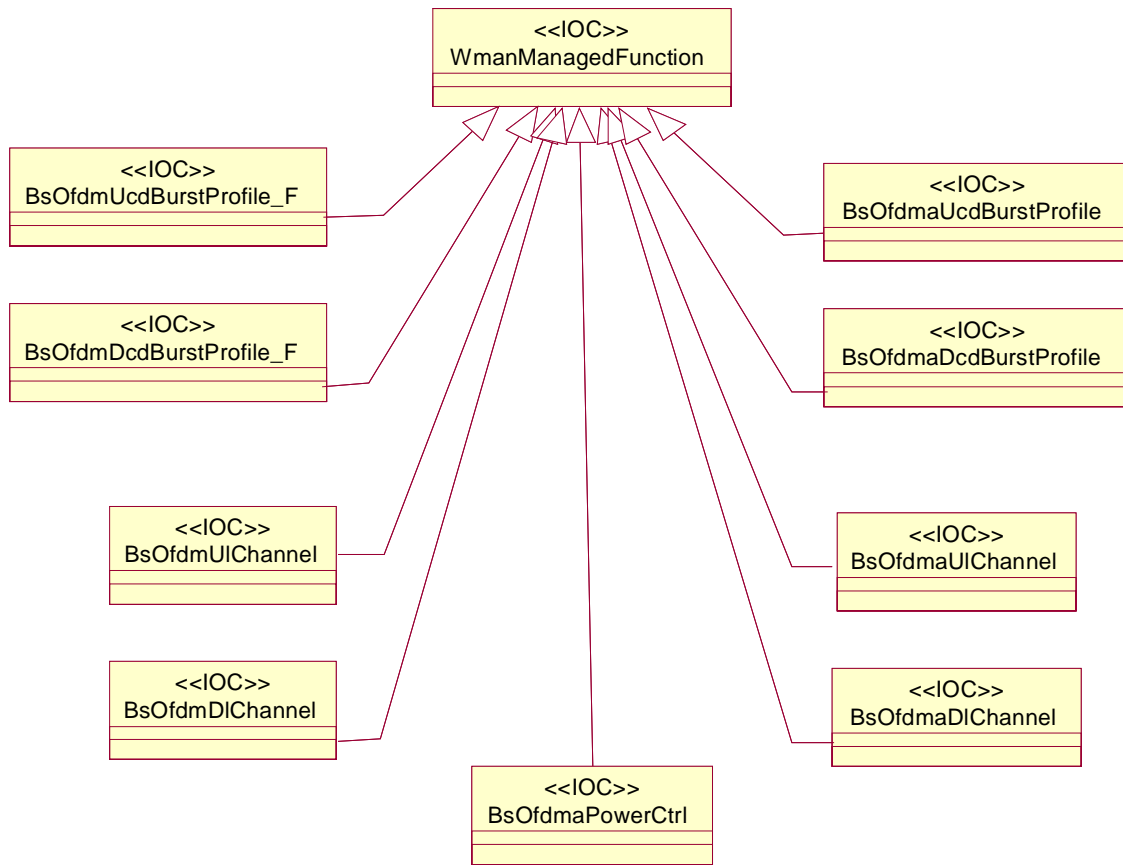


Figure 16—Bs Phy Inheritance Hierarchy Diagram

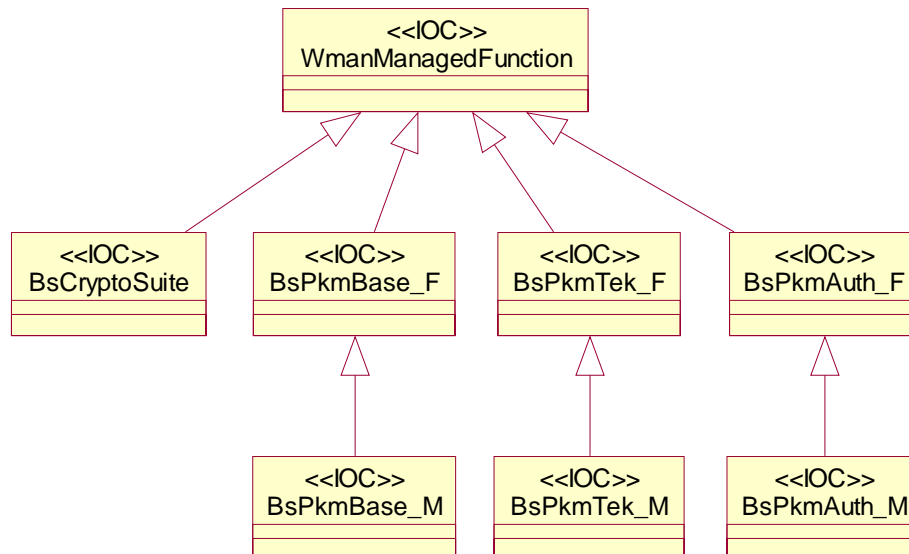


Figure 17—Bs Security Inheritance Hierarchy Diagram

15.1.1.2.2.3 Ss Object Inheritance

TBD

15.1.1.3 Information object classes definition

15.1.1.3.1 IOC WmanBsFunction

15.1.1.3.1.1 Definition

This IOC represents a WMAN Base Station. It is derived from WmanManagedFunction

15.1.1.3.1.2 Attributes

Table 4—Attributes of WmanBsFunction

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
OperatorID	+	M	M	M
BSID	+	M	M	M
HandoverSupportedType	+	M	M	M
SystemResourceRetainTime	+	M	M	M
HOOptimizationMSTimer	+	M	M	M
MSHOREtransmissionTimer	+	M	M	M
MobilitySupportedIndication	+	M	M	M
MSHOCConnectionProcessTime	+	M	M	M
MSHOTeKProcessTime	+	M	M	M
ULPermutationBase	+	M	M	M
DLPermutationBase	+	M	M	M
PreambleIndex	+	M	M	M
SegmentNumber	+	M	M	M

15.1.1.3.2 IOC ExternalWmanBsFunction

15.1.1.3.2.1 Definition

This IOC represents a WMAN base station which belongs to the other subnetwork. It is derived from WmanManagedFunction

15.1.1.3.2.2 Attributes

Table 5—Attributes of ExternalWmanBsFunction

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
ExternalBSId	+	M	M	-
FAIndex	+	M	M	M
BSEIRP	+	M	M	M
SchedulingServiceSupported	+	M	M	M
HOPProcessOptimization	+	M	M	M
Bandwidth	+	M	M	M
FFTSize	+	M	M	M
CyclePrefix	+	M	M	M
FramDurationCode	+	M	M	M
ULPermutationBase	+	M	M	M
DLPermutationBase	+	M	M	M
SegmentNumber	+	M	M	M
PreambleIndex	+	M	M	M

15.1.1.3.3 IOC WmanBsRelation

15.1.1.3.3.1 Definition

This IOC represents the relation between two neighbor WMAN base stations. It is derived from WmanManagedFunction.

15.1.1.3.3.2 Attributes

Table 6—Attributes of WmanBsRelation

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
BSRelationId	+	M	M	-
adjacentBS	+	M	M	M
FAIndex	+	M	M	M
BSEIRP	+	M	M	M
SchedulingServiceSupported	+	M	M	M
HOPprocessOptimization	+	M	M	M
Bandwidth	+	M	M	M
FFTSize	+	M	M	M
CyclePrefix	+	M	M	M
FramDurationCode	+	M	M	M
ULPermutationBase	+	M	M	M
DLPermutationBase	+	M	M	M
SegmentNumber	+	M	M	M
PreambleIndex	+	M	M	M

15.1.1.3.4 IOC BsPagingGroup

15.1.1.3.4.1 Definition

This IOC represents the BS related paging group information. It is derived from WmanManagedFunction.

15.1.1.3.4.2 Attributes

Table 7—Attributes of BsPagingGroup

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
PagingControlId	+	M	M	M
PagingGroupId	+	M	M	M
MgmtResourceHoldingTimer	+	M	M	M
T46Timer	+	M	M	M
PagingRetryCount	+	M	M	M
REQDuration	+	M	M	M
MACHashSkipThreshold	+	M	M	M
BsCDMATransmissionOpportunityAssignment	+	M	M	M
PagingResponseWindow	+	M	M	M
IdleModeTimer	+	M	M	M
IdleModeSystemTimer	+	M	M	M
PagingIntervalLength	+	M	M	M
PagingCycle	+	M	M	M

15.1.1.3.5 IOC BsOfdmaPowerCtrl

15.1.1.3.5.1 Definition

This Information Object Class represents the power control entity of 802.16 BS. For more information about the BS, see subclause 8.4.10.3 of 802.16-2004 and 802.16e-2005.

PowerCtrl is an object which is derived from the WmanManagedFunction.

15.1.1.3.5.2 Attributes

Table 8—Attributes of BsOfdmaPowerCtrl

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
powerCtrlId	--	+	M	M	--
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
msUpPwrAdjStep	--	+	M	M	O
msDownPwrAdjStep	--	+	M	M	O
minPwrAdjLever	--	+	M	M	O
maxPwrAdjLever	--	+	M	M	O
txPwrRepThreshold	--	+	M	M	O
txPwrRepInterval	--	+	M	M	O
alphaPAvg	--	+	M	M	O
txPwrRepThresholdCQI	--	+	M	M	O
txPwrRepIntervalCQI	--	+	M	M	O
alphaPAvgCQI	--	+	M	M	O

15.1.1.3.6 IOC BsSecurity

15.1.1.3.6.1 Definition

This IOC represents a SecurityManagmentFunction object. It is derived from WmanManagedFunction

15.1.1.3.6.2 Attributes

Table 9—Attributes of BsSecurity

Attribute name					
Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier	
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
securityManagementId	-	+	M	M	-

15.1.1.3.7 IOC PkmBase_F

15.1.1.3.7.1 Definition

This IOC represents a PkmBase object. It is derived from WmanManagedFunction.

15.1.1.3.7.2 Attributes

Table 10—Attributes of PkmBase_F

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
wmanIfBsPkmBaseId	-	+	M	M	-
wmanIfBsPkmDefaultAuthLifetime	-	+	M	M	M
wmanIfBsPkmDefaultTekLifetime	-	+	M	M	M
wmanIfBsPkmDefaultSelfSig-ManufCertTrust	-	+	M	M	M
wmanIfBsPkmCheckCertValidity-Periods	-	+	M	M	M
wmanIfBsPMKDefaultPreHandshakeLifetime	-	+	M	M	M
wmanIfBsPMKDefaultLifetime	-	+	M	M	M
wmanIfBsDefaultSACChallengeTimer	-	+	M	M	M
wmanIfBsDefaultSaChallenge-MaxResends	-	+	M	M	M
wmanIfBsDefaultSATEKTimer	-	+	M	M	M
wmanIfBsDefaultSATEKRequest-MaxResends	-	+	M	M	M

15.1.1.3.8 IOC PkmTek_F

15.1.1.3.8.1 Definition

This IOC represents a PkmTek object. It is derived from WmanManagedFunction.

15.1.1.3.8.2 Attributes

Table 11—Attributes of PkmTek_F

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
wmanIfBsPkmTekId	-	+	M	M	-
wmanIfBsPkmTekSAId	-	+	M	-	-
wmanIfBsPkmTekSAType	-	+	M	M	-
wmanIfBsPkmTekDataEncryptAlg	-	+	M	M	-
wmanIfBsPkmTekDataAuthentAlg	-	+	M	M	-
wmanIfBsPkmTekEncryptAlg	-	+	M	M	-
wmanIfBsPkmTekLifetime	-	+	M	M	-
wmanIfBsPkmTekKeySequenceNumber	-	+	M	M	-
wmanIfBsPkmTekExpiresOld	-	+	M	M	-
wmanIfBsPkmTekExpiresNew	-	+	M	M	-
wmanIfBsPkmTekReset	-	+	M	M	M
wmanIfBsPkmAssociatedGKEKSequenceNumber	-	+	M	M	-
wmanIfBsPkmSAServiceType	-	+	M	M	-

15.1.1.3.9 IOC BsPkmAuth_F

15.1.1.3.9.1 Definition

This IOC represents a MS/SSPkmAuth object. It is derived from WmanManagedFunction.

15.1.1.3.9.2 Attributes

Table 12—Attributes of BsPkmAuth_F

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
wmanIfBsMsPkmAuthID	-	+	M	M	-
wmanIfBsSsPkmAuthMacAddress	-	-	M	-	-
wmanIfBsSsPkmAuthKeySequenceNumber	-	+	M	M	-
wmanIfBsSsPkmAuthExpiresOld	-	+	M	M	-
wmanIfBsSsPkmAuthExpiresNew	-	+	M	M	-
wmanIfBsSsPkmAuthLifetime	-	+	M	M	-
wmanIfBsSsPkmAuthReset	-	+	M	M	M
wmanIfBsSsPkmAuthPrimarySAId	-	+	M	M	-
wmanIfBsSsPkmAuthValidStatus	-	+	M	M	-
wmanIfBsMsCMACPacketNumbercounter	-	+	M	M	
wmanIfBsMsCMAC_PN_UL	-	+	M	M	
wmanIfBsMsCMAC_PN_DL	-	+	M	M	
wmanIfBsMsCMACValue	-	+	M	M	
wmanIfBsMsPkmAuthResultCode	-	+	M	M	
wmanIfBsMsPkmAKId	-	+	M	M	
wmanIfBsKeyPushMode	-	+	O	M	
wmanIfBsKeyPushCounter	-	+	O	M	

15.1.1.3.10 IOC BsOfdmUlChannel

15.1.1.3.10.1 Definition

This IOC represents a BsOfdmUlChannel object. It is derived from WmanWmanManagedFunction.

15.1.1.3.10.2 Attributes

Table 13—Attributes of BsOfdmUlChannel

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
BsOfdmUpLinkChannelId	-	+	M	M	M
BsOfdmCtBasedResvTimeout	-	+	O	M	M
BsOfdmBwReqOppSize	-	+	O	M	M
BsOfdmRangReqOppSize	-	+	O	M	M
BsOfdmUplinkCenterFreq	-	+	O	M	M
BsOfdmNumSubChReqRegion-Full	-	+	O	M	M
BsOfdmNumSymbolsReqRegionFull	-	+	O	M	M
BsOfdmSubChFocusCtCode	-	+	O	M	M

15.1.1.3.11 IOC BsOfdmDlChannel

15.1.1.3.11.1 Definition

This IOC represents a BsOfdmDlChannel object. It is derived from WmanManagedFunction.

15.1.1.3.11.2 Attributes

Table 14—Attributes of BsOfdmDlChannel

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
BsOfdmDownLinkChannelId	-	+	M	M	M
BsOfdmBsEIRP	-	+	O	M	M
BsOfdmChannelNumber	-	+	O	M	M
BsOfdmTTG	-	+	O	M	M
BsOfdmRTG	-	+	O	M	M
BsOfdmInitRngMaxRSS	-	+	O	M	M
BsOfdmDownlinkCenterFreq	-	+	O	M	M
BsOfdmBsId	-	+	O	M	M
BsOfdmMacVersion	-	+	O	M	M
BsOfdmFrameDurationCode	-	+	O	M	M

15.1.1.3.12 IOC BsOfdmUcdBurstProfile_F

15.1.1.3.12.1 Definition

This IOC represents a BsOfdmUcdBurstProfile_F object. It is derived from WmanManagedFunction.

15.1.1.3.12.2 Attributes

Table 15—Attributes of BsOfdmUcdBurstProfile_F

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
BsOfdmUcdBurstProfileId	-	+	M	M	M
BsOfdmUiucIndex	-	+	O	-	-
BsOfdmUcdFecCodeType	-	+	O	M	M
BsOfdmFocusCtPowerBoost	-	+	O	M	M
BsOfdmUcdTcsEnable	-	+	O	M	M

15.1.1.3.13 IOC BsOfdmDcdBurstProfile_F

15.1.1.3.13.1 Definition

This IOC represents a BsOfdmDcdBurstProfile_F object. It is derived from WmanManagedFunction.

15.1.1.3.13.2 Attributes

Table 16—Attributes of BsOfdmDcdBurstProfile_

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
BsOfdmDcdBurstProfileId	-	+	M	M	-
BsOfdmDiucIndex	-	+	O	-	-
BsOfdmDownlinkFrequency	-	+	O	M	M
BsOfdmDcdFecCodeType	-	+	O	M	M
BsOfdmDiucMandatoryExitThresh	-	+	O	M	M
BsOfdmDiucMinEntryThresh	-	+	O	M	M
BsOfdmTcsEnable	-	+	O	M	M

15.1.1.3.14 IOC BsClassifierRule_F

15.1.1.3.14.1 Definition

This IOC represents a BsClassifierRule_F object . It is derived from WmanManagedFunction.

15.1.1.3.14.2 Attributes

Table 17—Attributes of BsClassifierRule_F

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
BsClassifierRule_Fld	-	+	M	M	M
BsClassifierRulePriority	-	+	M	M	O
BsClassifierRuleIpTosLow	-	+	M	M	O
BsClassifierRuleIpTosHigh	-	+	M	M	O
BsClassifierRuleIpTosMask	-	+	M	M	O
BsClassifierRuleIpProtocol	-	+	M	M	O
BsClassifierRuleIpSourceAddr	-	+	M	M	O
BsClassifierRuleIpSourceMask	-	+	M	M	O
BsClassifierRuleIpDestAddr	-	+	M	M	O
BsClassifierRuleIpDestMask	-	+	M	M	O
BsClassifierRuleSourcePortStart	-	+	M	M	O
BsClassifierRuleSourcePortEnd	-	+	M	M	O
BsClassifierRuleDestPortStart	-	+	M	M	O
BsClassifierRuleDestPortEnd	-	+	M	M	O
BsClassifierRuleDestMacAddr	-	+	M	M	O
BsClassifierRuleDestMacMask	-	+	M	M	O
BsClassifierRuleSourceMacAddr	-	+	M	M	O
BsClassifierRuleSourceMacMask	-	+	M	M	O
BsClassifierRuleEnetProtocolType	-	+	M	M	O
BsClassifierRuleEnetProtocol	-	+	M	M	O
BsClassifierRuleUserPriLow	-	+	M	M	O
BsClassifierRuleUserPriHigh	-	+	M	M	O
BsClassifierRuleVlanId	-	+	M	M	O
BsClassifierRuleState	-	+	M	M	O

Table 17—Attributes of BsClassifierRule_F

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
BsClassifierRulePhsSize	-	+	M	M	O
BsClassifierRulePhsMask	-	+	M	M	O
BsClassifierRulePhsVerify	-	+	M	M	O
BsClassifierRuleIpv6FlowLabel	-	+	M	M	O

15.1.1.3.15 IOC BsClassifierRule_M**15.1.1.3.15.1 Definition**

This IOC represents a BsClassifierRule_M object . It is derived from WmanManagedFunction.

15.1.1.3.15.2 Attributes**Table 18—Attributes of BsClassifierRule_M**

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	WmanManagedFunction	+inherited	Minherited	Minherited	Minherited
BsClassifierRule_MId	-	+	M	M	M
BsClassifierContextId	-	+	M	M	O
BsClassifierActionRule	-	+	M	M	O
BsClassifierShortFormatContextId	-	+	M	M	O

15.1.1.4 Information relationships definition**15.1.1.4.1 ExternalNeighbourWmanBsRelation****15.1.1.4.1.1 Definition**

This represents a unidirectional relation from BSRelation to the ExternalBSFunction. The role of the relation shall be mapped to a reference attribute, named adjacentBS, of the IOC.

15.1.1.4.1.2 Roles

Table 19—Roles of the relation ExternalNeighbourWmanBsRelation

Name	Definition
BSRelation -ExternalBSFunction	This role (when present) represents BSRelation capability to identify one ExternalBSFunction. When this role is present, the BSRelation.adjacentBS shall contain one ExternalBS DN.

15.1.1.4.1.3 Constraints

This role (for a particular WmanBsRelation) shall be present if the NeighbourWmanBsRelation of this particular WmanBsRelation is absent. This role shall be absent if the NeighbourWmanBsRelation of this particular WmanBsRelation is present.

15.1.1.5 Notifications

15.1.1.6 Information attributes definition

15.1.1.6.1 Definition and legal values

The following table defines the attributes that are present in several Information Object Classes (IOCs) of the present document.

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
OperatorID	Operator Identifier	
BSID	BS Identifier	
HandoverSupportedType	The Handover supported field indicates what type(s) of HO the BS and the MS support.	Type: Enumerated value Range: (MDHO/FBSS HO not supported (0), FBSS/MDHO DLRF combining supported(1), MDHO DL soft combining supported monitoring single MAP from anchor BS(2), MDHO DL soft combining supported monitoring MAPS from active BSs(3))
SystemResourceRetain-Time	The Resource_Retain_Time is the duration for MS's connection information that will be retained in serving BS. BS shall start Resource_Retain_Time timer at MS notification of pending HO attempt through MOB_HO-IND or by detecting an MS drop. The unit of this value is 100 milliseconds.	
HOOptimizationMSTimer	the duration in frames MS shall wait until receipt of the next unsolicited network re-entry MAC management message as indicated in the HO Process Optimization element of the RNG-RSP message.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
MSHOREtransmission-Timer	After a MS transmits MOB_MSHO-REQ to initiate a handover process, it shall start MS Handover Retransmission Timer and shall not transmit another MOB_MSHO-REQ until the expiration of the MS Handover Retransmission Timer.	
MobilitySupportedIndication	The Mobility features supported field indicates whether or not the MS supports mobility modes.	Type: Enumerated value Range :(Handover Support(0), Sleep-mode Support(1), Idle-mode Support(2))
MSHOConnectionProcess-Time	Time in ms the MS needs to process information on connections provided in RNGRSP or REG-RSP message during HO	
MSHOTeKProcessTime	Time in ms the MS needs to completely process TEK information during HO	
ULPermutationBase	Uplink subcarrier allocation	
DLPermutationBase	Downlink subcarrier allocation	
PreambleIndex	Downlink synchronization by MS	
SegmentNumber	An unique segment identifier	
ExternalBSId	External BS Identifier	
FAIndex	Frequency Assignment Index	
BSEIRP	Neighbour BS EIRP	
HOPROcessOptimization	Identifies re-entry process management messages that may be omitted during the current HO attempt due to the availability of MS service and operational context information, and the MS service and operational status post-HO completion.	
SchedulingServiceSupported	Indicate neighbouring BS scheduling service type.	Type: Enumerated value Range: (Non-real-time Polling Service(0), Real-time Polling Service(0), Extended real-time Polling Service(0), Unsolicited Grant Service(0), Best Effort(3))
Bandwidth	Indicate neighbouring BS bandwidth.	
FFTSize	Indicate neighbouring BS FFT size	
CyclePrefix	indicate neighbouring BS Cycle Prefix	
FramDurationCode	Indicate neighbouring BS Frame duration code	
ULPermutationBase	Indicate neighbouring BS uplink permutation base.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
DLPermutationBase	Indicate neighbouring BS uplink permutation base.	
SegmentNumber	Indicate neighbouring BS segment number.	
PreambleIndex	Indicate neighbouring BS preamble index.	
BSRelationId		
adjacentBS	It carries the DN of the BS or the ExternalBS.	
PagingControlId	indicate paging controller identifier connected by BS	
PagingGroupId	indicate the paging group identifier assigned to BS by network	
MgmtResourceHolding-Timer	Time the BS maintain connection information with the MS after the BS send DREG-CMD to the MS	
T46Timer	Time the BS waits for DREGREQ in case of unsolicited Idle Mode initiation from BS	
PagingRetryCount	Number of retries on paging transmission. If the BS does not receive RNG-REQ from the MS until this value decreases to zero, it determines that the MS is unavailable.	
REQDuration	Waiting value for the DREG-REQ message re-transmission(measured in frames)	
MACHashSkipThreshold	Maximum number of successive MOB_PAG-ADV messages that may be sent from a BS without individual notification for an MS for which BS is allowed to skip MS MAC address Hash when the Action Code for the MS is 0b00,'No Action Required'.	
BsCDMATransmissionOpportunityAssignment	The CDMA code and transmission opportunity assignment field indicates the assigned code and transmission opportunity for a MS who is paged to use over dedicated CDMA ranging region	
PagingResponseWindow	The Page-Response Window indicates the Page-Response window for a MS who is paged to transmit the assigned code for CDMA ranging channel.	
IdleModeTimer	MS timed interval to conduct Location Update. Set timer to MS Idle Mode Timeout capabilities setting. Timer recycles on successful Idle Mode Location Update.	Range: (128..65536)
IdleModeSystemTimer	For BS acting as Paging Controller, timed interval to receive notification of MS Idle Mode Location Update. Set timer to MS Idle Mode Timeout. Timer recycles on successful Idle Mode Location Update.	Range: (128..65536)
PagingIntervalLength	time duration of Paging Interval of the BS	Range: (2..5)
PagingCycle	Cycle in which the paging message is transmitted within the paging group.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
powerCtrlId	An attribute whose "name+value" can be used as an RDN when naming an instance of the object class. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
msUpPwrAdjStep	MS-specific up power offset adjustment step	
msDownPwrAdjStep	MS-specific down power offset adjustment step	
minPwrAdjLever	Minimum level of power offset adjustment	
maxPwrAdjLever	Maximum level of power offset adjustment	
txPwrRepThreshold	Tx Power Report Threshold	
txPwrRepInterval	Tx Power Report Interval	
alphaPAvg	Alpha of p_avg	
txPwrRepThresholdCQI	Tx Power Report Threshold,CQICH is allocated to the SS	
txPwrRepIntervalCQI	Tx Power Report Interval,CQICH is allocated to the SS	
alphaPAvgCQI	Alpha of p_avg,CQICH is allocated to the SS	
securityManagementId	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
wmanIfBsPkmBaseId	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
wmanIfBsPkmTekId	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
wmanIfBsMsPkmAuthID	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
wmanIfBsPkmDefaultAuthLifetime	The value of this object is the default lifetime, in seconds, the BS assigns to a new authorization key.	
wmanIfBsPkmDefaultTekLifetime	The value of this object is the default lifetime, in seconds, the BS assigns to a new Traffic Encryption Key(TEK).	
wmanIfBsPkmDefaultSelfSigManufCertTrust	This object determines the default trust of all (new) self-signed manufacturer certificates obtained after setting the object.	trusted (1), untrusted (2)

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
wmanIfBsPkmCheck-CertValidityPeriods	Setting this object to TRUE causes all certificates received thereafter to have their validity periods (and their chain's validity periods) checked against the current time of day. A FALSE setting will cause all certificates received Thereafter to not have their validity periods (nor their chain's validity periods) checked against the current time of day.	TRUE, FALSE
wmanIfBsPMKDefault-PreHandshakeLifetime	The lifetime assigned to PMK when created	
wmanIfBsPMKDefault-Lifetime	If MSK lifetime is unspecified (i.e. by AAA server), PMK lifetime shall be set to this value.(in seconds)	
wmanIfBsDefaultSACHallengeTimer	Time prior to re-send of SA-TEK-Challenge (in seconds)	
wmanIfBsDefaultSaChallengeMaxResends	Maximum number of transmissions of SATEK-Challenge	
wmanIfBsDefaultSATEK-Timer	Time prior to re-send of SA-TEK-Request (in seconds)	
wmanIfBsDefault-SATEKRequestMaxResends	Maximum number of transmissions of SATEK-Request	
wmanIfBsPkmTekSAId	The value of this object is the Security Association ID (SAID).	
wmanIfBsPkmTekSAType	The value of this object is the type of security association. Dynamic does not apply to SSs running in PKM mode.	primarySA (0) staticSA (1) dynamicSA (2)
wmanIfBsPkmTekDataEncryptAlg	The value of this object is the data encryption algorithm being utilized.	No Data Encryption(0) CBC-Mode(1) AES, CCM Mode(2)
wmanIfBsPkmTek-DataAuthentAlg	The value of this object is the data authentication algorithm being utilized.	No Data Authentication(0)
wmanIfBsPkmTekEncryptAlg	The value of this object is the TEK key encryption algorithm being utilized.	3-DES EDE with 128-bit key(1) RSA with 1024-bit key(2) AES with 128-bit key(3)
wmanIfBsPkmTekLifetime	The value of this object is the lifetime, in seconds, the BS assigns to keys for this TEK association.	
wmanIfBsPkmTekKeySequenceNumber	The value of this object is the most recent TEK key sequence number for this SAID.	
wmanIfBsPkmTekExpireOld	The value of this object is the actual clock time for expiration of the immediate predecessor of the most recent TEK for this FSM. If this FSM has only one TEK, then the value is the time of activation of this FSM.	
wmanIfBsPkmTekExpiresNew	The value of this object is the actual clock time for expiration of the most recent TEK for this FSM.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
wmanIfBsPkmTekReset	Setting this object to TRUE causes the BS to invalidate the current active TEK(s) (plural due to key transition periods), and to generate a new TEK for the associated SAID; the BS MAY also generate an unsolicited TEK Invalid message, to optimize the TEK synchronization between the BS and the SS. Reading this object always returns FALSE.	TRUE FALSE
wmanIfBsPkmAssociatedGKEKSequenceNumber	Associated GKEK sequence number with this TEK-Parameters	
wmanIfBsPkmSAServiceType	This attribute indicates service types of the corresponding SA type.	0: Unicast service 1: Group multicast service 2: MBS service 3-255: Reserved.
wmanIfBsSsPkmAuthMacAddress	The value of this object is the physical address of the SS to which the authorization association applies.	
wmanIfBsSsPkmAuthKeySequenceNumber	The value of this object is the most recent authorization key sequence number for this SS.	
wmanIfBsSsPkmAuthExpiresOld	The value of this object is the actual clock time for expiration of the immediate predecessor of the most recent authorization key for this FSM. If this FSM has only one authorization key, then the value is the time of activation of this FSM.	
wmanIfBsSsPkmAuthExpiresNew	The value of this object is the actual clock time for expiration of the most recent authorization key for this FSM	
wmanIfBsSsPkmAuthLifetime	The value of this object is the lifetime, in seconds, the BS assigns to an authorization key for this SS.	
wmanIfBsSsPkmAuthReset	Setting this object to invalidateAuth(2) causes the BS to invalidate the current SS authorization key(s), but not to transmit an Authorization Invalid message nor to invalidate unicast TEKs. Setting this object to sendAuthInvalid(3) causes the BS to invalidate the current SS authorization key(s), and to transmit an Authorization Invalid message to the SS, but not to invalidate unicast TEKs. Setting this object to invalidateTeks(4) causes the BS to invalidate the current SS authorization key(s), to transmit an Authorization Invalid message to the SS, and to invalidate all unicast TEKs associated with this SS authorization. Reading this object returns the most-recently-set value of this object, or returns noResetRequested(1) if the object has not been set since the last BS reboot.	noResetRequested(1), invalidateAuth(2), sendAuthInvalid(3), invalidateTeks(4)
wmanIfBsSsPkmAuthPrimarySAID	The value of this object is the Primary Security Association identifier.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
wmanIfBsSsPkmAuthValidStatus	Contains the reason why a SS's certificate is deemed valid or invalid. Return unknown if the SS is running PKM mode. ValidSsChained means the certificate is valid because it chains to a valid certificate. ValidSsTrusted means the certificate is valid because it has been provisioned to be trusted. InvalidSsUntrusted means the certificate is invalid because it has been provisioned to be untrusted. InvalidCAUntrusted means the certificate is invalid because it chains to an untrusted certificate. InvalidSsOther and InvalidCAOther refer to errors in parsing, validity periods, etc, which are attributable to the SS certificate or its chain respectively.	unknown (0), validSsChained (1), validSsTrusted (2), invalidSsUntrusted (3), invalidCAUntrusted (4), invalidSsOther (5), invalidCAOther (6)
wmanIfBsMsCMACPacketNumbercounter		
wmanIfBsMsCMAC_PN_UL		
wmanIfBsMsCMAC_PN_DL		
wmanIfBsMsCMACValue		
wmanIfBsMsPkmAuthResultCode	Contains the result code of the RSA-based authorization (only for PKMv2)	
wmanIfBsMsPkmAKId	Identify the AK as defined in Table 133	
wmanIfBsKeyPushMode	Distinguish usage code of a PKMv2 Group Key Update Command message	
wmanIfBsKeyPush-Counter	Protect for replay attack.	
BsOfdmUpLinkChannelId	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
BsOfdmDownLinkChannelId	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
BsOfdmUcdBurstProfileId	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
BsOfdmDcdBurstProfileId	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
BsOfdmCtBasedResvTimeout	The number of UL-MAPs to receive before contention-based reservation is attempted again for the same connection.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
BsOfdmBwReqOppSize	Size (in units of PS) of PHY payload that SS may use to format and transmit a bandwidth request message in a contention request opportunity. The value includes all PHY overhead as well as allowance for the MAC data the message may hold.	
BsOfdmRangReqOppSize	Size (in units of PS) of PHY payload that SS may use to format and transmit a RNG-REQ message in a contention request opportunity. The value includes all PHY overhead as well as allowance for the MAC data the message may hold and the maximum SS/BS roundtrip propagation delay.	
BsOfdmUplinkCenterFreq	Uplink center frequency (kHz)	
BsOfdmNumSubChReqRegionFull	Number of subchannels used by each transmit opportunity when REQ Region-Full is allocated in subchannelization region.	oneSubchannel(0), twoSubchannels(1), fourSubchannels(2), eightSubchannels(3), sixteenSubchannels(4)
BsOfdmNumSymbolsReqRegionFull	Number of OFDM symbols used by each transmit opportunity when REQ Region-Full is allocated in subchannelization region.	
BsOfdmSubChFocusCtCode	Number of contention codes (CSE) that shall only be used to request a subchanneled allocation.	Default value 0. Allowed values 0-8.
BsOfdmBsEIRP	The EIRP is the equivalent isotropic radiated power of the base station, which is computed for a simple single-antenna transmitter.	
BsOfdmChannelNumber	Downlink channel number as defined in 8.5. Used for license-exempt operation only.	
BsOfdmTTG	Transmit / Receive Transition Gap.	
BsOfdmRTG	Receive / Transmit Transition Gap.	
BsOfdmInitRngMaxRSS	Initial Ranging Max. Received Signal Strength at BS Signed in units of 1 dBm.	
BsOfdmDownlinkCenterFreq	Downlink center frequency (kHz).	
BsOfdmBsId	Base station ID.	
BsOfdmMacVersion	This parameter specifies the version of 802.16 to which the message originator conforms.	
BsOfdmFrameDurationCode	The duration of the frame. The frame duration code values are specified in Table 230.	
BsOfdmUiucIndex	The Uplink Interval Usage Code indicates the uplink burst profile in the UCD message, and is used along with ifIndex to identify an entry in the wmanIfBsOfdmUcdBurstProfileTable.	
BsOfdmUcdFecCodeType	Uplink FEC code type and modulation type	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
BsOfdmFocusCtPowerBoost	The power boost in dB of focused contention carriers	
BsOfdmUcdTcsEnable	This parameter determines the transmission convergence sublayer, as described in 8.1.4.3, can be enabled on a per-burst basis for both uplink and downlink. Through DIUC/UIUC messages.	tcsDisabled(0), tcsEnabled(1)
BsOfdmDiucIndex	The Downlink Interval Usage Code indicates the downlink burst profile in the DCD message, and is used along with ifIndex to identify an entry in the wmanIfBsOfdmDcdBurstProfileTable.	
BsOfdmDownlinkFrequency	Downlink Frequency (kHz).	
BsOfdmDcdFecCodeType	Downlink FEC code type and modulation type	
BsOfdmDiucMandatoryExitThresh	DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or below where this DIUC can no longer be used and where this change to a more robust DIUC is required in 0.25 dB units.	
BsOfdmDiucMinEntryThresh	DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR required to start using this DIUC when changing from a more robust DIUC is required, in 0.25 dB units.	
BsOfdmTcsEnable	Indicates whether Transmission Convergence Sublayer is enabled or disabled.	tcsDisabled(0), tcsEnabled(1)
BsClassifierRule_Fld	It contains 'name+value' that is the RDN, when naming an instance, of this object class containing this attribute. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
BsClassifierRulePriority	The value specifies the priority for the Classifier, which is used for determining the order of the Classifier. A higher value indicates higher priority. Classifiers may have priorities in the range 0..255.	0..255
BsClassifierRuleIpTosLow	The low value of a range of TOS byte values. If the referenced parameter is not present in a classifier, this object reports the value of 0.	
BsClassifierRuleIpTosHigh	The 8-bit high value of a range of TOS byte values. If the referenced parameter is not present in a classifier, this object reports the value of 0.	
BsClassifierRuleIpTosMask	The value of this object specifies the matching parameter for the IP type of service/DSCP [IETF RFC 2474] byte mask. An IP packet with IP type of service (ToS) byte value ip-tos matches this parameter if tos-low less than or equal (ip-tos AND tos-mask) less than or equal tos-high.	
BsClassifierRuleIpProtocol	This object indicates the value of the IP Protocol field required for IP packets to match this rule. If the referenced parameter is not present in a classifier, this object reports the value of 0.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
BsClassifierRuleIpSourceAddr	This object specifies the value of the IP Source Address required for packets to match this rule. An IP packet matches the rule when the packet ip source address bitwise ANDed with the BsClassifierRuleIpSourceMask value equals the BsClassifierRuleIpSourceAddr value. If the referenced parameter is not present in a classifier, this object reports the value of 0.0.0.0.	
BsClassifierRuleIpSourceMask	This object specifies which bits of a packet's IP Source Address that are compared to match this rule. An IP packet matches the rule when the packet source address bitwise ANDed with the BsClassifierRuleIpSourceMask value equals the BsClassifierRuleIpSourceAddr value. If the referenced parameter is not present in a classifier, this object reports the value of 0.0.0.0.	
BsClassifierRuleIpDestAddr	This object specifies the value of the IP Destination Address required for packets to match this rule. An IP packet matches the rule when the packet IP destination address bitwise ANDed with the BsClassifierRuleIpDestMask value equals the BsClassifierRuleIpDestAddr value. If the referenced parameter is not present in a classifier, this object reports the value of 0.0.0.0.	
BsClassifierRuleIpDestMask	This object specifies which bits of a packet's IP Destination Address that are compared to match this rule. An IP packet matches the rule when the packet destination address bitwise ANDed with the BsClassifierRuleIpDestMask value equals the BsClassifierRuleIpDestAddr value. If the referenced parameter is not present in a classifier, this object reports the value of 0.0.0.0.	
BsClassifierRuleSourcePortStart	This object specifies the low end inclusive range of TCP/UDP source port numbers to which a packet is compared. This object is irrelevant for non-TCP/UDP IP packets. If the referenced parameter is not present in a classifier, this object reports the value of 0.	
BsClassifierRuleSourcePortEnd	This object specifies the high end inclusive range of TCP/UDP source port numbers to which a packet is compared. This object is irrelevant for non-TCP/UDP IP packets. If the referenced parameter is not present in a classifier, this object reports the value of 65535.	
BsClassifierRuleDestPortStart	This object specifies the low end inclusive range of TCP/UDP destination port numbers to which a packet is compared. If the referenced parameter is not present in a classifier, this object reports the value of 0.	
BsClassifierRuleDestPortEnd	This object specifies the high end inclusive range of TCP/UDP destination port numbers to which a packet is compared. If the referenced parameter is not present in a classifier, this object reports the value of 65535.	
BsClassifierRuleDestMacAddr	An Ethernet packet matches an entry when its destination MAC address bitwise ANDed with BsClassifierRuleDestMacMask equals the value of BsClassifierRuleDestMacAddr. If the referenced parameter is not present in a classifier, this object reports the value of '000000000000'H.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
BsClassifierRuleDestMacMask	An Ethernet packet matches an entry when its destination MAC address bitwise ANDed with BsClassifierRuleDestMacMask equals the value of BsClassifierRuleDestMacAddr. If the referenced parameter is not present in a classifier, this object reports the value of '000000000000'H.	
BsClassifierRuleSourceMacAddr	An Ethernet packet matches this entry when its source MAC address bitwise ANDed with BsClassifierRuleSourceMacMask equals the value of BsClassifierRuleSourceMacAddr. If the referenced parameter is not present in a classifier, this object reports the value of '000000000000'H.	
BsClassifierRuleSourceMacMask	An Ethernet packet matches an entry when its source MAC address bitwise ANDed with BsClassifierRuleSourceMacMask equals the value of BsClassifierRuleSourceMacAddr. If the referenced parameter is not present in a classifier, this object reports the value of '000000000000'H.	
BsClassifierRuleEnetProtocolType	This object indicates the format of the layer 3 protocol id in the Ethernet packet. A value of none(0) means that the rule does not use the layer 3 protocol type as a matching criteria. A value of ethertype(1) means that the rule applies only to frames which contains an EtherType value. Ethertype values are contained in packets using the Dec-Intel-Xerox (DIX) encapsulation or the RFC1042 Sub-Network Access Protocol (SNAP) encapsulation formats. A value of dsap(2) means that the rule applies only to frames using the IEEE802.3 encapsulation format with a Destination Service Access Point (DSAP) other than 0xAA(which is reserved for SNAP). If the Ethernet frame contains an 802.1P/Q Tag header (i.e. EtherType 0x8100), this object applies to the embedded EtherType field within the 802.1P/Q header. If the referenced parameter is not present in a classifier, this object reports the value of 0.	none(0), ethertype(1), dsap(2)
BsClassifierRuleEnetProtocol	If BsClassifierRuleEnetProtocolType is none(0),this object is ignored when considering whether a packet matches the current rule. If BsClassifierRuleEnetProtocolType is ethertype(1), this object gives the 16-bit value of the EtherType that the packet must match in order to match the rule. If BsClassifierRuleEnetProtocolType is dsap(2), the lower 8 bits of this object's value must match the DSAP byte of the packet in order to match the rule. If the Ethernet frame contains an 802.1P/Q Tag header (i.e. EtherType 0x8100), this object applies to the embedded EtherType field within the 802.1P/Q header. If the referenced parameter is not present in the classifier, the value of this object is reported as 0.	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
BsClassifierRuleUserPriLow	This object applies only to Ethernet frames using the 802.1P/Q tag header (indicated with EtherType 0x8100). Such frames include a 16-bit Tag that contains a 3 bit Priority field and a 12 bit VLAN number. Tagged Ethernet packets must have a 3-bit Priority field within the range of BsClassifierRuleUserPriLow and BsClassifierRuleUserPriHigh in order to match this rule. If the referenced parameter is not present in the classifier, the value of this object is reported as 0.	
BsClassifierRuleUserPriHigh	This object applies only to Ethernet frames using the 802.1P/Q tag header (indicated with EtherType 0x8100). Such frames include a 16-bit Tag that contains a 3 bit Priority field and a 12 bit VLAN number. Tagged Ethernet packets must have a 3-bit Priority field within the range of BsClassifierRuleUserPriLow and BsClassifierRuleUserPriHigh in order to match this rule. If the referenced parameter is not present in the classifier, the value of this object is reported as 7.	
BsClassifierRuleVlanId	This object applies only to Ethernet frames using the 802.1P/Q tag header. If this object's value is nonzero, tagged packets must have a VLAN Identifier that matches the value in order to match the rule. Only the least significant 12 bits of this object's value are valid. If the referenced parameter is not present in the classifier, the value of this object is reported as 0.	
BsClassifierRuleState	This object indicates whether or not the classifier is enabled to classify packets to a Service Flow. If the referenced parameter is not present in the classifier, the value of this object is reported as active(1).	active(1), inactive(2)
BsClassifierRulePhsSize	This object is used to configure the PHS rule for this classifier. The value of this field - PHSS is the total number of bytes in the header to be suppressed and then restored in a service flow that uses PHS. If the value of this field is 0 bytes then PHS is disabled for this classifier. If flag phs-Mask in BsClassifierRuleBitMap is set to 0 and flag phs-Size in BsClassifierRuleBitMap is set to 0, then BS can still create PHS rules using its own custom mask (i.e. the rule is not configured by NMS).	

Table 20—Information attributes definition

Attribute Name	Definition	Legal Values
BsClassifierRulePhsMask	This object is used to configure the PHS rule for this classifier. It is encoded as follows bit 0: 0 = don't suppress the 1st byte of the suppression field 1 = suppress first byte of the suppression field bit 1: 0 = don't suppress the 2nd byte of the suppression field 1 = suppress second byte of the suppression field bit x: 0 = don't suppress the (x+1) byte of the suppression field 1 = suppress (x+1) byte of the suppression field where the length of the octet string is ceiling (BsClassifierRulePhs-Size/8). BS should use this value to create a new PHS rule index (PHSI) and field (PHSF) as defined in the standard. If flag phsMask in BsClassifierRuleBitMap is set to 0 and flag phsSize in BsClassifierRuleBitMap is set to 0, then BS can still create PHS rules using its own custom mask (i.e. the rule is not configured by NMS).	
BsClassifierRulePhsVerify	The value of this field indicates to the sending entity whether or not the packet header contents are to be verified prior to performing suppression.	
BsClassifierRuleIpv6FlowLabel	The value of this field specifies the matching values for the IPv6 Flow label field.	
BsClassifierContextId	The values of the field specify the context ID for ROHC- or ECRTP-compressed packets. The CS will attempt to match the context ID with the payload packet's one-byte or two-byte embedded Context ID field according to the scheme described in RFC 3095 section 5.1.3.	
BsClassifierActionRule	The value of this field specifies an action associate with the classifier rule.	bit 0: 0 = none. 1 = Discard packet bit 1-7: Reserved.
BsClassifierShortFormat-ContextId	The values of the field specify a short-format context ID for ROHC- or ECRTP-compressed packets. The CS will attempt to match the context ID with the payload packet's zero- or one-byte prefix Context ID field according to the scheme described in RFC 3095 section 5.1.3.	

15.2 NRM IRP SNMP Solution Set

15.2.1 wmanIf2Mib

Figure 18 shows the high level MIB structure of wmanIf2Mib for 802.16. The MIB structure is organized based on the the reference model as defined in IEEE 802.16-2004 standard.

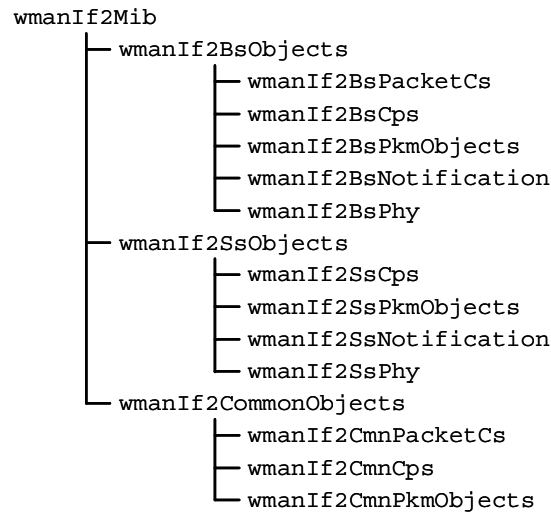


Figure 18—wmanIf2Mib structure

15.2.1.1 wmanIf2BsObjects

15.2.1.1.1 wmanIf2BsPacketCs

Figure 19 shows the structure of wmanIf2BsPacketCs subtree that contains BS managed objects related to the Packet CS management entity layer.

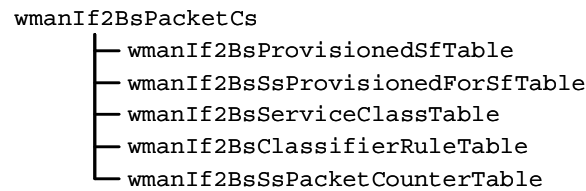


Figure 19—wmanIf2BsPacketCs structure

15.2.1.1.1.1 wmanIf2BsProvisionedSfTable

wmanIf2BsProvisionedSfTable contains provisioned service flow profiles for SSs, and pointers to wmanIf2BsServiceClassTable and wmanIf2BsClassifierRuleTable for QoS parameters and classifier rules respectively.

15.2.1.1.1.2 wmanIf2BsProvisionedForSfTable

wmanIf2BsProvisionedForSfTable maps the MAC addresses of SSs to the service flows provisioned in wmanIf2BsProvisionedSfTable. It enables downlink multicast services where MAC addresses of multiple SSs can be mapped to the same service flow.

15.2.1.1.1.3 wmanIf2BsServiceClassTable

Each entry of the wmanIf2BsServiceClassTable contains QoS parameter set, as defined in subclause 6.3.14 and 11.13 in IEEE 802.16-2004 standard.

15.2.1.1.1.4 wmanIf2BsClassifierRuleTable

wmanIf2BsClassifierRuleTable contains the packet classifier rules associated with service flows.

15.2.1.1.1.5 wmanIf2BsSsPacketCounterTable

wmanIf2BsSsPacketCounterTable contains counters to keep track of the number of packets and octets that have been received or transmitted on the per service flow basis.

15.2.1.1.2 wmanIf2BsCps

Figure 20 shows the structure of wmanIf2BsCps subtree that contains BS managed objects related to the MAC CPS management entity layer.

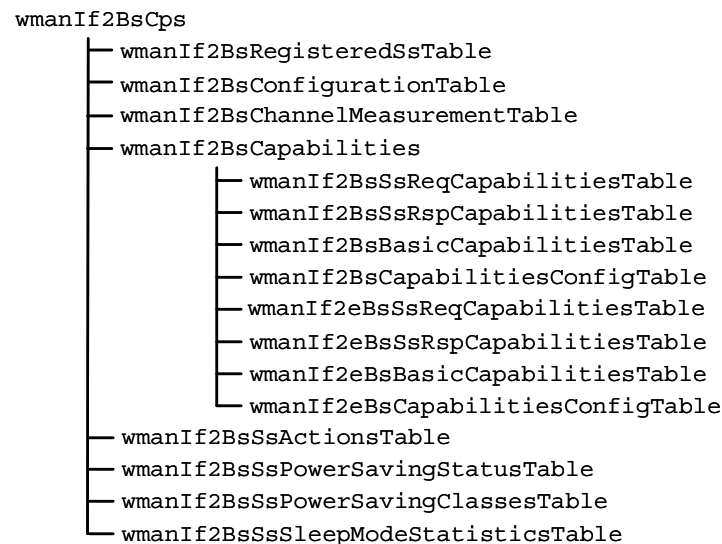


Figure 20—wmanIf2BsCps structure

15.2.1.1.2.1 wmanIf2BsRegisteredSsTable

Each entry in the wmanIf2BsRegisteredSsTable contains the information of SS that has been registered through REG-REQ and REG-RSP messages.

15.2.1.1.2.2 wmanIf2BsConfigurationTable

wmanIf2BsConfigurationTable contains objects for BS system parameters and constants as defined in subclause 10.1 of IEEE 802.16-2004 standard. wmanIf2BsConfigurationTable also contains objects that define the default behaviour of the BS for 2nd Management Channel scheduling and SFID allocation as well as configuration parameters of the CPS scheduler and AAS system.

15.2.1.1.2.3 wmanIf2BsChannelMeasurementTable

wmanIf2BsChannelMeasurementTable contains channel measurement information on the uplink signal that were received from SS, and the downlink signal were obtained from SS using REP-REQ/RSP messages..

15.2.1.1.2.4 wmanIf2BsCapabilities

15.2.1.1.2.4.1 wmanIf2BsSsReqCapabilitiesTable

wmanIf2BsSsReqCapabilitiesTable contains the basic capability information of SSs that have been reported by SSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.

15.2.1.1.2.4.2 wmanIf2BsSsRspCapabilitiesTable

wmanIf2BsSsRspCapabilitiesTable contains the basic capability information of SSs that have been negotiated and agreed between BS and SS via RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages.

15.2.1.1.2.4.3 wmanIf2BsBasicCapabilitiesTable

wmanIf2BsBasicCapabilitiesTable contains the basic capabilities of the BS as implemented in BS hardware and software. These capabilities along with the configuration for them (wmanIf2BsCapabilitiesConfigTable) are used for negotiation of basic capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP messages.

15.2.1.1.2.4.4 wmanIf2BsCapabilitiesConfigTable

wmanIf2BsCapabilitiesConfigTable contains the configuration for basic capabilities of BS. The table is intended to be used to restrict the Capabilities implemented by BS, for example in order to comply with local regulatory requirements. The BS should use the configuration along with the implemented Capabilities (wmanIf2BsBasicCapabilitiesTable) for negotiation of basic capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP messages.

15.2.1.1.2.4.5 wmanIf2eBsSsReqCapabilitiesTable

wmanIf2eBsSsReqCapabilitiesTable is the AUGMENTS to wmanIf2BsRegisteredSsTable to contain new basic capability information of SS that have been added to IEEE 802.16e 2005.

15.2.1.1.2.4.6 wmanIf2eBsSsRspCapabilitiesTable

wmanIf2eBsSsRspCapabilitiesTable is the AUGMENTS to wmanIf2BsRegisteredSsTable to contain new basic capability information of SS that have been added to IEEE 802.16e 2005.

15.2.1.1.2.4.7 wmanIf2eBsBasicCapabilitiesTable

wmanIf2eBsBasicCapabilitiesTable is the AUGMENTS to wmanIf2BsRegisteredSsTable to contain new basic capability information of SS that have been added to IEEE 802.16e 2005.

15.2.1.1.2.4.8 wmanIf2eBsCapabilitiesConfigTable

wmanIf2eBsCapabilitiesConfigTable is the AUGMENTS to wmanIf2BsRegisteredSsTable to contain new basic capability information of SS that have been added to IEEE 802.16e 2005.

15.2.1.1.2.5 wmanIf2BsSsActionsTable

wmanIf2BsSsActionsTable contains all the actions specified for SSs in the standard. The actions are routed down to SS using nsolicited MAC messages: REG-RSP, DREG-REQ and RES-CMD. The table also contains the parameters of the actions in cases where they are specified by the standard.

15.2.1.1.2.6 wmanIfBsSsPowerSavingStatusTable

wmanIfBsSsPowerSavingStatusTable contains the power saving status for each CID in an SS.

15.2.1.1.2.7 wmanIfBsSsPowerSavingClassesTable

wmanIfBsSsPowerSavingClassesTable contains the power saving classes definitions, and activation / deactivation information that are provided by MOB_SLP-REQ and MOB_SLP-RSP messages.

15.2.1.1.2.8 wmanIfBsSsSleepModeStatisticsTable

wmanIfBsSsSleepModeStatisticsTable contains the sleep mode statistic for SS.

15.2.1.1.3 wmanIf2BsPkmObjects

Figure 21 shows the structure of wmanIf2BsPkmObjects subtree that contains BS managed objects related to the MAC privacy management entity.

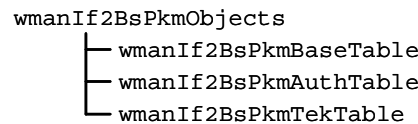


Figure 21—wmanIf2BsPkmObjects structure

15.2.1.1.3.1 wmanIf2BsPkmBaseTable

wmanIf2BsPkmBaseTable contains base station PKM operational parameters described in subclause 10.2 of IEEE 802.16-2004 standard.

15.2.1.1.3.2 wmanIf2BsSsPkmAuthTable

wmanIf2BsSsPkmAuthTable contains runtime subscriber station authentication and authorization parameters for each base station.

15.2.1.1.3.3 wmanIf2BsPkmTekTable

wmanIf2BsPkmTekTable is double indexed by ifIndex and SAId and contains runtime Security association parameters for each base station.

15.2.1.1.4 wmanIf2BsNotification

Figure 22 shows the structure of wmanIf2BsNotification subtree that contains BS traps to report fault events and exceptions, such as power status, RSSI threshold crossing.

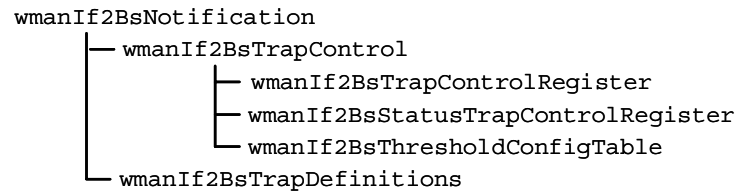


Figure 22—wmanIf2BsNotification structure

15.2.1.1.4.1 wmanIf2BsTrapControl

15.2.1.1.4.1.1 wmanIf2BsTrapControlRegister

wmanIf2BsTrapControlRegister is used to enable or disable Base traps independently.

15.2.1.1.4.1.2 wmanIf2BsStatusTrapControlRegister

wmanIf2BsStatusTrapControlRegister is used to enable or disable Base Station status notification traps.

15.2.1.1.4.1.3 wmanIf2BsThresholdConfigTable

wmanIf2BsThresholdConfigTable contains threshold objects that can be set to detect the threshold crossing events.

15.2.1.1.4.2 wmanIf2BsTrapDefinitions

wmanIf2BsTrapDefinitions object group defines all the traps reported by BS.

15.2.1.1.5 wmanIf2BsPhy

Figure 23 shows the structure of wmanIf2BsPhy subtree that contains BS managed objects related to the Physical layer.

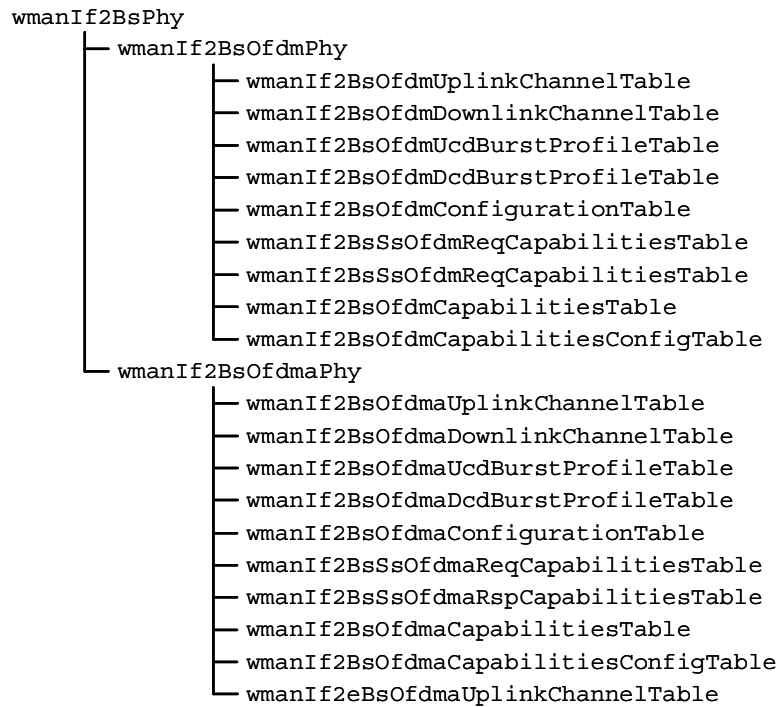


Figure 23—wmanIf2BsPhy structure

15.2.1.1.5.1 wmanIf2BsOfdmPhy

wmanIf2BsOfdmPhy is a group containing objects specific to OFDM PHY.

15.2.1.1.5.1.1 wmanIf2BsOfdmUplinkChannelTable

wmanIf2BsOfdmUplinkChannelTable contains OFDM UCD (Uplink Channel Descriptor) channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.1.5.1.2 wmanIf2BsOfdmDownlinkChannelTable

wmanIf2BsOfdmDownlinkChannelTable contains OFDM DCD (Downlink Channel Descriptor) channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.1.5.1.3 wmanIf2BsOfdmUcdBurstProfileTable

wmanIf2BsOfdmUcdBurstProfileTable contains OFDM UCD burst profiles for each uplink channel.

15.2.1.1.5.1.4 wmanIf2BsOfdmDcdBurstProfileTable

wmanIf2BsOfdmDcdBurstProfileTable provides one row for each OFDM DCD burst profile.

15.2.1.1.5.1.5 wmanIf2BsOfdmConfigurationTable

wmanIf2BsOfdmConfigurationTable contains BS configuration objects, specific to OFDM PHY.

15.2.1.1.5.1.6 wmanIf2BsSsOfdmReqCapabilitiesTable

wmanIf2BsSsOfdmReqCapabilitiesTable contains the basic capability information, specific to OFDM Phy, of SSs that have been reported by SSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages. Entries in this table should be created when an SS registers with a BS.

15.2.1.1.5.1.7 wmanIf2BsSsOfdmRspCapabilitiesTable

wmanIf2BsSsOfdmRspCapabilitiesTable contains the basic capability information, specific to OFDM Phy, of SSs that have been negotiated and agreed between BS and SS via RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages. This table augments the wmanIf2BsRegisteredSsTable.

15.2.1.1.5.1.8 wmanIf2BsOfdmCapabilitiesTable

wmanIf2BsOfdmCapabilitiesTable contains the basic capabilities, specific to OFDM Phy, of the BS as implemented in BS hardware and software.

15.2.1.1.5.1.9 wmanIf2BsOfdmCapabilitiesConfigTable

wmanIf2BsOfdmCapabilitiesConfigTable contains the configuration for basic capabilities of BS, specific to OFDM Phy. The table is intended to be used to restrict the Capabilities implemented by BS.

15.2.1.1.5.1.10 wmanIf2eBsOfdmaUplinkChannelTable

wmanIf2eBsOfdmaUplinkChannelTable is the AUGMENTS to wmanIf2BsOfdmaUplinkChannelTable to contain new UCD attributes that have been added to IEEE 802.16e 2005.

15.2.1.1.5.2 wmanIf2BsOfdmaPhy

wmanIf2BsOfdmaPhy is a group containing objects specific to OFDMA PHY.

15.2.1.1.5.2.1 wmanIf2BsOfdmaUplinkChannelTable

wmanIf2BsOfdmaUplinkChannelTable contains OFDMA UCD channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.1.5.2.2 wmanIf2BsOfdmaDownlinkChannelTable

wmanIf2BsOfdmaDownlinkChannelTable contains OFDMA DCD channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.1.5.2.3 wmanIf2BsOfdmaUcdBurstProfileTable

wmanIf2BsOfdmaUcdBurstProfileTable contains OFDMA UCD burst profiles for each uplink channel.

15.2.1.1.5.2.4 wmanIf2BsOfdmaDcdBurstProfileTable

wmanIf2BsOfdmaDcdBurstProfileTable provides one row for each OFDMA DCD burst profile.

15.2.1.1.5.2.5 wmanIf2BsOfdmaConfigurationTable

wmanIf2BsOfdmaConfigurationTable contains BS configuration objects, specific to OFDMA PHY.

15.2.1.1.5.2.6 wmanIf2BsSsOfdmaReqCapabilitiesTable

wmanIf2BsSsOfdmaReqCapabilitiesTable contains the basic capability information, specific to OFDMA Phy, of SSs or MSs that have been reported by SSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages. Entries in this table should be created when an SS registers with a BS.

15.2.1.1.5.2.7 wmanIf2BsSsOfdmaRspCapabilitiesTable

wmanIf2BsSsOfdmaRspCapabilitiesTable contains the basic capability information, specific to OFDMA Phy, of SSs or MSs that have been negotiated and agreed between BS and SS via RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages. This table augments the wmanIf2BsRegisteredSsTable.

15.2.1.1.5.2.8 wmanIf2BsOfdmaCapabilitiesTable

wmanIf2BsOfdmaCapabilitiesTable contains the basic capabilities, specific to OFDMA Phy, of the BS as implemented in BS hardware and software.

15.2.1.1.5.2.9 wmanIf2BsOfdmaCapabilitiesConfigTable

wmanIf2BsOfdmaCapabilitiesConfigTable contains the configuration for basic capabilities of BS, specific to OFDMA Phy. The table is intended to be used to restrict the Capabilities implemented by BS.

15.2.1.2 wmanIf2SsObjects

15.2.1.2.1 wmanIf2SsCps

Figure 24 shows the structure of wmanIf2SsCps subtree that contains SS managed objects related to the MAC CPS management entity layer.

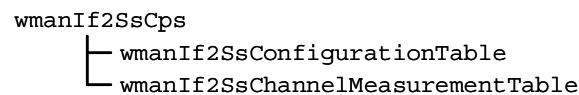


Figure 24—wmanIf2SsCps structure

15.2.1.2.1.1 wmanIf2SsConfigurationTable

wmanIf2SsConfigurationTable contains objects for SS system parameters and constants as defined in sub-clause 10.1 of IEEE 802.16-2004 standard.

15.2.1.2.1.2 wmanIf2SsChannelMeasurementTable

wmanIf2SsChannelMeasurementTable contains downlink channel measurement information for each SS.

15.2.1.2.2 wmanIf2SsPkmObjects

Figure 25 shows the structure of wmanIf2SsPkmObjects subtree that contains subscriber station manageable objects related to the privacy management entity.

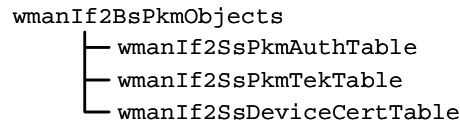


Figure 25—wmanIf2SsPkmObjects structure

15.2.1.2.2.1 wmanIf2SsPkmAuthTable

wmanIf2SsPkmAuthTable contains subscriber station authentication and authorization parameters including those described in subclause 10.2 of IEEE 802.16-2004.

15.2.1.2.2.2 wmanIf2SsPkmTekTable

wmanIf2SsPkmTekTable contains subscriber station runtime parameters for each active security association.

15.2.1.2.2.3 wmanIf2SsDeviceCertTable

wmanIf2SsDeviceCertTable describes the PKM device certificates for each SS wireless interface.

15.2.1.2.3 wmanIf2SsNotification

Figure 26 shows the structure of wmanIf2SsNotification subtree that contains SS traps to report fault events and exceptions, such as RSSI threshold crossing.

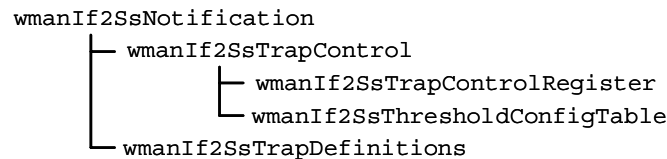


Figure 26—wmanIf2SsNotification structure

15.2.1.2.3.1 wmanIf2SsTrapControl

15.2.1.2.3.1.1 wmanIf2SsTrapControlRegister

wmanIf2SsTrapControlRegister is used to enable or disable Subscriber Station traps.

15.2.1.2.3.1.2 wmanIf2SsThresholdConfigTable

wmanIf2SsThresholdConfigTable contains threshold objects that can be set to detect the threshold crossing events.

15.2.1.2.3.2 wmanIf2SsTrapDefinitions

wmanIf2SsTrapDefinitions group defines all the traps reported by SS.

15.2.1.2.4 wmanIf2SsPhy

Figure 27 shows the structure of wmanIf2SsPhy subtree that contains SS managed objects related to the Physical layer.

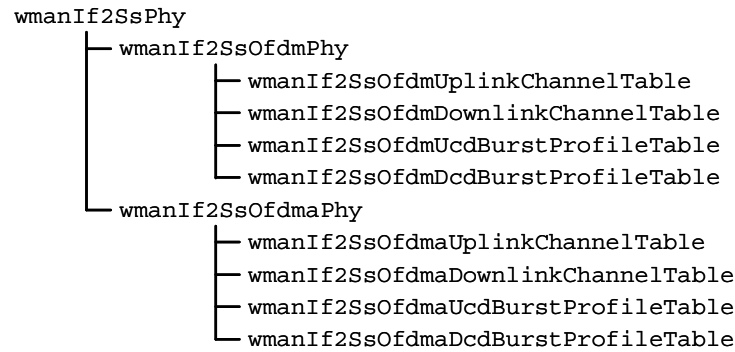


Figure 27—wmanIf2SsPhy structure

15.2.1.2.4.1 wmanIf2SsOfdmPhy

wmanIf2SsOfdmPhy is a group containing objects specific to OFDM PHY.

15.2.1.2.4.1.1 wmanIf2SsOfdmUplinkChannelTable

wmanIf2SsOfdmUplinkChannelTable contains OFDM UCD channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.2.4.1.2 wmanIf2SsOfdmDownlinkChannelTable

wmanIf2SsOfdmUplinkChannelTable contains OFDM DCD channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.2.4.1.3 wmanIf2SsOfdmUcdBurstProfileTable

wmanIf2SsOfdmUcdBurstProfileTable contains OFDM UCD burst profiles for each uplink channel.

15.2.1.2.4.1.4 wmanIf2SsOfdmDcdBurstProfileTable

wmanIf2SsOfdmDcdBurstProfileTable provides one row for each OFDM DCD burst profile.

15.2.1.2.4.2 wmanIf2SsOfdmaPhy

wmanIf2SsOfdmaPhy is a group containing objects specific to OFDMA PHY.

15.2.1.2.4.2.1 wmanIf2SsOfdmaUplinkChannelTable

wmanIf2SsOfdmaUplinkChannelTable contains OFDMA UCD channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.2.4.2.2 wmanIf2SsOfdmaDownlinkChannelTable

wmanIf2SsOfdmaDownlinkChannelTable contains OFDMA DCD channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.2.4.2.3 wmanIf2SsOfdmaUcdBurstProfileTable

wmanIf2SsOfdmaUcdBurstProfileTable contains OFDMA UCD burst profiles for each uplink channel.

15.2.1.2.4.2.4 wmanIf2SsOfdmaDcdBurstProfileTable

wmanIf2SsOfdmaDcdBurstProfileTable provides one row for each OFDMA DCD burst profile.

15.2.1.3 wmanIf2CommonObjects

15.2.1.3.1 wmanIf2CmnPacketCs

Figure 28 shows the structure of wmanIf2CmnPacketCs subtree that contains common managed objects related to the Packet CS management entity layer.

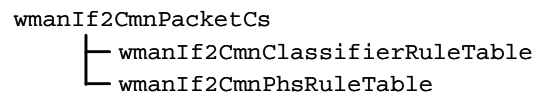


Figure 28—wmanIf2CmnPacketCs structure

15.2.1.3.1.1 wmanIf2CmnClassifierRuleTable

wmanIf2CmnClassifierRuleTable contains runtime classifier rules screening criteria for each service flow.

15.2.1.3.1.2 wmanIf2CmnPhsRuleTable

wmanIf2CmnPhsRuleTable contains PHS rule dictionary entries. Each entry contains the data of the header to be suppressed along with its identification - PHSI.

15.2.1.3.2 wmanIf2CmnCps

Figure 29 shows the structure of wmanIf2CmnCps subtree that contains common managed objects related to the MAC CPS management entity.

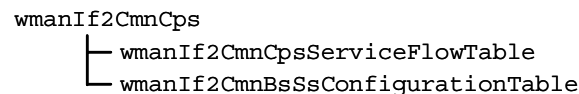


Figure 29—wmanIf2CmnCps structure

15.2.1.3.2.1 wmanIf2CmnCpsServiceFlowTable

wmanIf2CmnCpsServiceFlowTable contains Service Flow managed objects that are common in BS and SS.

15.2.1.3.2.2 wmanIf2CmnBsSsConfigurationTable

wmanIf2CmnBsSsConfigurationTable provides one row for each BS sector that contains the system parameters common in both SS and BS. All SSs shall have the same parameters as the BS to which the SSs are associated.

15.2.1.3.3 wmanIf2CmnPkmObjects

Figure 30 shows the structure of wmanIf2CmnPkmObjects subtree that contains common PKM objects.

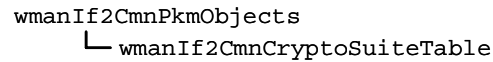


Figure 30—wmanIf2CmnPkmObjects structure

15.2.1.3.3.1 wmanIf2CmnCryptoSuiteTable

wmanIf2CmnCryptoSuiteTable contains supported crypto suites for the particular SS and other crypto parameters such as key lifetimes.

15.2.2 wmanIf2mMib

wmanIf2mMib for 802.16. defines managed objects that are specific to mobile Broadband Wireless Networks.

15.2.3 ASN.1 Definitions of 802.16 MIB for SNMP

15.2.3.1 WMAN-IF2-MIB

WMAN-IF2-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY,
 OBJECT-TYPE,
 NOTIFICATION-TYPE,
 Unsigned32, Integer32, Counter32,
 Counter64, transmission
 FROM SNMPv2-SMI
 SnmpAdminString
 FROM SNMP-FRAMEWORK-MIB
 TEXTUAL-CONVENTION,
 MacAddress, RowStatus, TruthValue,
 TimeStamp, DateAndTime
 FROM SNMPv2-TC
 InetAddressType, InetAddress
 FROM INET-ADDRESS-MIB
 OBJECT-GROUP,
 MODULE-COMPLIANCE,
 NOTIFICATION-GROUP
 FROM SNMPv2-CONF
 ifIndex
 FROM IF-MIB;

wmanIf2Mib MODULE-IDENTITY

LAST-UPDATED "200610160000Z" -- October 16, 2006
 ORGANIZATION "IEEE 802.16"
 CONTACT-INFO

"WG E-mail: stds-802-16@ieee.org
 WG Chair: Roger B. Marks
 Postal: (U.S.) National Institute
 of Standards and Technology
 E-mail: r.b.marks@ieee.org

TGf Chair: Phillip Barber
 Postal: Huawei Technologies Co., Ltd
 E-mail: pbarber@futurewei.com

Editor: Joey Chou
 Postal: Intel Corporation
 5000 W. Chandler Blvd,
 Chandler, AZ 85227, USA
 E-mail: joey.chou@intel.com"

DESCRIPTION

"This material is from IEEE Std 802.16i
 Copyright (c) 2006 IEEE.
 This MIB Module defines managed objects for
 Subscriber Station and Base Station based on IEEE Std
 802.16-2004 and its amendment IEEE Std 802.16e-2005.
 The MIB contains managed objects that are common for

```

1      both fixed and mobile Broadband Wireless Networks."
2  REVISION      "200610160000Z"
3  DESCRIPTION
4      "The second revision of WMAN2-IF2-MIB module."
5  REVISION      "200605230000Z"
6  DESCRIPTION
7      "The first revision of WMAN-IF2-MIB module that is
8      enhanced to support IEEE 802.16e-2005 standard."
9  ::= { transmission 184 }
10
11  wmanIf2MibObjects OBJECT IDENTIFIER ::= { wmanIf2Mib 1 }
12  wmanIf2BsObjects OBJECT IDENTIFIER ::= { wmanIf2MibObjects 1 }
13  wmanIf2SsObjects OBJECT IDENTIFIER ::= { wmanIf2MibObjects 2 }
14  wmanIf2CommonObjects OBJECT IDENTIFIER ::= { wmanIf2MibObjects 3 }
15
16  -- Textual Conventions
17  WmanIf2SfSchedulingType ::= TEXTUAL-CONVENTION
18      STATUS      current
19      DESCRIPTION
20          "The scheduling service provided by a SC for an
21          upstream service flow. If the parameter is omitted
22          from an upstream QOS Parameter Set, this object takes
23          the value of bestEffort (2). This parameter must be
24          reported as undefined (1) for downstream QOS Parameter
25          Sets."
26      SYNTAX      INTEGER {undefined(1),
27                  bestEffort(2),
28                  nonRealTimePollingService(3),
29                  realTimePollingService(4),
30                  reserved(5),
31                  unsolicitedGrantService(6)}
32
33  WmanIf2PhsRuleVerify ::= TEXTUAL-CONVENTION
34      STATUS      current
35      DESCRIPTION
36          "The value of this field indicates to the sending entity
37          whether or not the packet header contents are to be
38          verified prior to performing suppression. If PHSV is
39          enabled, the sender shall compare the bytes in the packet
40          header with the bytes in the PHSF that are to be
41          suppressed as indicated by the PHSM."
42      REFERENCE
43          "Subclause 11.13.19.3.7.5 in IEEE Std 802.16-2004"
44      SYNTAX      INTEGER {phsVerifyEnable(0),
45                          phsVerifyDisable(1)}
46
47  WmanIf2ClassifierBitMap ::= TEXTUAL-CONVENTION
48      STATUS      current
49      DESCRIPTION
50          "A bit of of this object is set to 1 if the parameter
51          indicated by the comment was present in the classifier
52          encoding, and 0 otherwise.
53          Note: that BITS are encoded most significant bit first,
54          so that if e.g. bits 6 and 7 are set, this object is
55

```

```

1         encoded as the octet string '030000'H."
2     REFERENCE
3         "Subclause 11.13.19.3.4 in IEEE Std 802.16-2004"
4     SYNTAX      BITS {priority(0),
5                  ipTos(1),
6                  ipProtocol(2),
7                  ipMaskedSrcAddr(3),
8                  ipMaskedDestAddr(4),
9                  srcPort(5),
10                 destPort(6),
11                 destMacAddr(7),
12                 srcMacAddr(8),
13                 ethernetProtocol(9),
14                 userPriority(10),
15                 vlanId(11),
16                 ipv6FlowLabel(12)}
17
18 WmanIf2SfState ::= TEXTUAL-CONVENTION
19     STATUS      current
20     DESCRIPTION
21         "Defines the state of a service flow."
22     SYNTAX      INTEGER {authorized(1),
23                  admitted(2),
24                  active(3)}
25
26 WmanIf2ServClassName ::= TEXTUAL-CONVENTION
27     STATUS      current
28     DESCRIPTION
29         "Defines the type of service class name."
30     SYNTAX      OCTET STRING (SIZE(2..128))
31
32 WmanIf2CsSpecification ::= TEXTUAL-CONVENTION
33     STATUS      current
34     DESCRIPTION
35         "Defines the types of convergence sublayer."
36     REFERENCE
37         "Subclause 11.13.19.1 in IEEE Std 802.16e-2005"
38     SYNTAX      INTEGER {reserved(0),
39                  packetIPv4(1),
40                  packetIPv6(2),
41                  packet802dot3Ethernet(3),
42                  packet802dot1QVlan(4),
43                  packetIPv4Over802dot3(5),
44                  packetIPv6Over802dot3(6),
45                  packetIPv4Over802dot1Q(7),
46                  packetIPv6Over802dot1Q(8),
47                  atm(9),
48                  packet802dot3EthernetRohcHc(10),
49                  packet802dot3EthernetEcrtphc(11),
50                  packetIp2RohcHc(12),
51                  packetIp2Ecrtphc(13)}
52
53 WmanIf2MacVersion ::= TEXTUAL-CONVENTION
54     STATUS      current

```

```

1      DESCRIPTION
2          "Version number of IEEE 802.16."
3      SYNTAX      INTEGER {ieee802Dot16Of2001(1),
4                      ieee802Dot16cOf2002(2),
5                      ieee802Dot16aOf2003(3),
6                      ieee802Dot16Of2004(4),
7                      ieee802Dot16e(5),
8                      tbd(6)}
9
10
11
12      WmanIf2CidType ::= TEXTUAL-CONVENTION
13          STATUS      current
14          DESCRIPTION
15              "Type of CID."
16          SYNTAX      INTEGER (0 .. 65535)
17
18
19      WmanIf2DataEncryptAlgId ::= TEXTUAL-CONVENTION
20          STATUS      current
21          DESCRIPTION
22              "Data encryption algorithm identifiers."
23          REFERENCE
24              "Table 375 in IEEE Std 802.16-2004"
25          SYNTAX      INTEGER {none(0),
26                      des56BitCbcMode(1),
27                      aesCcmMode(2)}
28
29
30
31      WmanIf2DataAuthAlgId ::= TEXTUAL-CONVENTION
32          STATUS      current
33          DESCRIPTION
34              "Data authentication algorithm identifiers."
35          REFERENCE
36              "Table 376 in IEEE Std 802.16-2004"
37          SYNTAX      INTEGER {noDataAuthentication(0),
38                      reserved(1)}
39
40
41
42      WmanIf2TekEncryptAlgId ::= TEXTUAL-CONVENTION
43          STATUS      current
44          DESCRIPTION
45              "TEK encryption algorithm identifiers."
46          REFERENCE
47              "Table 377 in IEEE Std 802.16-2004"
48          SYNTAX      INTEGER {tripleDes128BitKey(1),
49                      rsa1024BitKey(2),
50                      aes128BitKey(3)}
51
52
53
54      WmanIf2ChannelNumber ::= TEXTUAL-CONVENTION
55          STATUS      current
56          DESCRIPTION
57              "Physical channel number"
58          SYNTAX      INTEGER (0 .. 199)
59
60
61
62      WmanIf2OfdmFecCodeType ::= TEXTUAL-CONVENTION
63          STATUS      current
64          DESCRIPTION
65              "FEC code type and modulation type"

```

REFERENCE

"Table 356 and Table 362 in IEEE Std 802.16-2004"

SYNTAX INTEGER {bpskCc1Over2(0),
 qpskRsCcCc1Over2(1),
 qpskRsCcCc3Over4(2),
 sixteenQamRsCcCc1Over2(3),
 sixteenQamRsCcCc3Over4(4),
 sixtyFourQamRsCcCc2Over3(5),
 sixtyFourQamRsCcCc3Over4(6),
 qpskBtc1Over2(7),
 qpskBtc3Over4(8),
 sixteenQamBtc3Over4(9),
 sixteenQamBtc4Over5(10),
 sixtyFourQamBtc2Over3(11),
 sixtyFourQamBtc5Over6(12),
 qpskCtc1Over2(13),
 qpskCtc2Over3(14),
 qpskCtc3Over4(15),
 sixteenQamCtc1Over2(16),
 sixteenQamCtc3Over4(17),
 sixtyFourQamCtc2Over3(18),
 sixtyFourQamCtc3Over4(19)}

WmanIf2OfdmaFecCodeType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"FEC code type and modulation type"

REFERENCE

"Table 356 and Table 362 in IEEE Std 802.16-2004"

SYNTAX INTEGER {qpskCc1Over2(0),
 qpskCc3Over4(1),
 sixteenQamCc1Over2(2),
 sixteenQamCc3Over4(3),
 sixtyFourQamCc2Over3(4),
 sixtyFourQamCc3Over4(5),
 qpskBtc1Over2(6),
 qpskBtc2Over3(7),
 sixteenQamBtc3Over5(8),
 sixteenQamBtc4Over5(9),
 sixtyFourQamBtc5Over8(10),
 sixtyFourQamBtc4Over5(11),
 qpskCtc1Over2(12),
 qpskCtc2Over3(13),
 qpskCtc3Over4(14),
 sixteenQamCtc1Over2(15),
 sixteenQamCtc3Over4(16),
 sixtyFourQamCtc2Over3(17),
 sixtyFourQamCtc3Over4(18),
 sixtyFourQamCtc5Over6(19),
 qpskZtCc1Over2(20),
 qpskZtCc3Over4(21),
 sixteenQamZtCc1Over2(22),
 sixteenQamZtCc3Over4(23),
 sixtyFourQamZtCc2Over3(24),

```

1          sixtyFourQamZtCc3Over4(25) }
2
3
4  -- Textual convention for capabilities encodings
5  WmanIf2eNumOfCid ::= TEXTUAL-CONVENTION
6      STATUS          current
7      DESCRIPTION
8          "The object of this type shows the number of CIDs that
9           SS can support."
10
11      REFERENCE
12          "Subclause 11.7.6 in IEEE Std 802.16e-2005"
13      SYNTAX          INTEGER (2..65535)
14
15  WmanIf2ArqSupportType ::= TEXTUAL-CONVENTION
16      STATUS          current
17      DESCRIPTION
18          "The object of this type indicates whether the SS support
19           ARQ."
20
21      REFERENCE
22          "Subclause 11.7.8.1 in IEEE Std 802.16-2004"
23      SYNTAX          INTEGER {arqNotSupported(0),
24                               arqSupported(1)}
25
26
27  WmanIf2MaxDsxFLOWType ::= TEXTUAL-CONVENTION
28      STATUS          current
29      DESCRIPTION
30          "The object of this type specifies the maximum number of
31           concurrent DSA, DSC, or DSD transactions that may be
32           outstanding."
33
34      REFERENCE
35          "Subclause 11.7.8.2 in IEEE Std 802.16-2004"
36      SYNTAX          INTEGER (0..255)
37
38
39  WmanIf2MacCrcSupport ::= TEXTUAL-CONVENTION
40      STATUS          current
41      DESCRIPTION
42          "The object of this type indicates whether or not the SS
43           supports MAC level CRC."
44
45      REFERENCE
46          "Subclause 11.7.8.3 in IEEE Std 802.16-2004"
47      SYNTAX          INTEGER {noMacCrcSupport(0),
48                               macCrcSupport(1)}
49
50
51  WmanIf2MaxMcaFlowType ::= TEXTUAL-CONVENTION
52      STATUS          current
53      DESCRIPTION
54          "The object of this type specifies the maximum number of
55           concurrent MCA transactions that may be outstanding."
56
57      REFERENCE
58          "Subclause 11.7.8.4 in IEEE Std 802.16-2004"
59      SYNTAX          INTEGER (0..255)
60
61
62  WmanIf2MaxMcpGroupCid ::= TEXTUAL-CONVENTION
63      STATUS          current
64      DESCRIPTION
65

```

```

1         "The object of this type indicates the maximum number of
2         simultaneous Multicast Polling Groups the SS is
3         capable of belonging to."
4
5     REFERENCE
6         "Subclause 11.7.8.5 in IEEE Std 802.16-2004"
7     SYNTAX      INTEGER (0..255)
8
9
10    WmanIf2MaxPkmFlowType ::= TEXTUAL-CONVENTION
11        STATUS      current
12        DESCRIPTION
13            "The object of this type specifies the maximum number of
14            concurrent PKM transactions that may be outstanding."
15        REFERENCE
16            "Subclause 11.7.8.6 in IEEE Std 802.16-2004"
17        SYNTAX      INTEGER (0..255)
18
19
20    WmanIf2AuthPolicyType ::= TEXTUAL-CONVENTION
21        STATUS      current
22        DESCRIPTION
23            "The object of this type specifies authorization policy
24            that both SS and BS need to negotiate and synchronize.
25            A bit value of 0 = 'not supported', 1 = 'supported'. If
26            this field is omitted, then both SS and BS shall use the
27            IEEE 802.16 security, constituting X.509 digital
28            certificates and the RSA public key encryption
29            algorithm, as authorization policy."
30        REFERENCE
31            "Subclause 11.7.8.7 in IEEE Std 802.16-2004"
32        SYNTAX      BITS {ieee802Dot16PrivacySupported(0),
33                          reserved1(1),
34                          reserved2(2),
35                          reserved3(3),
36                          reserved4(4),
37                          reserved5(5),
38                          reserved6(6),
39                          reserved7(7)}
40
41
42    WmanIf2MaxNumOfSaType ::= TEXTUAL-CONVENTION
43        STATUS      current
44        DESCRIPTION
45            "This field specifies maximum number of supported
46            security association of the SS."
47        REFERENCE
48            "Subclause 11.7.8.8 in IEEE Std 802.16-2004"
49        SYNTAX      INTEGER (0..255)
50
51
52    WmanIf2IpVersionType ::= TEXTUAL-CONVENTION
53        STATUS      current
54        DESCRIPTION
55            "The object of this type indicates the version of IP used
56            on the Secondary Management Connection. The value should
57            be undefined if the 2nd management CID doesn't exist."
58        REFERENCE
59            "Subclause 11.7.4 in IEEE Std 802.16-2004"
60
61
62
63
64
65

```



```

1          SYNTAX          INTEGER {undefined(0),
2                               ipv4(1),
3                               ipv6(2)}
4
5
6 WmanIf2MacCsBitMap ::= TEXTUAL-CONVENTION
7     STATUS          current
8     DESCRIPTION
9         "The object of this type indicates the set of MAC
10        convergence sublayer support. When a bit is set, it
11        indicates the corresponding CS feature is supported."
12
13     REFERENCE
14         "Subclause 11.7.7.1 in IEEE Std 802.16e-2005"
15
16     SYNTAX          BITS {atm(0),
17                           packetIpv4(1),
18                           packetIpv6(2),
19                           packet802Dot3(3),
20                           packet802Dot1Q(4),
21                           packetIpv4Over802Dot3(5),
22                           packetIpv6Over802Dot3(6),
23                           packetIpv4Over802Dot1Q(7),
24                           packetIpv6Over802Dot1Q(8),
25                           packet802dot3EthernetRohcHc(9),
26                           packet802dot3EthernetEcrtPc(10),
27                           packetIpv4Orv6RohcHc(11),
28                           packetIpv4Orv6EcrtPc(12)}
29
30
31
32 WmanIf2MaxClassifiers ::= TEXTUAL-CONVENTION
33     STATUS          current
34     DESCRIPTION
35         "The object of this type indicates the maximum number of
36        admitted Classifiers that the SS is allowed to have."
37
38     REFERENCE
39         "Subclause 11.7.7.2 in IEEE Std 802.16-2004"
40
41     SYNTAX          INTEGER (0..65535)
42
43
44 WmanIf2PhsSupportType ::= TEXTUAL-CONVENTION
45     STATUS          current
46     DESCRIPTION
47         "The object of this type indicates the level
48        of PHS support."
49
50     REFERENCE
51         "Subclause 11.7.7.3 in IEEE Std 802.16e-2005"
52
53     SYNTAX          INTEGER {noPhsSupport(0),
54                             atmPhsSupport(1),
55                             packetPhsSupport(2),
56                             atmAndPacketPhsSupport(3)}
57
58
59 WmanIf2BwAllocSupport ::= TEXTUAL-CONVENTION
60     STATUS          current
61     DESCRIPTION
62         "This field indicates properties of the SS that the BS
63        needs to know for bandwidth allocation purposes. When
64        a bit is set, it indicates the corresponding feature
65        is supported. All unspecified and reserved bits should

```

```

1         be set to zero."
2     REFERENCE
3         "Subclause 11.8.1 in IEEE Std 802.16-2004"
4     SYNTAX      BITS {reserved(0),
5                  halfDuplexFdd(1),
6                  fullDuplexFdd(2)}
7
8
9
10    WmanIf2PduConstruction ::= TEXTUAL-CONVENTION
11        STATUS      current
12        DESCRIPTION
13            "Specifies capabilities for construction and transmission
14            of MAC PDUs. When piggybackedRequests bit is set, it
15            indicates that the piggybacked requests are supported. The
16            fsnValuesSize bit is coded as follows:
17                0 - only 3-bit FSN values are supported
18                1 - only 11-bit FSN values are supported
19            All unspecified and reserved bits should be set to zero."
20    REFERENCE
21        "Subclause 11.8.2 in IEEE Std 802.16e-2005"
22    SYNTAX      BITS {piggybackedRequests(0),
23                  fsnValuesSize(1)}
24
25
26
27    WmanIf2SsTransitionGap ::= TEXTUAL-CONVENTION
28        STATUS      current
29        DESCRIPTION
30            "This field indicates the transition speed SSTTG and SSRTG
31            for TDD and H-FDD SSs. Allowed values are:
32                OFDM mode: TDD and H-FDD 0..100
33                Other modes: TDD: 0..50; H-FDD: 0..100"
34    REFERENCE
35        "Subclause 11.8.3.1 in IEEE Std 802.16-2004"
36    SYNTAX      INTEGER (0..100)
37
38
39
40    WmanIf2MaxTxPowerType ::= TEXTUAL-CONVENTION
41        STATUS      current
42        DESCRIPTION
43            "This type is used to define maximum available power for
44            BPSK, QPSK, 16-QAM and 64-QAM constellations. The maximum
45            power parameters are reported in dBm and quantized in 0.5
46            dBm steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
47            (encoded 0xFF). Values outside this range shall be
48            assigned the closest extreme. SSs that do not support
49            QAM64 shall report the value of 0x00 in the maximum QAM64
50            power field."
51    REFERENCE
52        "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
53    SYNTAX      INTEGER (0..255)
54
55
56
57    WmanIf2OfdmFftSizes ::= TEXTUAL-CONVENTION
58        STATUS      current
59        DESCRIPTION
60            "This field indicates the FFT sizes supported by the SS.
61            For each FFT size, a bit value of 0 indicates 'not
62            supported' while 1 indicates 'supported'."
63
64
65

```

```

1      REFERENCE
2          "Subclause 11.8.3.6.1 in IEEE 802.16-2004"
3      SYNTAX      BITS {fft256(0),
4                      fft2048(1)}
5
6
7      WmanIf2OfdmSsDeModType ::= TEXTUAL-CONVENTION
8          STATUS      current
9          DESCRIPTION
10             "This field indicates the different demodulator options
11             supported by a WirelessMAN-OFDM PHY SS for downlink. This
12             field is not used for other PHY specifications. A bit
13             value of 0 indicates 'not supported' while 1 indicates
14             'supported'."
15
16      REFERENCE
17          "Subclause 11.8.3.6.2 in IEEE Std 802.16e-2005"
18      SYNTAX      BITS {qam64(0),
19                      btc(1),
20                      ctc(2),
21                      stc(3),
22                      aas(4),
23                      subchannelization(5)}
24
25
26
27      WmanIf2OfdmSsModType ::= TEXTUAL-CONVENTION
28          STATUS      current
29          DESCRIPTION
30             "This field indicates the different modulator options
31             supported by a WirelessMAN-OFDM PHY SS for uplink. This
32             field is not used for other PHY specifications. A bit
33             value of 0 indicates 'not supported' while 1 indicates
34             'supported'."
35
36      REFERENCE
37          "Subclause 11.8.3.6.3 in IEEE Std 802.16e-2005"
38      SYNTAX      BITS {qam64(0),
39                      btc(1),
40                      ctc(2),
41                      subchanellization(3),
42                      focusedCtBwReq(4),
43                      ulCyclicDelay(5)}
44
45
46
47
48      WmanIf2OfdmFocusedCt ::= TEXTUAL-CONVENTION
49          STATUS      current
50          DESCRIPTION
51             "This field indicates whether the SS supports Focused
52             Contention (see 8.3.7.3.3). A bit value of 0 indicates
53             'not supported' while 1 indicates 'supported'."
54
55      REFERENCE
56          "Subclause 11.8.3.6.4 in IEEE Std 802.16-2004"
57      SYNTAX      BITS {focusedCtSupport(0)}
58
59
60      WmanIf2OfdmTcSublayer ::= TEXTUAL-CONVENTION
61          STATUS      current
62          DESCRIPTION
63             "This field indicates whether or not the SS supports the
64             TC sublayer (see 8.3.4). A bit value of 0 indicates
65

```

```

1         'not supported' while 1 indicates 'supported'."
2     REFERENCE
3         "Subclause 11.8.3.6.5 in IEEE Std 802.16-2004"
4     SYNTAX      BITS {tcSublayerSupport(0)}
5
6
7 WmanIf2eOfdmPrivMap ::= TEXTUAL-CONVENTION
8     STATUS      current
9     DESCRIPTION
10        "This field indicates if the private map parameters
11        is supported. A bit value of 0 indicates
12        'not supported' while 1 indicates 'supported'."
13    REFERENCE
14        "Subclause 11.8.3.6.6 in IEEE Std 802.16e-2005"
15    SYNTAX      BITS {regularMap(0),
16                  compressedMap(1)}
17
18
19
20 WmanIf2eOfdmULPower ::= TEXTUAL-CONVENTION
21     STATUS      current
22     DESCRIPTION
23        "This field indicates the uplink power control options
24        supported by a WirelessMAN-OFDM PHY SS for uplink
25        transmission. A bit value of 0 indicates
26        'not supported' while 1 indicates 'supported'."
27    REFERENCE
28        "Subclause 11.8.3.7.10 in IEEE Std 802.16e-2005"
29    SYNTAX      BITS {ulOpenLoopPwrCntl(0),
30                  ulAasPreamblePwrCntl(1)}
31
32
33
34
35 WmanIf2BsIdType ::= TEXTUAL-CONVENTION
36     STATUS      current
37     DESCRIPTION
38        "Defines the encoding of BSID. The BSID is a 6 byte number
39        and follows the encoding rules of MacAddress textual
40        convention, i.e. as if it were transmitted
41        least-significant bit first. The value should be displayed
42        with 2 parts clearly separated by a colon e.g:
43        001DFF:00003A. The most significant part is representing
44        the Operator ID. "
45    SYNTAX      OCTET STRING (SIZE(6))
46
47
48
49 WmanIf2Ipv6FlowLabel ::= TEXTUAL-CONVENTION
50     STATUS      current
51     DESCRIPTION
52        "The value of this field specifies the matching values for
53        the IPv6 Flow label field. As the flow label field has a
54        length of 20 bits, the first 4 bits of the most
55        significant byte shall be set to 0x0 and disregarded."
56    SYNTAX      OCTET STRING (SIZE(3))
57
58
59
60 WmanIf2OfdmaFftSizes ::= TEXTUAL-CONVENTION
61     STATUS      current
62     DESCRIPTION
63        "This field indicates the FFT sizes supported by the SS/MS.
64        For each FFT size, a bit value of 0 indicates 'not
65

```

```

1         supported' while 1 indicates 'supported'."
2     REFERENCE
3         "Subclause 11.8.3.7.1 in IEEE 802.16e-2005"
4     SYNTAX      BITS {fft256(0),
5                  fft2048(1),
6                  fft128(2),
7                  fft512(3),
8                  fft1024(4)}
9
10
11
12 WmanIf2OfdmaMsDeModType ::= TEXTUAL-CONVENTION
13     STATUS      current
14     DESCRIPTION
15         "This field indicates the different demodulator options
16         supported by a WirelessMAN-OFDMA PHY SS for downlink.
17         A bit value of 0 indicates 'not supported' while 1
18         indicates 'supported'."
19     REFERENCE
20         "Subclause 11.8.3.7.2 in IEEE Std 802.16e-2005"
21     SYNTAX      BITS {qam64(0),
22                  btc(1),
23                  ctc(2),
24                  stc(3),
25                  ccWithInterleacer(4),
26                  harqChase(5),
27                  harqCtcIr(6),
28                  reserved(7),
29                  harqCcIr(8),
30                  ldpc(9),
31                  dedicatedPilots(10)}
32
33
34
35
36
37 WmanIf2OfdmaMsModType ::= TEXTUAL-CONVENTION
38     STATUS      current
39     DESCRIPTION
40         "This field indicates the different modulator options
41         supported by a WirelessMAN-OFDMA PHY SS for uplink. A bit
42         value of 0 indicates 'not supported' while 1 indicates
43         'supported'."
44     REFERENCE
45         "Subclause 11.8.3.7.3 in IEEE Std 802.16e-2005"
46     SYNTAX      BITS {qam64(0),
47                  btc(1),
48                  ctc(2),
49                  stc(3),
50                  harqChase(4),
51                  ctcIr(5),
52                  ccIr(6),
53                  ldpc(7)}
54
55
56
57
58
59 WmanIf2OfdmaPermutation ::= TEXTUAL-CONVENTION
60     STATUS      current
61     DESCRIPTION
62         "This field indicates the OFDMA SS Permutation support
63         A bit value of 0 indicates 'not supported' while 1
64         indicates 'supported'."
65

```

REFERENCE

"Subclause 11.8.3.7.4 in IEEE 802.16e"

SYNTAX BITS {optionalPuscSupport(0),
 optionalFuscSupport(1),
 amcOneBySixSupport(2),
 amcTwoByThreeSupport(3),
 amcThreeByTwoSupport(4),
 amcSupportWithHarqMap(5),
 tusc1Support(6),
 tusc2Support(7)}

WmanIf2eOfdmaDemMimo ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"This field indicates the MIMO capability of OFDMA MS demodulator. A bit value of 0 indicates 'not supported' while 1 indicates 'supported'."

REFERENCE

"Subclause 11.8.3.7.5 in IEEE 802.16e"

SYNTAX BITS {twoAntStcMatrixA(0),
 twoAntStcMatrixBVCoding(1),
 twoAntStcMatrixBHCoding(2),
 fourAntStcMatrixA(3),
 fourAntStcMatrixBVCoding(4),
 fourAntStcMatrixBHCoding(5),
 fourAntStcMatrixCVCoding(6),
 fourAntStcMatrixCHCodingt(7)}

WmanIf2eOfdmaMimoCap ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"This field indicates the MIMO capability of OFDMA MS demodulator."

REFERENCE

"Subclause 11.8.3.7.5 in IEEE 802.16e"

SYNTAX BITS {twoAntStcMatrixA(0),
 twoAntStcMatrixBVCoding(1),
 fourRxAntenna(2),
 fourAntStcMatrixA(3),
 fourAntStcMatrixBVCoding(4),
 fourAntStcMatrixBHCoding(5),
 fourAntStcMatrixCVCoding(6),
 fourAntStcMatrixCHCodingt(7),
 threeAntStcMatrixA(8),
 threeAntStcMatrixB(9),
 threeAntStcMatrixCVCoding(10),
 threeAntStcMatrixCHCodingt(11),
 calculatingPrecodingWeight(12),
 adaptiveRateControl(13),
 calculatingChannelMatrix(14),
 antennaGrouping(15),
 antennaSelection(16),
 codebookBasedPrecoding(17),
 longTermPrecoding(18),

```

1           mimoMidamble(19) }
2
3 WmanIf2eOfdmaUlMimo ::= TEXTUAL-CONVENTION
4     STATUS          current
5     DESCRIPTION
6         "This field indicates the different MIMO options supported
7         by a WirelessMAN-OFDMA PHY SS in the uplink.
8         A bit value of 0 indicates 'not supported' while 1
9         indicates 'supported'."
10
11     REFERENCE
12         "Subclause 11.8.3.7.6 in IEEE 802.16e"
13
14     SYNTAX          BITS {twoAntSttd(0),
15                          twoAntSmVCoding(1),
16                          oneAntCooperativeSm(2) }
17
18 WmanIf2eOfdmaPrivMap ::= TEXTUAL-CONVENTION
19     STATUS          current
20     DESCRIPTION
21         "This field indicates the AAS private map parameters
22         supported by a WirelessMAN-OFDMA SS. A bit value of
23         0 indicates 'not supported' while 1 indicates
24         'supported' for most bits, except chainConcurrency0,
25         chainConcurrency1 that indicates how many parallel
26         private map chains can be supported by an SS.
27         0:      no limit
28         1..3: maximum concurrent private map chains"
29
30     REFERENCE
31         "Subclause 11.8.3.7.7 in IEEE Std 802.16e-2005"
32
33     SYNTAX          BITS {harqMap(0),
34                          privMap(1),
35                          reducedPrivMap(2),
36                          privMapChainEnable(3),
37                          privMapDlFrameOffset(4),
38                          privMapUlFrameOffset(5),
39                          chainConcurrency0(6),
40                          chainConcurrency1(7) }
41
42 WmanIf2eOfdmaAasCap ::= TEXTUAL-CONVENTION
43     STATUS          current
44     DESCRIPTION
45         "This field indicates the different AAS options
46         supported by a WirelessMAN-OFDMA PHY SS in the
47         downlink. A bit value of 0 indicates 'not supported'
48         while 1 indicates 'supported' for most bits."
49
50     REFERENCE
51         "Subclause 11.8.3.7.8 in IEEE Std 802.16e-2005"
52
53     SYNTAX          BITS {aasZone(0),
54                          aasDiversityMapScan(1),
55                          aasFbckRsp(2),
56                          dlAasPreamble(3),
57                          ulAasPreamble(4) }
58
59 WmanIf2eOfdmaCinrCap ::= TEXTUAL-CONVENTION
60     STATUS          current
61
62
63
64
65

```

```

1      DESCRIPTION
2          "This field indicates the CINR measurement capability
3          supported by a WirelessMAN-OFDMA PHY SS in the
4          downlink. A bit value of 0 indicates 'not supported'
5          while 1 indicates 'supported'."
6      REFERENCE
7          "Subclause 11.8.3.7.9 in IEEE Std 802.16e-2005"
8      SYNTAX      BITS {phyCinrPreamble(0),
9                  phyCinrPilotSubc(1),
10                 phyCinrDataSubc(2),
11                 effectiveCinrPreamble(3),
12                 effectiveCinrPilotSubc(4),
13                 effectiveCinrDataSubc(5),
14                 twoCqiChannel(6),
15                 freqSelectivityReport(7)}
16
17 WmanIf2eOfdmaUlPower ::= TEXTUAL-CONVENTION
18     STATUS      current
19     DESCRIPTION
20         "This field indicates the power control options
21         supported by a WirelessMAN-OFDMA PHY SS for uplink
22         transmission. A bit value of 0 indicates
23         'not supported' while 1 indicates 'supported'."
24     REFERENCE
25         "Subclause 11.8.3.7.11 in IEEE Std 802.16e-2005"
26     SYNTAX      BITS {ulOpenLoopPwrCntl(0),
27                 ulAasPreamblePwrCntl(1)}
28
29 WmanIf2eOfdmaMapCap ::= TEXTUAL-CONVENTION
30     STATUS      current
31     DESCRIPTION
32         "This field indicates the different MAP options supported
33         by a WirelessMAN-OFDMA PHY SS. A bit value of 0
34         indicates 'not supported' while 1 indicates 'supported'."
35     REFERENCE
36         "Subclause 11.8.3.7.12 in IEEE Std 802.16e-2005"
37     SYNTAX      BITS {harqMap(0),
38                 extendedHarqIe(1),
39                 subMapFor1stZone(2),
40                 subMapForOtherZone(3),
41                 dlRegionDefinition(4)}
42
43 WmanIf2eOfdmaUlCntlCh ::= TEXTUAL-CONVENTION
44     STATUS      current
45     DESCRIPTION
46         "This field indicates different uplink control channels
47         supported by a WirelessMAN-OFDMA PHY SS. A bit value
48         of 0 indicates 'not supported' while 1 indicates
49         'supported'."
50     REFERENCE
51         "Subclause 11.8.3.7.13 in IEEE Std 802.16e-2005"
52     SYNTAX      BITS {threeBitMimoFastFeedback(0),
53                 enhancedFastFeedback(1),
54                 ulAck(2),

```



```

1           reserved(3),
2           uepFastFeedback(4),
3           fastDlMeasurementFeedback(5),
4           priSecFastFeedback(6),
5           diucCqiFastFeedback(7) }
6
7
8 WmanIf2eOfdmaMsCistCap ::= TEXTUAL-CONVENTION
9     STATUS          current
10    DESCRIPTION
11        "This field indicates MS capability of supporting CSIT
12        (uplink sounding). A bit value of 0 indicates 'not
13        supported' while 1 indicates 'supported'.
14
15        Bits 3..5: Time needed for SS to respond to a sounding
16        command transmitted by the BS
17
18        000    0.5ms
19        001    0.75ms
20        010    1ms
21        011    1.25ms
22        100    1.5ms
23        101    min(2ms, Next Frame)
24        110    min(5ms, Next Frame)
25        111    Next Frame
26
27        Bits 6..9: Max number of simultaneous sounding
28        instructions (0 = unlimited)"
29
30 REFERENCE
31     "Subclause 11.8.3.7.14 in IEEE Std 802.16e-2005"
32
33 SYNTAX      BITS {csitTypeA(0),
34                  csitTypeB(1),
35                  powerAssignment(2),
36                  soundingRspTime0(3),
37                  soundingRspTime1(4),
38                  soundingRspTime2(5),
39                  maxSimuSoundInst0(6),
40                  maxSimuSoundInst1(7),
41                  maxSimuSoundInst2(8),
42                  maxSimuSoundInst3(9),
43                  noP9Or18ForCsitTypeA(10),
44                  csitNotSupported(11) }
45
46
47 WmanIf2eOfdmaMaxHarq ::= TEXTUAL-CONVENTION
48     STATUS          current
49     DESCRIPTION
50         "This field indicates the maximum number of UL/DL HARQ
51         burst allocations for the SS in a single UL/DL subframe.
52
53         Bits 0..2: Maximum number of UL HARQ bursts per HARQ
54         enabled MS per frame
55         0b000 = 1 (default)
56
57         Bit      3: Indicates whether the maximum number of UL
58         HARQ bursts per frame in bits 0-2 includes the
59         one Non-HARQ burst.
60
61
62
63
64
65

```

```

1           0 = not included (default)
2           1 = included
3
4           Bits 4..7: Maximum number of DL HARQ bursts per HARQ
5                     enabled MS per frame.
6                     0b0000 = 1 (default)"
7
8 REFERENCE
9     "Subclause 11.8.3.7.15 in IEEE Std 802.16e-2005"
10 SYNTAX     BITS {maxUlHarqBurst0(0),
11                maxUlHarqBurst1(1),
12                maxUlHarqBurst2(2),
13                nonHarqBurstInUl(3),
14                maxDlHarqBurst0(4),
15                maxDlHarqBurst1(5),
16                maxDlHarqBurst2(6),
17                maxDlHarqBurst3(7)}
18
19
20
21 WmanIf2eOfdmaModMimo ::= TEXTUAL-CONVENTION
22     STATUS         current
23     DESCRIPTION
24         "This field indicates the MIMO capability of OFDMA SS
25         modulator. A bit value of 0 indicates 'not supported'
26         while 1 indicates 'supported'"
27     REFERENCE
28         "Subclause 11.8.3.7.16 in IEEE Std 802.16e-2005"
29 SYNTAX     BITS {twoTxAntenna(0),
30                txDiversity(1),
31                spatialMultiplexing(2),
32                beamforming(3),
33                adaptiveRateControl(4),
34                singleAntenna(5),
35                twoAntenna(6)}
36
37
38
39
40 WmanIf2eSdmaPilotCap ::= TEXTUAL-CONVENTION
41     STATUS         current
42     DESCRIPTION
43         "This field indicates SDMA pilot pattern support for
44         AMC zone."
45     REFERENCE
46         "Subclause 11.8.3.7.17 in IEEE Std 802.16e-2005"
47 SYNTAX     INTEGER {noSupport(0),
48                sdmaPilotAandB(1),
49                allSdmaPilotPatterns(2)}
50
51
52
53 WmanIf2eMultiBurst ::= TEXTUAL-CONVENTION
54     STATUS         current
55     DESCRIPTION
56         "This field indicates whether multiple FEC types are
57         supported in DL/UL burst profiles. A bit value of 0
58         indicates 'not supported' while 1 indicates
59         'supported'"
60     REFERENCE
61         "Subclause 11.8.3.7.18 in IEEE Std 802.16e-2005"
62 SYNTAX     INTEGER {dlWithMultiFecType(0),
63

```

```

1                                     ulWithMultiFecType(1) }
2
3
4 WmanIf2eIncrHarqBuf ::= TEXTUAL-CONVENTION
5     STATUS          current
6     DESCRIPTION
7         "This field indicates the maximal number of data
8         bits the SS is able to use for buffering for NEP/NSCH
9         based incremental redundancy CTC in downlink and uplink
10        transmissions.
11
12        Bits 0..3: NEP value indicating downlink HARQ buffering
13        capability for incremental redundancy CTC
14
15        Bit      4: Aggregation Flag for DL
16                0 = the number of bits is counted separately
17                for each channel
18                1 = buffering capability may be shared between
19                channels
20
21        Bits 5..7: reserved
22
23        Bits 8..11: NEP value indicating uplink HARQ buffering
24        capability for incremental redundancy CTC
25
26        Bit      12: Aggregation Flag for UL
27                0 = the number of bits is counted separately
28                for each channel
29                1 = buffering capability may be shared between
30                channels"
31
32 REFERENCE
33     "Subclause 11.8.3.7.19 in IEEE Std 802.16e-2005"
34
35 SYNTAX          INTEGER {dlNep0(0),
36                        dlNep1(1),
37                        dlNep2(2),
38                        dlNep3(3),
39                        dlAggFlag(4),
40                        reserved0(5),
41                        reserved1(6),
42                        reserved2(7),
43                        ulNep0(8),
44                        ulNep1(9),
45                        ulNep2(10),
46                        ulNep3(11),
47                        ulAggFlag(12) }
48
49
50 WmanIf2eChaseHarqBuf ::= TEXTUAL-CONVENTION
51     STATUS          current
52     DESCRIPTION
53         "This field indicates the maximal number of data
54         bits the SS is able to use for buffering for
55         DIUC/duration based HARQ methods (Chase combining and
56         CC-IR) in downlink and uplink transmissions.
57
58        Bits 0..5: Downlink HARQ buffering capability for
59
60
61
62
63
64
65

```

```

1           chase combining (K)
2
3           Bit      6: Aggregation Flag for DL
4                   0 = the number of bits is counted separately
5                   for each channel
6                   1 = buffering capability may be shared between
7                   channels
8
9
10          Bits      7: reserved
11
12
13          Bits 8..13: Uplink HARQ buffering capability for chase
14                    combining (K)
15
16          Bit      14: Aggregation Flag for UL
17                   0 = the number of bits is counted separately
18                   for each channel
19                   1 = buffering capability may be shared between
20                   channels"
21
22  REFERENCE
23      "Subclause 11.8.3.7.19 in IEEE Std 802.16e-2005"
24
25  SYNTAX      INTEGER {dlChaseComb0(0),
26                  dlChaseComb1(1),
27                  dlChaseComb2(2),
28                  dlChaseComb3(3),
29                  dlChaseComb4(4),
30                  dlChaseComb5(5),
31                  dlAggFlag(6),
32                  reserved(7),
33                  ulChaseComb0(8),
34                  ulChaseComb1(9),
35                  ulChaseComb2(10),
36                  ulChaseComb3(11),
37                  ulChaseComb4(12),
38                  ulChaseComb5(13),
39                  ulAggFlag(14)}
40
41
42
43
44  WmanIf2eOfdmaMobility ::= TEXTUAL-CONVENTION
45      STATUS      current
46      DESCRIPTION
47          "This field indicates whether or not the MS supports
48          mobility hand-over, Sleepmode, and Idle-mode. A bit
49          value of 0 indicates 'not supported' while 1 indicates
50          it is supported."
51
52  REFERENCE
53      "Subclause 11.8.3.7.5 in IEEE 802.16e"
54
55  SYNTAX      BITS {handoverSupport(0),
56                  sleepModeSupport(1),
57                  idleModeSupport(2)}
58
59
60  WmanIf2PsClassId ::= TEXTUAL-CONVENTION
61      STATUS      current
62      DESCRIPTION
63          "Indicates the index to Power Saving Classes. The ID shall
64          be unique within the group of Power Saving Classes
65

```

```

1         associated with the MS. This ID may be used in further
2         MOB_SLP-REQ/RSP messages for activation / deactivation
3         of Power Saving Class."
4
5     REFERENCE
6         "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
7     SYNTAX         INTEGER (0..63)
8
9
10    WmanIf2ePsClassType ::= TEXTUAL-CONVENTION
11        STATUS         current
12        DESCRIPTION
13            "The types of power saving classes."
14        REFERENCE
15            "Table 374a in IEEE Std 802.16e-2005"
16        SYNTAX         INTEGER {powerSavingClassTypeI(1),
17                                powerSavingClassTypeII(2),
18                                powerSavingClassTypeIII(3)}
19
20
21    WmanIf2ePsClassCidDir ::= TEXTUAL-CONVENTION
22        STATUS         current
23        DESCRIPTION
24            "The direction of power saving class's CIDs.
25             0b00 = Unspecified. Each CID has its own direction
26             assign in its connection creation. Can be
27             DL, UL, or both (in the case of management
28             connections).
29             0b01 = Downlink direction only.
30             0b10 = Uplink direction only."
31        REFERENCE
32            "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
33        SYNTAX         INTEGER {unspecified(0),
34                                downlink(1),
35                                uplink(2)}
36
37
38    WmanIf2PowerSavingMode ::= TEXTUAL-CONVENTION
39        STATUS         current
40        DESCRIPTION
41            "Power saving class mode active or not active."
42        REFERENCE
43            "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
44        SYNTAX         INTEGER {psNotActive(0),
45                                psActive(1)}
46
47
48    WmanIf2eMaxMacLevel ::= TEXTUAL-CONVENTION
49        STATUS         current
50        DESCRIPTION
51            "maximum amount of MAC level data including MAC headers
52            and HARQ retransmission bursts the MS is capable of
53            processing in the DL/UL part of a single MAC frame."
54        REFERENCE
55            "Subclause 11.7.8.10 in IEEE Std 802.16e-2005"
56        SYNTAX         INTEGER (0..65535)
57
58
59    WmanIf2ePackingSupport ::= TEXTUAL-CONVENTION
60        STATUS         current
61

```

```

1      DESCRIPTION
2          "Indicates the availability of MS support for Packing"
3      REFERENCE
4          "Subclause 11.7.8.11 in IEEE Std 802.16e-2005"
5      SYNTAX      INTEGER {noPackingSupport(0),
6                      packingSupported(1)}
7
8
9      WmanIf2eExtRtpsSupport ::= TEXTUAL-CONVENTION
10         STATUS      current
11         DESCRIPTION
12             "Indicates the availability of MS support for Extended
13             rtPS."
14         REFERENCE
15             "Subclause 11.7.8.12 in IEEE Std 802.16e-2005"
16         SYNTAX      INTEGER {noExtendedRtpsSupport(0),
17                             extendedRtpsSupported(1)}
18
19
20
21      WmanIf2eIpAllocMethod ::= TEXTUAL-CONVENTION
22         STATUS      current
23         DESCRIPTION
24             "Indicates the method of allocating IP address for the
25             secondary management connection. A bit value of 0
26             indicates 'not supported' while 1 indicates 'supported'."
27         REFERENCE
28             "Subclause 11.7.8.11 in IEEE Std 802.16e-2005"
29         SYNTAX      BITS {dhcp(0),
30                             mobileIpv4(1),
31                             dhcpV6(2),
32                             ipv6Autoconfig(3)}
33
34
35
36
37      WmanIf2eHandoverType ::= TEXTUAL-CONVENTION
38         STATUS      current
39         DESCRIPTION
40             "Indicates what type(s) of Handover the BS and the MS
41             supports.
42             bit#0: when it is set to 1, MDHO/FBSS HO not supported.
43                     the BS shall ignore all other bits.
44             bit#1: when it is set to 1, FBSS/MDHO DL RF Combining
45                     is supported with monitoring MAPs from active BSs
46             bit#2: when it is set to 1, MDHO DL soft Combining is
47                     supported with monitoring single MAP from
48                     anchor BS
49             bit#3: when it is set to 1, MDHO DL soft combining is
50                     supported with monitoring MAPs from active BSs
51             bit#3: when it is set to 1, MDHO UL Multiple
52                     transmission is supported"
53         REFERENCE
54             "Subclause 11.7.8.12 in IEEE Std 802.16e-2005"
55         SYNTAX      BITS {mdhcFbssHoNotSupported(0),
56                             mdhcFbssDlMapsFromActiveBss(1),
57                             mdhcDlMapFromAnchorBs(2),
58                             mdhcDlMapsFromActiveBss(3),
59                             mdhcUlMultipleTx(4)}
60
61
62
63
64
65

```

```

1  WmanIf2eArqAckType ::= TEXTUAL-CONVENTION
2      STATUS          current
3      DESCRIPTION
4          "Specifies the ARQ ACK type supported by the MS."
5      REFERENCE
6          "Subclause 11.7.23 in IEEE Std 802.16e-2005"
7      SYNTAX          BITS {selectiveAck(0),
8                          cumulativeAck(1),
9                          cumWithSelAck(2),
10                         cumWithBlockSeqAck(3)}
11
12
13
14  WmanIf2eMacHeaderSupp ::= TEXTUAL-CONVENTION
15      STATUS          current
16      DESCRIPTION
17          "Indicates whether or not the MS and BS support various
18          types of MAC header and extended subheaders. A bit
19          value of 0 indicates 'not supported', while 1 indicates
20          'supported'."
21
22          Bits 8-10: parameters of SDU_SN extended subheader that
23          represent the period of SDU_SN transmission for
24          connection with ARQ disabled = once every 2^p MAC
25          PDUs."
26
27      REFERENCE
28          "Subclause 11.7.25 in IEEE Std 802.16e-2005"
29      SYNTAX          BITS {bwReqUlTxPowerReport(0),
30                          bwReqCinrReport(1),
31                          cqichAlloationReq(2),
32                          phyChannelReport(3),
33                          bwReqUlSleepCntl(4),
34                          snReport(5),
35                          feedbackReport(6),
36                          sduSn(7),
37                          sdnSnPeriod0(8),
38                          sdnSnPeriod1(9),
39                          sdnSnPeriod2(10),
40                          dlSleepControl(11),
41                          feedbackRequest(12),
42                          mimcModeFeedback(13),
43                          ulTxPowerReport(14),
44                          miniFeedback(15),
45                          snRequest(16),
46                          shortPduSn(17),
47                          longPduSn(18)}
48
49
50
51
52
53
54
55  WmanIf2eHarqAckDelay ::= TEXTUAL-CONVENTION
56      STATUS          current
57      DESCRIPTION
58          "HARQ ACK delay for DL burst
59          1 = one frame offset
60          2 = two frames offset
61          3 = three frames offset"
62
63      REFERENCE
64          "Table 353 in IEEE Std 802.16e-2005"
65

```

```

1          SYNTAX          INTEGER {oneframeoffset(1),
2                                twoframesoffset(2),
3                                threeframesoffset(3)}
4
5
6 WmanIf2eAasBeamSel ::= TEXTUAL-CONVENTION
7     STATUS          current
8     DESCRIPTION
9         "Boolean to indicate whether unsolicited AAS Beam Select
10        messages (see 6.3.2.3.41 in IEEE 802.16e-2005) should be
11        sent by the MS.
12        0: MS should not send AAS Beam Select Messages
13        1: MS may send AAS Beam Select Messages"
14
15     REFERENCE
16         "Table 353 in IEEE Std 802.16e-2005"
17     SYNTAX          INTEGER {notAllowed(0),
18                             allowed(1)}
19
20
21 --
22 -- BS object group - containing tables and objects to be implemented in
23 -- the Base station
24 --
25 --
26 -- wmanIf2BsPacketCs contain the Base Station Packet Convergence
27 -- Sublayer objects
28 --
29
30 wmanIf2BsPacketCs OBJECT IDENTIFIER ::= { wmanIf2BsObjects 1 }
31
32 wmanIf2BsProvisionedSfTable OBJECT-TYPE
33     SYNTAX          SEQUENCE OF WmanIf2BsProvisionedSfEntry
34     MAX-ACCESS      not-accessible
35     STATUS          current
36     DESCRIPTION
37         "This table contains service flow profiles provisioned by
38         NMS. The service flow should be created with SS(s)
39         following instruction given by wmanIf2BsSfState object.
40         1. The QoS parameters of the service flow are provisioned
41         in wmanIf2BsServiceClassTable and referenced by
42         wmanIf2BsServiceClassIndex.
43         2. The classifier rules of the service flow are provisioned
44         in wmanIf2BsClassifierRuleTable, where they refer to SF
45         via wmanIf2BsSfId.
46
47         The MAC addresses of SSs the service flow is created with
48         are provisioned in wmanIf2BsSsProvisionedForSfTable, where
49         they refer to SF via wmanIf2BsSfId."
50
51     REFERENCE
52         "Subclause 6.3.13 and 6.3.14 in IEEE Std 802.16-2004"
53     ::= { wmanIf2BsPacketCs 1 }
54
55 wmanIf2BsProvisionedSfEntry OBJECT-TYPE
56     SYNTAX          WmanIf2BsProvisionedSfEntry
57     MAX-ACCESS      not-accessible
58     STATUS          current
59     DESCRIPTION
60         "This table provides one row for each service flow

```



```

1         provisioned by NMS. The table is indexed by ifIndex and
2         wmanIf2BsSfId. ifIndex is associated with the BS sector."
3     INDEX { ifIndex, wmanIf2BsSfId }
4     ::= { wmanIf2BsProvisionedSfTable 1 }
5
6
7     WmanIf2BsProvisionedSfEntry ::= SEQUENCE {
8         wmanIf2BsSfId                Unsigned32,
9         wmanIf2BsSfDirection          INTEGER,
10        wmanIf2BsServiceClassIndex    INTEGER,
11        wmanIf2BsSfState               WmanIf2SfState,
12        wmanIf2BsSfProvisionedTime     TimeStamp,
13        wmanIf2BsSfCsSpecification     WmanIf2CsSpecification,
14        wmanIf2BsProvisionedSfRowStatus RowStatus}
15
16
17
18     wmanIf2BsSfId OBJECT-TYPE
19         SYNTAX      Unsigned32 (1 .. 4294967295)
20         MAX-ACCESS  not-accessible
21         STATUS      current
22         DESCRIPTION
23             "A 32 bit quantity that uniquely identifies a service flow
24             to both the subscriber station and base station (BS)."
```

```

25     ::= { wmanIf2BsProvisionedSfEntry 1 }
26
27
28
29     wmanIf2BsSfDirection OBJECT-TYPE
30         SYNTAX      INTEGER {downstream(1),
31                             upstream(2)}
32         MAX-ACCESS  read-create
33         STATUS      current
34         DESCRIPTION
35             "An attribute indicating the service flow is downstream or
36             upstream."
```

```

37     ::= { wmanIf2BsProvisionedSfEntry 2 }
38
39
40
41     wmanIf2BsServiceClassIndex OBJECT-TYPE
42         SYNTAX      INTEGER (1..65535)
43         MAX-ACCESS  read-create
44         STATUS      current
45         DESCRIPTION
46             "The index in wmanIf2BsServiceClassTable describing the
47             service class or QoS parameters for such service flow.
48             If no associated entry in wmanIf2BsServiceClassTable
49             exists, this object returns a value of zero."
```

```

50     ::= { wmanIf2BsProvisionedSfEntry 3 }
51
52
53
54     wmanIf2BsSfState OBJECT-TYPE
55         SYNTAX      WmanIf2SfState
56         MAX-ACCESS  read-create
57         STATUS      current
58         DESCRIPTION
59             "wmanIf2BsSfState determines the requested state of a service
60             flow.
61             - authorized state: A service flow is provisioned but
62             not resource is reserved yet
63             - admitted state: service flow has resources reserved.
```

```

1         - active state: has resources committed by the BS (e.g., is
2           actively sending maps containing unsolicited grants for a
3           UGS-based service flow),"
4
5     REFERENCE
6         "Subclause 6.3.14.6, in IEEE Std 802.16-2004"
7     ::= { wmanIf2BsProvisionedSfEntry 4 }
8
9
10    wmanIf2BsSfProvisionedTime OBJECT-TYPE
11        SYNTAX      TimeStamp
12        MAX-ACCESS   read-create
13        STATUS       current
14        DESCRIPTION
15            "Indicates the date and time when the service flow is
16             provisioned."
17        ::= { wmanIf2BsProvisionedSfEntry 5 }
18
19
20    wmanIf2BsSfCsSpecification OBJECT-TYPE
21        SYNTAX      WmanIf2CsSpecification
22        MAX-ACCESS   read-create
23        STATUS       current
24        DESCRIPTION
25            "This parameter specifies the convergence sublayer
26             encapsulation mode."
27        REFERENCE
28            "Subclause 11.13.19.1 in IEEE Std 802.16-2004"
29        ::= { wmanIf2BsProvisionedSfEntry 6 }
30
31
32
33    wmanIf2BsProvisionedSfRowStatus OBJECT-TYPE
34        SYNTAX      RowStatus
35        MAX-ACCESS   read-create
36        STATUS       current
37        DESCRIPTION
38            "This object is used to create a new row or modify or
39             delete an existing row in this table.
40
41
42             If the implementator of this MIB has choosen not
43             to implement 'dynamic assignment' of profiles, this
44             object is not useful and should return noSuchName
45             upon SNMP request."
46        ::= { wmanIf2BsProvisionedSfEntry 7 }
47
48
49
50    wmanIf2BsSsProvisionedForSfTable OBJECT-TYPE
51        SYNTAX      SEQUENCE OF WmanIf2BsSsProvisionedForSfEntry
52        MAX-ACCESS   not-accessible
53        STATUS       current
54        DESCRIPTION
55            "This table maps the MAC addresses of SSs to the service
56             flows provisioned in wmanIf2BsProvisionedSfTable."
57        REFERENCE
58            "Subclause 6.3.14 in IEEE Std 802.16-2004"
59        ::= { wmanIf2BsPacketCs 2 }
60
61
62
63    wmanIf2BsSsProvisionedForSfEntry OBJECT-TYPE
64        SYNTAX      WmanIf2BsSsProvisionedForSfEntry
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table is indexed by wmanIf2BsSsProvMacAddress and
5          wmanIf2BsProvSfId."
6      INDEX { wmanIf2BsSsProvMacAddress, wmanIf2BsProvSfId }
7      ::= { wmanIf2BsSsProvisionedForSfTable 1 }
8
9
10
11  WmanIf2BsSsProvisionedForSfEntry ::= SEQUENCE {
12      wmanIf2BsSsProvMacAddress      MacAddress,
13      wmanIf2BsProvSfId              Unsigned32,
14      wmanIf2BsSsProvisionedForSfRowStatus  RowStatus}
15
16
17  wmanIf2BsSsProvMacAddress OBJECT-TYPE
18      SYNTAX      MacAddress
19      MAX-ACCESS  not-accessible
20      STATUS      current
21      DESCRIPTION
22          "The MAC address of the SS, the service flow is created
23          with."
24      ::= { wmanIf2BsSsProvisionedForSfEntry 1 }
25
26
27
28  wmanIf2BsProvSfId OBJECT-TYPE
29      SYNTAX      Unsigned32 (1 .. 4294967295)
30      MAX-ACCESS  not-accessible
31      STATUS      current
32      DESCRIPTION
33          "A 32 bit quantity that uniquely identifies a service flow.
34          The value of this object can be used by BS to index the
35          wman2BsProvisionedSfTable."
36      ::= { wmanIf2BsSsProvisionedForSfEntry 2 }
37
38
39
40  wmanIf2BsSsProvisionedForSfRowStatus OBJECT-TYPE
41      SYNTAX      RowStatus
42      MAX-ACCESS  read-create
43      STATUS      current
44      DESCRIPTION
45          "This object is used to ensure that the write, create,
46          delete operation to multiple columns is guaranteed to
47          be treated as atomic operation by agent."
48      ::= { wmanIf2BsSsProvisionedForSfEntry 3 }
49
50
51
52  wmanIf2BsServiceClassTable OBJECT-TYPE
53      SYNTAX      SEQUENCE OF WmanIf2BsServiceClassEntry
54      MAX-ACCESS  not-accessible
55      STATUS      current
56      DESCRIPTION
57          "This table is provisioned and is indexed by
58          wmanIf2BsQoSProfileIndex. Each entry of the table contains
59          corresponding service flow characteristic attributes
60          (e.g. QoS parameter set). The value of
61          wmanIf2BsQoSProfileIndex is obtained from
62          wmanIf2BsServiceClassIndex in wmanIf2BsProvisionedSfTable"
63      REFERENCE
64
65

```

```

1         "Subclause 6.3.14.4 in IEEE Std 802.16-2004"
2         ::= { wmanIf2BsPacketCs 3 }
3
4
5 wmanIf2BsServiceClassEntry OBJECT-TYPE
6     SYNTAX      WmanIf2BsServiceClassEntry
7     MAX-ACCESS  not-accessible
8     STATUS      current
9     DESCRIPTION
10        "This table provides one row for each service class"
11     INDEX { ifIndex, wmanIf2BsQoSProfileIndex }
12     ::= { wmanIf2BsServiceClassTable 1 }
13
14
15 WmanIf2BsServiceClassEntry ::= SEQUENCE {
16     wmanIf2BsQoSProfileIndex      INTEGER,
17     wmanIf2BsQoSServiceClassName  WmanIf2ServClassName,
18     wmanIf2BsQoSSTrafficPriority   INTEGER,
19     wmanIf2BsQoSMaxSustainedRate   Unsigned32,
20     wmanIf2BsQoSMaxTrafficBurst    Unsigned32,
21     wmanIf2BsQoSMinReservedRate    Unsigned32,
22     wmanIf2BsQOSToleratedJitter    Unsigned32,
23     wmanIf2BsQoSMaxLatency         Unsigned32,
24     wmanIf2BsQoSFixedVsVariableSduInd INTEGER,
25     wmanIf2BsQOSSduSize            Unsigned32,
26     wmanIf2BsQoSScSchedulingType   WmanIf2SfSchedulingType,
27     wmanIf2BsQoSScArqEnable        TruthValue,
28     wmanIf2BsQoSScArqWindowSize    INTEGER,
29     wmanIf2BsQoSScArqBlockLifetime INTEGER,
30     wmanIf2BsQoSScArqSyncLossTimeout INTEGER,
31     wmanIf2BsQoSScArqDeliverInOrder TruthValue,
32     wmanIf2BsQoSScArqRxPurgeTimeout INTEGER,
33     wmanIf2BsQoSScArqBlockSize     INTEGER,
34     wmanIf2BsQoSSCMinRsvdTolerableRate Unsigned32,
35     wmanIf2BsQoSReqTxPolicy        BITS,
36     wmanIf2BsQOSServiceClassRowStatus RowStatus}
37
38
39 wmanIf2BsQoSProfileIndex OBJECT-TYPE
40     SYNTAX      INTEGER (1 .. 65535)
41     MAX-ACCESS  not-accessible
42     STATUS      current
43     DESCRIPTION
44        "The index value which uniquely identifies an entry
45         in the wmanIf2BsServiceClassTable"
46     ::= { wmanIf2BsServiceClassEntry 1 }
47
48
49 wmanIf2BsQoSServiceClassName OBJECT-TYPE
50     SYNTAX      WmanIf2ServClassName
51     MAX-ACCESS  read-create
52     STATUS      current
53     DESCRIPTION
54        "Refers to the Service Class Name"
55     REFERENCE
56        "Subclause 11.13.3 in IEEE Std 802.16-2004"
57     ::= { wmanIf2BsServiceClassEntry 2 }
58
59
60
61
62
63
64
65

```

```

1  wmanIf2BsQoSTrafficPriority OBJECT-TYPE
2      SYNTAX      INTEGER (0..7)
3      MAX-ACCESS  read-create
4      STATUS      current
5      DESCRIPTION
6          "The value of this parameter specifies the priority
7           assigned to a service flow. For uplink service flows,
8           the BS should use this parameter when determining
9           precedence in request service and grant generation,
10          and the SS shall preferentially select contention
11          Request opportunities for Priority Request CIDs
12          based on this priority. Higher numbers indicate higher
13          priority"
14      REFERENCE
15          "Subclause 11.13.5 in IEEE Std 802.16-2004"
16      ::= { wmanIf2BsServiceClassEntry 3 }
17
18  wmanIf2BsQoSMaxSustainedRate OBJECT-TYPE
19      SYNTAX      Unsigned32
20      UNITS        "b/s"
21      MAX-ACCESS  read-create
22      STATUS      current
23      DESCRIPTION
24          "This parameter defines the peak information rate
25           of the service. The rate is expressed in bits per
26           second and pertains to the SDUs at the input to
27           the system."
28      REFERENCE
29          "Subclause 11.13.6 in IEEE Std 802.16-2004"
30      ::= { wmanIf2BsServiceClassEntry 4 }
31
32  wmanIf2BsQoSMaxTrafficBurst OBJECT-TYPE
33      SYNTAX      Unsigned32
34      UNITS        "byte"
35      MAX-ACCESS  read-create
36      STATUS      current
37      DESCRIPTION
38          "This parameter defines the maximum burst size that
39           must be accommodated for the service."
40      REFERENCE
41          "Subclause 11.13.7 in IEEE Std 802.16-2004"
42      ::= { wmanIf2BsServiceClassEntry 5 }
43
44  wmanIf2BsQoSMinReservedRate OBJECT-TYPE
45      SYNTAX      Unsigned32
46      UNITS        "b/s"
47      MAX-ACCESS  read-create
48      STATUS      current
49      DESCRIPTION
50          "This parameter specifies the minimum rate reserved
51           for this service flow."
52      REFERENCE
53          "Subclause 11.13.8 in IEEE Std 802.16-2004"
54      ::= { wmanIf2BsServiceClassEntry 6 }

```

```

1
2 wmanIf2BsQoSSToleratedJitter OBJECT-TYPE
3     SYNTAX      Unsigned32
4     UNITS       "millisecond"
5     MAX-ACCESS  read-create
6     STATUS      current
7     DESCRIPTION
8         "This parameter defines the Maximum delay
9         variation (jitter) for the connection."
10
11 REFERENCE
12     "Subclause 11.13.13 in IEEE Std 802.16-2004"
13 ::= { wmanIf2BsServiceClassEntry 7 }
14
15
16 wmanIf2BsQoSMaxLatency OBJECT-TYPE
17     SYNTAX      Unsigned32
18     UNITS       "millisecond"
19     MAX-ACCESS  read-create
20     STATUS      current
21     DESCRIPTION
22         "The value of this parameter specifies the maximum
23         latency between the reception of a packet by the BS
24         or SS on its network interface and the forwarding
25         of the packet to its RF Interface."
26
27 REFERENCE
28     "Subclause 11.13.14 in IEEE Std 802.16-2004"
29 ::= { wmanIf2BsServiceClassEntry 8 }
30
31
32 wmanIf2BsQoSFixedVsVariableSduInd OBJECT-TYPE
33     SYNTAX      INTEGER {variableLength(0),
34                        fixedLength(1)}
35     MAX-ACCESS  read-create
36     STATUS      current
37     DESCRIPTION
38         "The value of this parameter specifies whether the SDUs
39         on the service flow are variable-length (0) or
40         fixed-length (1). The parameter is used only if
41         packing is on for the service flow. The default value
42         is 0, i.e., variable-length SDUs."
43
44 REFERENCE
45     "Subclause 11.13.15 in IEEE Std 802.16-2004"
46
47 DEFVAL { variableLength }
48 ::= { wmanIf2BsServiceClassEntry 9 }
49
50
51
52 wmanIf2BsQoSsdSize OBJECT-TYPE
53     SYNTAX      Unsigned32
54     UNITS       "byte"
55     MAX-ACCESS  read-create
56     STATUS      current
57     DESCRIPTION
58         "The value of this parameter specifies the length of the
59         SDU for a fixed-length SDU service flow. This parameter
60         is used only if packing is on and the service flow is
61         indicated as carrying fixed-length SDUs. The default
62         value is 49 bytes, i.e., VC-switched ATM cells with PHS.
63
64
65

```

```

1           The parameter is relevant for both ATM and Packet
2           Convergence Sublayers."
3
4       REFERENCE
5           "Subclause 11.13.16 in IEEE Std 802.16-2004"
6       DEFVAL      { 49 }
7       ::= { wmanIf2BsServiceClassEntry 10 }
8
9
10      wmanIf2BsQosScSchedulingType OBJECT-TYPE
11          SYNTAX      WmanIf2SfSchedulingType
12          MAX-ACCESS   read-create
13          STATUS      current
14          DESCRIPTION
15              "Specifies the upstream scheduling service used for
16              upstream service flow. If the referenced parameter
17              is not present in the corresponding 802.16 QOS
18              Parameter Set of an upstream service flow, the
19              default value of this object is bestEffort(2)."

```

```

1         time limit is reached, the fragment is discarded.
2         A value of 0 means Infinite."
3
4     REFERENCE
5         "Subclause 11.13.18 in IEEE Std 802.16-2004"
6     DEFVAL    {0}
7     ::= { wmanIf2BsServiceClassEntry 14 }
8
9
10    wmanIf2BsQosScArqSyncLossTimeout OBJECT-TYPE
11        SYNTAX      INTEGER (0 .. 65535 )
12        UNITS        "10 us"
13        MAX-ACCESS   read-create
14        STATUS       current
15        DESCRIPTION
16            "The maximum interval before declaring a loss
17             of synchronization of the sender and receiver
18             state machines. A value of 0 means Infinite."
19        REFERENCE
20            "Subclause 11.13.18 in IEEE Std 802.16-2004"
21        DEFVAL    {0}
22        ::= { wmanIf2BsServiceClassEntry 15 }
23
24
25
26    wmanIf2BsQosScArqDeliverInOrder OBJECT-TYPE
27        SYNTAX      TruthValue
28        MAX-ACCESS   read-create
29        STATUS       current
30        DESCRIPTION
31            "Indicates whether or not data is to be delivered
32             by the receiving MAC to its client application
33             in the order in which data was handed off to the
34             originating MAC."
35        REFERENCE
36            "Subclause 11.13.18 in IEEE Std 802.16-2004"
37        ::= { wmanIf2BsServiceClassEntry 16 }
38
39
40
41
42    wmanIf2BsQosScArqRxPurgeTimeout OBJECT-TYPE
43        SYNTAX      INTEGER (0 .. 65535)
44        UNITS        "10 us"
45        MAX-ACCESS   read-create
46        STATUS       current
47        DESCRIPTION
48            "Indicates the time interval the ARQ window is advanced
49             after a fragment is received. A value of 0 means
50             Infinite."
51        REFERENCE
52            "Subclause 11.13.18 in IEEE Std 802.16-2004"
53        DEFVAL    {0}
54        ::= { wmanIf2BsServiceClassEntry 17 }
55
56
57
58
59    wmanIf2BsQosScArqBlockSize OBJECT-TYPE
60        SYNTAX      INTEGER (1..2040)
61        UNITS        "byte"
62        MAX-ACCESS   read-create
63        STATUS       current
64        DESCRIPTION
65

```


"The value of this parameter specifies the size of an ARQ block. This parameter shall be established by negotiation during the connection creation dialog."

REFERENCE

"Subclause 11.13.18.8 in IEEE Std 802.16-2004"
`::= { wmanIf2BsServiceClassEntry 18 }`

wmanIf2BsQoSsCMinRsVdTolerableRate OBJECT-TYPE

SYNTAX Unsigned32

UNITS "b/s"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Minimum Tolerable Traffic Rate = R (bits/sec) with time base T(sec) means the following. Let S denote additional demand accumulated at the MAC SAP of the transmitter during an arbitrary time interval of the length T. Then the amount of data forwarded at the receiver to CS (in bits) during this interval should be not less than min {S, R * T}."

REFERENCE

"Subclause 11.13.9 in IEEE Std 802.16-2004"
`::= { wmanIf2BsServiceClassEntry 19 }`

wmanIf2BsQoSReqTxPolicy OBJECT-TYPE

SYNTAX BITS {noBroadcastBwReq(0),
 reserved1(1),
 noPiggybackReq(2),
 noFragmentData(3),
 noPHS(4),
 noSduPacking(5),
 noCrc(6),
 reserved2(7)}

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The value of this parameter provides the capability to specify certain attributes for the associated service flow. An attribute is enabled by setting the corresponding bit position to 1."

REFERENCE "Subclause 11.13.12 in IEEE Std 802.16-2004"
`::= { wmanIf2BsServiceClassEntry 20 }`

wmanIf2BsQoSServiceClassRowStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This object is used to create a new row or modify or delete an existing row in this table."

If the implementator of this MIB has chosen not to implement 'dynamic assignment' of profiles, this object is not useful and should return noSuchName

```

1      upon SNMP request."
2      ::= { wmanIf2BsServiceClassEntry 21 }
3
4
5  wmanIf2BsClassifierRuleTable OBJECT-TYPE
6      SYNTAX      SEQUENCE OF WmanIf2BsClassifierRuleEntry
7      MAX-ACCESS  not-accessible
8      STATUS      current
9      DESCRIPTION
10         "This table contains packet classifier rules associated
11         with service flows."
12
13     REFERENCE
14         "Subclause 11.13.19.3.4 in IEEE Std 802.16-2004"
15     ::= { wmanIf2BsPacketCs 4 }
16
17
18  wmanIf2BsClassifierRuleEntry OBJECT-TYPE
19      SYNTAX      WmanIf2BsClassifierRuleEntry
20      MAX-ACCESS  not-accessible
21      STATUS      current
22      DESCRIPTION
23         "This table provides one row for each packet classifier
24         rule, and is indexed by ifIndex, wmanIf2BsSfId, and
25         wmanIf2BsClassifierRuleIndex. IfIndex is associated with
26         the BS sector. wmanIf2BsSfId identifies the service flow,
27         while wmanIf2BsClassifierRuleIndex identifies the packet
28         classifier rule."
29
30     INDEX { ifIndex, wmanIf2BsSfId, wmanIf2BsClassifierRuleIndex }
31     ::= { wmanIf2BsClassifierRuleTable 1 }
32
33
34
35  WmanIf2BsClassifierRuleEntry ::= SEQUENCE {
36      wmanIf2BsClassifierRuleIndex      Unsigned32,
37      wmanIf2BsClassifierRulePriority    INTEGER,
38      wmanIf2BsClassifierRuleIpTosLow   INTEGER,
39      wmanIf2BsClassifierRuleIpTosHigh  INTEGER,
40      wmanIf2BsClassifierRuleIpTosMask  INTEGER,
41      wmanIf2BsClassifierRuleIpProtocol Integer32,
42      wmanIf2BsClassifierRuleIpSourceAddr InetAddress,
43      wmanIf2BsClassifierRuleIpSourceMask InetAddress,
44      wmanIf2BsClassifierRuleIpDestAddr  InetAddress,
45      wmanIf2BsClassifierRuleIpDestMask  InetAddress,
46      wmanIf2BsClassifierRuleSourcePortStart Integer32,
47      wmanIf2BsClassifierRuleSourcePortEnd Integer32,
48      wmanIf2BsClassifierRuleDestPortStart Integer32,
49      wmanIf2BsClassifierRuleDestPortEnd Integer32,
50      wmanIf2BsClassifierRuleDestMacAddr MacAddress,
51      wmanIf2BsClassifierRuleDestMacMask MacAddress,
52      wmanIf2BsClassifierRuleSourceMacAddr MacAddress,
53      wmanIf2BsClassifierRuleSourceMacMask MacAddress,
54      wmanIf2BsClassifierRuleEnetProtocolType INTEGER,
55      wmanIf2BsClassifierRuleEnetProtocol Integer32,
56      wmanIf2BsClassifierRuleUserPriLow Integer32,
57      wmanIf2BsClassifierRuleUserPriHigh Integer32,
58      wmanIf2BsClassifierRuleVlanId      Integer32,
59      wmanIf2BsClassifierRuleState       INTEGER,
60      wmanIf2BsClassifierRulePhsSize      Integer32,
61
62
63
64
65

```

```

1      wmanIf2BsClassifierRulePhsMask      OCTET STRING,
2      wmanIf2BsClassifierRulePhsVerify    WmanIf2PhsRuleVerify,
3      wmanIf2BsClassifierRuleIpv6FlowLabel WmanIf2Ipv6FlowLabel,
4      wmanIf2BsClassifierRuleBitMap       WmanIf2ClassifierBitMap,
5      wmanIf2BsClassifierRuleRowStatus    RowStatus}
6
7
8      wmanIf2BsClassifierRuleIndex OBJECT-TYPE
9          SYNTAX      Unsigned32 (1..4294967295)
10         MAX-ACCESS  not-accessible
11         STATUS      current
12         DESCRIPTION
13             "An index is assigned to a classifier in BS classifiers
14              table"
15         ::= { wmanIf2BsClassifierRuleEntry 1 }
16
17
18      wmanIf2BsClassifierRulePriority OBJECT-TYPE
19          SYNTAX      INTEGER (0..255)
20          MAX-ACCESS  read-create
21          STATUS      current
22          DESCRIPTION
23              "The value specifies the priority for the Classifier, which
24               is used for determining the order of the Classifier. A
25               higher value indicates higher priority. Classifiers may
26               have priorities in the range 0..255."
27          REFERENCE
28              "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
29          DEFVAL      { 0 }
30          ::= { wmanIf2BsClassifierRuleEntry 2 }
31
32
33      wmanIf2BsClassifierRuleIpTosLow OBJECT-TYPE
34          SYNTAX      INTEGER (0..255)
35          MAX-ACCESS  read-create
36          STATUS      current
37          DESCRIPTION
38              "The low value of a range of TOS byte values. If the
39               referenced parameter is not present in a classifier, this
40               object reports the value of 0."
41          REFERENCE
42              "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
43          ::= { wmanIf2BsClassifierRuleEntry 3 }
44
45
46      wmanIf2BsClassifierRuleIpTosHigh OBJECT-TYPE
47          SYNTAX      INTEGER (0..255)
48          MAX-ACCESS  read-create
49          STATUS      current
50          DESCRIPTION
51              "The 8-bit high value of a range of TOS byte values.
52               If the referenced parameter is not present in a classifier,
53               this object reports the value of 0."
54          REFERENCE
55              "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
56          ::= { wmanIf2BsClassifierRuleEntry 4 }
57
58
59      wmanIf2BsClassifierRuleIpTosMask OBJECT-TYPE
60
61
62
63
64
65

```

```

1      SYNTAX      INTEGER (0..255)
2      MAX-ACCESS  read-create
3      STATUS      current
4      DESCRIPTION
5
6          "The value of this object specifies the matching parameter
7          for the IP type of service/DSCP [IETF RFC 2474] byte mask.
8          An IP packet with IP type of service (ToS) byte value
9          ip-tos matches this parameter if tos-low less than or
10         equal (ip-tos AND tos-mask) less than or equal tos-high."
11
12     REFERENCE
13         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
14     ::= { wmanIf2BsClassifierRuleEntry 5 }
15
16 wmanIf2BsClassifierRuleIpProtocol OBJECT-TYPE
17     SYNTAX      Integer32 (0..255)
18     MAX-ACCESS  read-create
19     STATUS      current
20     DESCRIPTION
21
22         "This object indicates the value of the IP Protocol field
23         required for IP packets to match this rule. If the
24         referenced parameter is not present in a classifier, this
25         object reports the value of 0."
26
27     REFERENCE
28         "Subclause 11.13.19.3.4.3 in IEEE Std 802.16-2004"
29     ::= { wmanIf2BsClassifierRuleEntry 6 }
30
31
32 wmanIf2BsClassifierRuleIpSourceAddr OBJECT-TYPE
33     SYNTAX      InetAddress
34     MAX-ACCESS  read-create
35     STATUS      current
36     DESCRIPTION
37
38         "This object specifies the value of the IP Source Address
39         required for packets to match this rule. An IP packet
40         matches the rule when the packet ip source address bitwise
41         ANDed with the wmanIf2BsClassifierRuleIpSourceMask value
42         equals the wmanIf2BsClassifierRuleIpSourceAddr value.
43         If the referenced parameter is not present in a classifier,
44         this object reports the value of 0.0.0.0."
45
46     REFERENCE
47         "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
48     ::= { wmanIf2BsClassifierRuleEntry 7 }
49
50
51 wmanIf2BsClassifierRuleIpSourceMask OBJECT-TYPE
52     SYNTAX      InetAddress
53     MAX-ACCESS  read-create
54     STATUS      current
55     DESCRIPTION
56
57         "This object specifies which bits of a packet's IP Source
58         Address that are compared to match this rule. An IP packet
59         matches the rule when the packet source address bitwise
60         ANDed with the
61         wmanIf2BsClassifierRuleIpSourceMask value equals the
62         wmanIf2BsClassifierRuleIpSourceAddr value.
63         If the referenced parameter is not present in a classifier,
64
65

```

```

1         this object reports the value of 0.0.0.0."
2     REFERENCE
3         "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
4     ::= { wmanIf2BsClassifierRuleEntry 8 }
5
6
7     wmanIf2BsClassifierRuleIpDestAddr OBJECT-TYPE
8         SYNTAX      InetAddress
9         MAX-ACCESS   read-create
10        STATUS      current
11        DESCRIPTION
12            "This object specifies the value of the IP Destination
13             Address required for packets to match this rule. An IP
14             packet matches the rule when the packet IP destination
15             address bitwise ANDed with the
16             wmanIf2BsClassifierRuleIpDestMask value equals the
17             wmanIf2BsClassifierRuleIpDestAddr value.
18             If the referenced parameter is not present in a
19             classifier, this object reports the value of 0.0.0.0."
20        REFERENCE
21            "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
22        ::= { wmanIf2BsClassifierRuleEntry 9 }
23
24
25     wmanIf2BsClassifierRuleIpDestMask OBJECT-TYPE
26         SYNTAX      InetAddress
27         MAX-ACCESS   read-create
28         STATUS      current
29         DESCRIPTION
30            "This object specifies which bits of a packet's IP
31             Destination Address that are compared to match this rule.
32             An IP packet matches the rule when the packet destination
33             address bitwise ANDed with the
34             wmanIf2BsClassifierRuleIpDestMask value equals the
35             wmanIf2BsClassifierRuleIpDestAddr value.
36             If the referenced parameter is not present in a classifier
37             , this object reports the value of 0.0.0.0."
38        REFERENCE
39            "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
40        ::= { wmanIf2BsClassifierRuleEntry 10 }
41
42
43     wmanIf2BsClassifierRuleSourcePortStart OBJECT-TYPE
44         SYNTAX      Integer32 (0..65535)
45         MAX-ACCESS   read-create
46         STATUS      current
47         DESCRIPTION
48            "This object specifies the low end inclusive range of
49             TCP/UDP source port numbers to which a packet is compared.
50             This object is irrelevant for non-TCP/UDP IP packets.
51             If the referenced parameter is not present in a
52             classifier, this object reports the value of 0."
53        REFERENCE
54            "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
55        ::= { wmanIf2BsClassifierRuleEntry 11 }
56
57
58     wmanIf2BsClassifierRuleSourcePortEnd OBJECT-TYPE

```

```

1      SYNTAX      Integer32 (0..65535)
2      MAX-ACCESS  read-create
3      STATUS      current
4      DESCRIPTION
5          "This object specifies the high end inclusive range of
6          TCP/UDP source port numbers to which a packet is compared.
7          This object is irrelevant for non-TCP/UDP IP packets.
8          If the referenced parameter is not present in a classifier,
9          this object reports the value of 65535."
10     REFERENCE
11         "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
12     ::= { wmanIf2BsClassifierRuleEntry 12 }
13
14 wmanIf2BsClassifierRuleDestPortStart OBJECT-TYPE
15     SYNTAX      Integer32 (0..65535)
16     MAX-ACCESS  read-create
17     STATUS      current
18     DESCRIPTION
19         "This object specifies the low end inclusive range of
20         TCP/UDP destination port numbers to which a packet is
21         compared. If the referenced parameter is not present
22         in a classifier, this object reports the value of 0."
23     REFERENCE
24         "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
25     ::= { wmanIf2BsClassifierRuleEntry 13 }
26
27 wmanIf2BsClassifierRuleDestPortEnd OBJECT-TYPE
28     SYNTAX      Integer32 (0..65535)
29     MAX-ACCESS  read-create
30     STATUS      current
31     DESCRIPTION
32         "This object specifies the high end inclusive range of
33         TCP/UDP destination port numbers to which a packet is
34         compared. If the referenced parameter is not present
35         in a classifier, this object reports the value of
36         65535."
37     REFERENCE
38         "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
39     ::= { wmanIf2BsClassifierRuleEntry 14 }
40
41 wmanIf2BsClassifierRuleDestMacAddr OBJECT-TYPE
42     SYNTAX      MacAddress
43     MAX-ACCESS  read-create
44     STATUS      current
45     DESCRIPTION
46         "An Ethernet packet matches an entry when its destination
47         MAC address bitwise ANDed with
48         wmanIf2BsClassifierRuleDestMacMask equals the value of
49         wmanIf2BsClassifierRuleDestMacAddr. If the referenced
50         parameter is not present in a classifier, this object
51         reports the value of '000000000000'H."
52     REFERENCE
53         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
54     ::= { wmanIf2BsClassifierRuleEntry 15 }

```

```

1
2 wmanIf2BsClassifierRuleDestMacMask OBJECT-TYPE
3     SYNTAX      MacAddress
4     MAX-ACCESS  read-create
5     STATUS      current
6     DESCRIPTION
7         "An Ethernet packet matches an entry when its destination
8         MAC address bitwise ANDed with
9         wmanIf2BsClassifierRuleDestMacMask equals the value of
10        wmanIf2BsClassifierRuleDestMacAddr. If the referenced
11        parameter is not present in a classifier, this object
12        reports the value of '000000000000'H."
13
14     REFERENCE
15         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
16     ::= { wmanIf2BsClassifierRuleEntry 16 }
17
18 wmanIf2BsClassifierRuleSourceMacAddr OBJECT-TYPE
19     SYNTAX      MacAddress
20     MAX-ACCESS  read-create
21     STATUS      current
22     DESCRIPTION
23         "An Ethernet packet matches this entry when its source
24         MAC address bitwise ANDed with
25         wmanIf2BsClassifierRuleSourceMacMask equals the value
26         of wmanIf2BsClassifierRuleSourceMacAddr. If the
27         referenced parameter is not present in a classifier,
28         this object reports the value of '000000000000'H."
29
30     REFERENCE
31         "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
32     ::= { wmanIf2BsClassifierRuleEntry 17 }
33
34 wmanIf2BsClassifierRuleSourceMacMask OBJECT-TYPE
35     SYNTAX      MacAddress
36     MAX-ACCESS  read-create
37     STATUS      current
38     DESCRIPTION
39         "An Ethernet packet matches an entry when its source
40         MAC address bitwise ANDed with
41         wmanIf2BsClassifierRuleSourceMacMask equals the value of
42         wmanIf2BsClassifierRuleSourceMacAddr. If the referenced
43         parameter is not present in a classifier, this object
44         reports the value of '000000000000'H."
45
46     REFERENCE
47         "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
48     ::= { wmanIf2BsClassifierRuleEntry 18 }
49
50 wmanIf2BsClassifierRuleEnetProtocolType OBJECT-TYPE
51     SYNTAX      INTEGER { none(0),
52                        ethertype(1),
53                        dsap(2) }
54     MAX-ACCESS  read-create
55     STATUS      current
56     DESCRIPTION
57         "This object indicates the format of the layer 3 protocol
58

```

id in the Ethernet packet. A value of none(0) means that the rule does not use the layer 3 protocol type as a matching criteria. A value of ethertype(1) means that the rule applies only to frames which contains an EtherType value. Ethertype values are contained in packets using the Dec-Intel-Xerox (DIX) encapsulation or the RFC1042 Sub-Network Access Protocol (SNAP) encapsulation formats. A value of dsap(2) means that the rule applies only to frames using the IEEE802.3 encapsulation format with a Destination Service Access Point (DSAP) other than 0xAA (which is reserved for SNAP). If the Ethernet frame contains an 802.1P/Q Tag header (i.e. EtherType 0x8100), this object applies to the embedded EtherType field within the 802.1P/Q header. If the referenced parameter is not present in a classifier, this object reports the value of 0."

REFERENCE

"Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsClassifierRuleEntry 19 }

wmanIf2BsClassifierRuleEnetProtocol OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"If wmanIf2BsClassifierRuleEnetProtocolType is none(0), this object is ignored when considering whether a packet matches the current rule.
 If wmanIf2BsClassifierRuleEnetProtocolType is ethertype(1), this object gives the 16-bit value of the EtherType that the packet must match in order to match the rule.
 If wmanIf2BsClassifierRuleEnetProtocolType is dsap(2), the lower 8 bits of this object's value must match the DSAP byte of the packet in order to match the rule.
 If the Ethernet frame contains an 802.1P/Q Tag header (i.e. EtherType 0x8100), this object applies to the embedded EtherType field within the 802.1P/Q header.
 If the referenced parameter is not present in the classifier, the value of this object is reported as 0."

REFERENCE

"Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsClassifierRuleEntry 20 }

wmanIf2BsClassifierRuleUserPriLow OBJECT-TYPE

SYNTAX Integer32 (0..7)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This object applies only to Ethernet frames using the 802.1P/Q tag header (indicated with EtherType 0x8100). Such frames include a 16-bit Tag that contains a 3 bit Priority field and a 12 bit VLAN number. Tagged Ethernet packets must have a 3-bit Priority field within the range of wmanIf2BsClassifierRuleUserPriLow and


```

1         wmanIf2BsClassifierRuleUserPriHigh in order to match this
2         rule.
3         If the referenced parameter is not present in the
4         classifier, the value of this object is reported as 0."
5     REFERENCE
6         "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
7     ::= { wmanIf2BsClassifierRuleEntry 21 }
8
9
10    wmanIf2BsClassifierRuleUserPriHigh OBJECT-TYPE
11        SYNTAX      Integer32 (0..7)
12        MAX-ACCESS   read-create
13        STATUS       current
14        DESCRIPTION
15            "This object applies only to Ethernet frames using the
16            802.1P/Q tag header (indicated with EtherType 0x8100).
17            Such frames include a 16-bit Tag that contains a 3 bit
18            Priority field and a 12 bit VLAN number.
19            Tagged Ethernet packets must have a 3-bit Priority
20            field within the range of wmanIf2BsClassifierRuleUserPriLow
21            and wmanIf2BsClassifierRuleUserPriHigh in order to match
22            this rule.
23            If the referenced parameter is not present in the
24            classifier, the value of this object is reported as 7."
25        REFERENCE
26            "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
27        ::= { wmanIf2BsClassifierRuleEntry 22 }
28
29
30    wmanIf2BsClassifierRuleVlanId OBJECT-TYPE
31        SYNTAX      Integer32 (0..4095)
32        MAX-ACCESS   read-create
33        STATUS       current
34        DESCRIPTION
35            "This object applies only to Ethernet frames using the
36            802.1P/Q tag header.
37            If this object's value is nonzero, tagged packets must
38            have a VLAN Identifier that matches the value in order
39            to match the rule.
40            Only the least significant 12 bits of this object's
41            value are valid.
42            If the referenced parameter is not present in the
43            classifier, the value of this object is reported as 0."
44        REFERENCE
45            "Subclause 11.13.19.3.4.12 in IEEE Std 802.16-2004"
46        ::= { wmanIf2BsClassifierRuleEntry 23 }
47
48
49    wmanIf2BsClassifierRuleState OBJECT-TYPE
50        SYNTAX      INTEGER {active(1),
51                           inactive(2)}
52        MAX-ACCESS   read-create
53        STATUS       deprecated
54        DESCRIPTION
55            "This object indicates whether or not the classifier is
56            enabled to classify packets to a Service Flow.
57            If the referenced parameter is not present in the

```

```

1         classifier, the value of this object is reported
2         as active(1)."
```

::= { wmanIf2BsClassifierRuleEntry 24 }

wmanIf2BsClassifierRulePhsSize OBJECT-TYPE

```

7     SYNTAX      Integer32 (0..255)
8     UNITS       "byte"
9     MAX-ACCESS  read-create
10    STATUS      current
11    DESCRIPTION
12
13        "This object is used to configure the PHS rule for this
14        classifier. The value of this field - PHSS is the total
15        number of bytes in the header to be suppressed and then
16        restored in a service flow that uses PHS. If the value of
17        this field is 0 bytes then PHS is disabled for this
18        classifier. If flag phsMask in wmanIf2BsClassifierRuleBitMap
19        is set to 0 and flag phsSize in
20        wmanIf2BsClassifierRuleBitMap is set to 0, then BS can still
21        create PHS rules using its own custom mask (i.e. the rule
22        is not configured by NMS)."
```

REFERENCE

```

26        "Subclause 11.13.19.3.7.4 in IEEE Std 802.16-2004"
```

DEFVAL {0}

::= { wmanIf2BsClassifierRuleEntry 25 }

wmanIf2BsClassifierRulePhsMask OBJECT-TYPE

```

32    SYNTAX      OCTET STRING (SIZE(0..65535))
33    MAX-ACCESS  read-create
34    STATUS      current
35    DESCRIPTION
36
37        "This object is used to configure the PHS rule for this
38        classifier. It is encoded as follows:
39        bit 0:
40            0 = don't suppress the 1st byte of the suppression field
41            1 = suppress first byte of the suppression field
42        bit 1:
43            0 = don't suppress the 2nd byte of the suppression field
44            1 = suppress second byte of the suppression field
45        bit x:
46            0 = don't suppress the (x+1) byte of the suppression
47            field
48            1 = suppress (x+1) byte of the suppression field
49        where the length of the octet string is ceiling
50        (wmanIf2BsClassifierRulePhsSize/8). BS should use this value
51        to create a new PHS rule index (PHSI) and field (PHSF) as
52        defined in the standard. If flag phsMask in
53        wmanIf2BsClassifierRuleBitMap is set to 0 and flag phsSize
54        in wmanIf2BsClassifierRuleBitMap is set to 0, then BS can
55        still create PHS rules using its own custom mask (i.e. the
56        rule is not configured by NMS)."
```

REFERENCE

```

62        "Subclause 11.13.19.3.7.3 in IEEE Std 802.16-2004"
```

::= { wmanIf2BsClassifierRuleEntry 26 }

```

1  wmanIf2BsClassifierRulePhsVerify OBJECT-TYPE
2      SYNTAX      WmanIf2PhsRuleVerify
3      MAX-ACCESS  read-create
4      STATUS      current
5      DESCRIPTION
6          "The value of this field indicates to the sending entity
7              whether or not the packet header contents are to be
8              verified prior to performing suppression."
9      DEFVAL      { phsVerifyEnable }
10     ::= { wmanIf2BsClassifierRuleEntry 27 }
11
12  wmanIf2BsClassifierRuleIpv6FlowLabel OBJECT-TYPE
13      SYNTAX      WmanIf2Ipv6FlowLabel
14      MAX-ACCESS  read-create
15      STATUS      current
16      DESCRIPTION
17          "The value of this field specifies the matching values for
18              the IPv6 Flow label field."
19      ::= { wmanIf2BsClassifierRuleEntry 28 }
20
21  wmanIf2BsClassifierRuleBitMap OBJECT-TYPE
22      SYNTAX      WmanIf2ClassifierBitMap
23      MAX-ACCESS  read-create
24      STATUS      current
25      DESCRIPTION
26          "This object indicates which parameter encodings were
27              actually present in the entry. A bit set to '1' indicates
28              the corresponding classifier encoding is present, and '0'
29              means otherwise"
30      ::= { wmanIf2BsClassifierRuleEntry 29 }
31
32  wmanIf2BsClassifierRuleRowStatus OBJECT-TYPE
33      SYNTAX      RowStatus
34      MAX-ACCESS  read-create
35      STATUS      current
36      DESCRIPTION
37          "This object is used to create a new row or modify or
38              delete an existing row in this table.
39
40              If the implementator of this MIB has choosen not
41              to implement 'dynamic assignment' of profiles, this
42              object is not useful and should return noSuchName
43              upon SNMP request."
44      ::= { wmanIf2BsClassifierRuleEntry 30 }
45
46  wmanIf2BsSsPacketCounterTable OBJECT-TYPE
47      SYNTAX      SEQUENCE OF WmanIf2BsSsPacketCounterEntry
48      MAX-ACCESS  not-accessible
49      STATUS      current
50      DESCRIPTION
51          "This table contains counters to keep track of the number
52              of packets and octets that have been received or
53              transmitted on the per service flow basis."
54      ::= { wmanIf2BsPacketCs 5 }

```

```

1
2 wmanIf2BsSsPacketCounterEntry OBJECT-TYPE
3     SYNTAX          WmanIf2BsSsPacketCounterEntry
4     MAX-ACCESS      not-accessible
5     STATUS          current
6     DESCRIPTION
7         "This table provides one row for each service flow, and
8         is indexed by ifIndex, wmanIf2CmnCpsSfMacAddress, and
9         wmanIf2CmnCpsSfId."
10    INDEX { ifIndex, wmanIf2CmnCpsSfMacAddress,
11            wmanIf2CmnCpsSfId }
12    ::= { wmanIf2BsSsPacketCounterTable 1 }
13
14 WmanIf2BsSsPacketCounterEntry ::= SEQUENCE {
15     wmanIf2BsSsMacSduCount          Counter64,
16     wmanIf2BsSsOctetCount          Counter64,
17     wmanIf2BsSsResetCounter        INTEGER,
18     wmanIf2BsSsResetCounterTime    TimeStamp}
19
20 wmanIf2BsSsMacSduCount OBJECT-TYPE
21     SYNTAX          Counter64
22     MAX-ACCESS      read-only
23     STATUS          current
24     DESCRIPTION
25         "This object counts the number of MAC SDUs that have
26         been transmitted or received."
27     ::= { wmanIf2BsSsPacketCounterEntry 1 }
28
29 wmanIf2BsSsOctetCount OBJECT-TYPE
30     SYNTAX          Counter64
31     MAX-ACCESS      read-only
32     STATUS          current
33     DESCRIPTION
34         "This object counts the number of octets of MAC SDUs
35         that have been transmitted or received."
36     ::= { wmanIf2BsSsPacketCounterEntry 2 }
37
38 wmanIf2BsSsResetCounter OBJECT-TYPE
39     SYNTAX          INTEGER {null(0),
40                             resetCounter(1)}
41     MAX-ACCESS      read-write
42     STATUS          current
43     DESCRIPTION
44         "When this attribute is SET to resetCounter(1), the
45         corresponding entry of packet counters will be reset.
46         A GET operation performed on this object will always
47         return null(0). The counter is normally reset after
48         the packet count information is retrieved. "
49     ::= { wmanIf2BsSsPacketCounterEntry 3 }
50
51 wmanIf2BsSsResetCounterTime OBJECT-TYPE
52     SYNTAX          TimeStamp
53     MAX-ACCESS      read-only
54     STATUS          current
55
56
57
58
59
60
61
62
63
64
65

```

```

1      DESCRIPTION
2          "Indicates the date and time when the counter is
3          reset."
4      ::= { wmanIf2BsSsPacketCounterEntry 4 }
5
6
7      --
8      -- wmanIf2BsCps contain the Base Station Common Part Sublayer objects
9      --
10
11  wmanIf2BsCps OBJECT IDENTIFIER ::= { wmanIf2BsObjects 2 }
12
13  wmanIf2BsRegisteredSsTable OBJECT-TYPE
14      SYNTAX      SEQUENCE OF WmanIf2BsRegisteredSsEntry
15      MAX-ACCESS  not-accessible
16      STATUS      current
17      DESCRIPTION
18          "This table contains the basic capability information
19          of SSs that have been negotiated and agreed between
20          BS and SS via REG-REQ and REG-RSP messages. An entry
21          in this table indicates the SS has entered and registered
22          into the BS."
23
24  REFERENCE
25      "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
26  ::= { wmanIf2BsCps 1 }
27
28
29
30  wmanIf2BsRegisteredSsEntry OBJECT-TYPE
31      SYNTAX      WmanIf2BsRegisteredSsEntry
32      MAX-ACCESS  not-accessible
33      STATUS      current
34      DESCRIPTION
35          "This table provides one row for each SS that has been
36          registered in the BS, and is indexed by
37          wmanIf2BsSsMacAddress. The primary index is the ifIndex
38          with an ifType of propBWAmp2Mp, indicating the BS sector
39          with which the SS is associated. wmanIf2BsSsMacAddress
40          identifies the SS being registered."
41
42  INDEX { ifIndex, wmanIf2BsSsMacAddress }
43  ::= { wmanIf2BsRegisteredSsTable 1 }
44
45
46
47  WmanIf2BsRegisteredSsEntry ::= SEQUENCE {
48      wmanIf2BsSsMacAddress      MacAddress,
49      wmanIf2BsSsBasicCid        WmanIf2CidType,
50      wmanIf2BsSsPrimaryCid      WmanIf2CidType,
51      wmanIf2BsSsSecondaryCid    WmanIf2CidType,
52      wmanIf2BsSsManagementSupport  INTEGER,
53      wmanIf2BsSsIpManagementMode  INTEGER,
54      wmanIf2BsSs2ndMgmtArqEnable  TruthValue,
55      wmanIf2BsSs2ndMgmtArqWindowSize  INTEGER,
56      wmanIf2BsSs2ndMgmtArqDnLinkTxDelay  INTEGER,
57      wmanIf2BsSs2ndMgmtArqUpLinkTxDelay  INTEGER,
58      wmanIf2BsSs2ndMgmtArqDnLinkRxDelay  INTEGER,
59      wmanIf2BsSs2ndMgmtArqUpLinkRxDelay  INTEGER,
60      wmanIf2BsSs2ndMgmtArqBlockLifetime  INTEGER,
61      wmanIf2BsSs2ndMgmtArqSyncLossTimeout  INTEGER,
62      wmanIf2BsSs2ndMgmtArqDeliverInOrder  TruthValue,
63
64
65

```

```

1      wmanIf2BsSs2ndMgmtArqRxPurgeTimeout      INTEGER,
2      wmanIf2BsSs2ndMgmtArqBlockSize           INTEGER,
3      wmanIf2BsSsVendorIdEncoding              OCTET STRING,
4      wmanIf2BsSsAasBroadcastPermission        INTEGER,
5      wmanIf2BsSsMaxTxPowerBpsk                WmanIf2MaxTxPowerType,
6      wmanIf2BsSsMaxTxPowerQpsk                WmanIf2MaxTxPowerType,
7      wmanIf2BsSsMaxTxPower16Qam               WmanIf2MaxTxPowerType,
8      wmanIf2BsSsMaxTxPower64Qam               WmanIf2MaxTxPowerType,
9      wmanIf2BsSsMacVersion                     WmanIf2MacVersion}
10
11
12
13      wmanIf2BsSsMacAddress OBJECT-TYPE
14          SYNTAX      MacAddress
15          MAX-ACCESS  not-accessible
16          STATUS      current
17          DESCRIPTION
18              "The MAC address of SS is received from the RNG-REQ
19              message. When SS registers, this MAC address is entered
20              into the table, and used as the identifier to the SS."
21          REFERENCE
22              "Subclause 6.3.2.3.5 in IEEE Std 802.16-2004"
23              ::= { wmanIf2BsRegisteredSsEntry 1 }
24
25
26
27      wmanIf2BsSsBasicCid OBJECT-TYPE
28          SYNTAX      WmanIf2CidType
29          MAX-ACCESS  read-only
30          STATUS      current
31          DESCRIPTION
32              "The value of this object indicates the SS's basic CID
33              that was sent in the RNG-RSP message."
34          REFERENCE
35              "Subclause 6.3.2.3.6 in IEEE Std 802.16-2004"
36              ::= { wmanIf2BsRegisteredSsEntry 2 }
37
38
39
40
41      wmanIf2BsSsPrimaryCid OBJECT-TYPE
42          SYNTAX      WmanIf2CidType
43          MAX-ACCESS  read-only
44          STATUS      current
45          DESCRIPTION
46              "The value of this object indicates the primary CID of the
47              SS received from the RNG-RSP message."
48          REFERENCE
49              "Subclause 6.3.2.3.6 in IEEE Std 802.16-2004"
50              ::= { wmanIf2BsRegisteredSsEntry 3 }
51
52
53
54      wmanIf2BsSsSecondaryCid OBJECT-TYPE
55          SYNTAX      WmanIf2CidType
56          MAX-ACCESS  read-only
57          STATUS      current
58          DESCRIPTION
59              "The value of this object indicates the secondary
60              management CID present in the REG-RSP message. The value
61              should be null if the 2nd management connection is not
62              available."
63          REFERENCE
64
65

```

```

1         "Subclause 6.4.2.3.8 in IEEE Std 802.16-2004"
2         ::= { wmanIf2BsRegisteredSsEntry 4 }
3
4
5 wmanIf2BsSsManagementSupport OBJECT-TYPE
6     SYNTAX      INTEGER {unmanagedSs(0),
7                     managedSs(1)}
8
9     MAX-ACCESS  read-only
10    STATUS      current
11    DESCRIPTION
12        "This object indicates whether or not the SS is managed."
13    REFERENCE
14        "Subclause 11.7.2 in IEEE Std 802.16-2004"
15    ::= { wmanIf2BsRegisteredSsEntry 5 }
16
17
18 wmanIf2BsSsIpManagementMode OBJECT-TYPE
19     SYNTAX      INTEGER {unmanaged(0),
20                     ipManaged(1)}
21
22     MAX-ACCESS  read-only
23     STATUS      current
24     DESCRIPTION
25         "The IP management mode parameter dictates whether
26         the provider intends to manage the SS on an ongoing
27         basis via IP-based mechanisms."
28     REFERENCE
29         "Subclause 11.7.3 in IEEE Std 802.16-2004"
30     ::= { wmanIf2BsRegisteredSsEntry 6 }
31
32
33 wmanIf2BsSs2ndMgmtArqEnable OBJECT-TYPE
34     SYNTAX      TruthValue
35     MAX-ACCESS  read-only
36     STATUS      current
37     DESCRIPTION
38         "True(1) ARQ enabling is requested for the 2nd
39         management channel."
40     REFERENCE
41         "Subclause 11.13.18.1 in IEEE Std 802.16-2004"
42     ::= { wmanIf2BsRegisteredSsEntry 7 }
43
44
45
46 wmanIf2BsSs2ndMgmtArqWindowSize OBJECT-TYPE
47     SYNTAX      INTEGER (1 .. 1024)
48     MAX-ACCESS  read-only
49     STATUS      current
50     DESCRIPTION
51         "Indicates the maximum number of unacknowledged
52         fragments at any time for 2nd management connection."
53     REFERENCE
54         "Subclause 11.13.18.2 in IEEE Std 802.16-2004"
55     ::= { wmanIf2BsRegisteredSsEntry 8 }
56
57
58
59 wmanIf2BsSs2ndMgmtArqDnLinkTxDelay OBJECT-TYPE
60     SYNTAX      INTEGER (0 .. 65535)
61     UNITS       "us"
62     MAX-ACCESS  read-only
63     STATUS      current
64
65

```

```

1      DESCRIPTION
2          "The object defines the ARQ transmitter delay for
3          downlink transmission."
4
5      REFERENCE
6          "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
7          ::= { wmanIf2BsRegisteredSsEntry 9 }
8
9
10     wmanIf2BsSs2ndMgmtArqUpLinkTxDelay OBJECT-TYPE
11         SYNTAX      INTEGER (0 .. 65535)
12         UNITS        "us"
13         MAX-ACCESS   read-only
14         STATUS        current
15         DESCRIPTION
16             "The object defines the ARQ transmitter delay for
17             uplink transmission."
18
19         REFERENCE
20             "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
21             ::= { wmanIf2BsRegisteredSsEntry 10 }
22
23
24     wmanIf2BsSs2ndMgmtArqDnLinkRxDelay OBJECT-TYPE
25         SYNTAX      INTEGER (0 .. 65535)
26         UNITS        "us"
27         MAX-ACCESS   read-only
28         STATUS        current
29         DESCRIPTION
30             "The object defines the ARQ receiver delay for
31             downlink transmission."
32
33         REFERENCE
34             "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
35             ::= { wmanIf2BsRegisteredSsEntry 11 }
36
37
38     wmanIf2BsSs2ndMgmtArqUpLinkRxDelay OBJECT-TYPE
39         SYNTAX      INTEGER (0 .. 65535)
40         UNITS        "us"
41         MAX-ACCESS   read-only
42         STATUS        current
43         DESCRIPTION
44             "The object defines the ARQ receiver delay for
45             uplink transmission."
46
47         REFERENCE
48             "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
49             ::= { wmanIf2BsRegisteredSsEntry 12 }
50
51
52     wmanIf2BsSs2ndMgmtArqBlockLifetime OBJECT-TYPE
53         SYNTAX      INTEGER (0 .. 65535)
54         UNITS        "10 us"
55         MAX-ACCESS   read-only
56         STATUS        current
57         DESCRIPTION
58             "The maximum time interval an ARQ fragment will be
59             managed by the transmitter ARQ machine, once
60             initial transmission of the fragment has occurred.
61             If transmission or retransmission of the fragment
62             is not acknowledged by the receiver before the
63             initial transmission of the fragment has occurred.
64             If transmission or retransmission of the fragment
65             is not acknowledged by the receiver before the

```



```

1         time limit is reached, the fragment is discarded.
2         A value of 0 means Infinite."
3     REFERENCE
4         "Subclause 11.13.18.4 in IEEE Std 802.16-2004"
5     DEFVAL    {0}
6     ::= { wmanIf2BsRegisteredSsEntry 13 }
7
8
9
10    wmanIf2BsSs2ndMgmtArqSyncLossTimeout OBJECT-TYPE
11        SYNTAX      INTEGER (0 .. 65535)
12        UNITS        "10 us"
13        MAX-ACCESS   read-only
14        STATUS        current
15        DESCRIPTION
16            "The maximum interval before declaring a loss
17             of synchronization of the sender and receiver
18             state machines. A value of 0 means Infinite."
19        REFERENCE
20            "Subclause 11.13.18.5 in IEEE Std 802.16-2004"
21        DEFVAL    {0}
22        ::= { wmanIf2BsRegisteredSsEntry 14 }
23
24
25
26    wmanIf2BsSs2ndMgmtArqDeliverInOrder OBJECT-TYPE
27        SYNTAX      TruthValue
28        MAX-ACCESS   read-only
29        STATUS        current
30        DESCRIPTION
31            "Indicates whether or not data is to be delivered
32             by the receiving MAC to its client application
33             in the order in which data was handed off to the
34             originating MAC."
35        REFERENCE
36            "Subclause 11.13.18.6 in IEEE Std 802.16-2004"
37        ::= { wmanIf2BsRegisteredSsEntry 15 }
38
39
40
41
42    wmanIf2BsSs2ndMgmtArqRxPurgeTimeout OBJECT-TYPE
43        SYNTAX      INTEGER (0 .. 65535)
44        UNITS        "10 us"
45        MAX-ACCESS   read-only
46        STATUS        current
47        DESCRIPTION
48            "Indicates the time interval the ARQ window is advanced
49             after a fragment is received. A value of 0 means Infinite."
50        REFERENCE
51            "Subclause 11.13.18.7 in IEEE Std 802.16-2004"
52        DEFVAL    {0}
53        ::= { wmanIf2BsRegisteredSsEntry 16 }
54
55
56
57    wmanIf2BsSs2ndMgmtArqBlockSize OBJECT-TYPE
58        SYNTAX      INTEGER (1 .. 2040)
59        MAX-ACCESS   read-only
60        STATUS        current
61        DESCRIPTION
62            "This parameter specifies the size of a ARQ block. This
63             parameter shall be established by negotiation during the
64             parameter shall be established by negotiation during the
65             parameter shall be established by negotiation during the

```

connection setup. The requester includes its desired setting in the REQ message. The receiver of the REQ message shall take the smaller of the value it prefers and value in the REQ message. The minimum value is included in the RSP message."

REFERENCE

"Subclause 11.13.18.8 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsRegisteredSsEntry 17 }

wmanIf2BsSsVendorIdEncoding OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(3))

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value field contains the vendor identification specified by the 3 byte vendor-specific organizationally unique identifier of the SS or BS MAC address. A vendor ID used in a REG-REQ shall be the Vendor ID of the SS sending the request. A vendor ID used in a REG-RSP shall be the Vendor ID of the BS sending the response."

REFERENCE

"Subclause 11.1.5 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsRegisteredSsEntry 18 }

wmanIf2BsSsAasBroadcastPermission OBJECT-TYPE

SYNTAX INTEGER {contBasedBwReqPermitted(0),
 contBasedBwReqNotPermitted(1)}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This parameter specifies if SS can issue contention-based bandwidth request or not."

REFERENCE

"Subclause 11.6 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsRegisteredSsEntry 19 }

wmanIf2BsSsMaxTxPowerBpsk OBJECT-TYPE

SYNTAX WmanIf2MaxTxPowerType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum available power for BPSK. The maximum power parameters are reported in dBm and quantized in 0.5 dBm steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm (encoded 0xFF). Values outside this range shall be assigned the closest extreme. This parameter is only applicable to systems supporting the SCa, OFDM or OFDMA PHY."

REFERENCE

"Subclause 11.8.3.2 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsRegisteredSsEntry 20 }

wmanIf2BsSsMaxTxPowerQpsk OBJECT-TYPE

SYNTAX WmanIf2MaxTxPowerType

MAX-ACCESS read-only

```

1      STATUS      current
2      DESCRIPTION
3          "The maximum available power for QPSK. The maximum power
4          parameters are reported in dBm and quantized in 0.5 dBm
5          steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
6          (encoded 0xFF). Values outside this range shall be assigned
7          to closest extreme. This parameter is only applicable to
8          systems supporting the SCa, OFDM or OFDMA PHY."
9
10     REFERENCE
11         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
12     ::= { wmanIf2BsRegisteredSsEntry 21 }
13
14
15     wmanIf2BsSsMaxTxPower16Qam OBJECT-TYPE
16         SYNTAX      WmanIf2MaxTxPowerType
17         MAX-ACCESS   read-only
18         STATUS      current
19         DESCRIPTION
20             "The maximum available power for 16-QAM constellations.
21             The maximum power parameters are reported in dBm and
22             quantized in 0.5 dBm steps ranging from -64 dBm (encoded
23             0x00) to 63.5 dBm (encoded 0xFF). Values outside this
24             range shall be assigned the closest extreme. This parameter
25             is only applicable to systems supporting the SCa, OFDM or
26             OFDMA PHY."
27
28     REFERENCE
29         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
30     ::= { wmanIf2BsRegisteredSsEntry 22 }
31
32
33     wmanIf2BsSsMaxTxPower64Qam OBJECT-TYPE
34         SYNTAX      WmanIf2MaxTxPowerType
35         MAX-ACCESS   read-only
36         STATUS      current
37         DESCRIPTION
38             "The maximum available power for 64-QAM constellations.
39             The maximum power parameters are reported in dBm and
40             quantized in 0.5 dBm steps ranging from -64 dBm (encoded
41             0x00) to 63.5 dBm (encoded 0xFF). Values outside this
42             range shall be assigned the closest extreme. SSs that do
43             not support QAM64 shall report the value of 0x00. This
44             parameter is only applicable to systems supporting the SCa,
45             OFDM or OFDMA PHY."
46
47     REFERENCE
48         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
49     ::= { wmanIf2BsRegisteredSsEntry 23 }
50
51
52     wmanIf2BsSsMacVersion OBJECT-TYPE
53         SYNTAX      WmanIf2MacVersion
54         MAX-ACCESS   read-only
55         STATUS      current
56         DESCRIPTION
57             "This parameter specifies the version of 802.16 to which the
58             message originator conforms."
59
60     REFERENCE
61         "Subclause 11.1.3 in IEEE Std 802.16-2004"
62
63
64
65

```

```

1      ::= { wmanIf2BsRegisteredSsEntry 24 }
2
3
4      --
5      -- wmanIf2BsConfigurationTable contains global parameters common in BS
6      --
7      wmanIf2BsConfigurationTable OBJECT-TYPE
8          SYNTAX      SEQUENCE OF WmanIf2BsConfigurationEntry
9          MAX-ACCESS   not-accessible
10         STATUS      current
11         DESCRIPTION
12             "This table provides one row for each BS sector that
13              contains the BS system parameters as defined in Subclause
14              10.1 of [3]. The objects in this table define the default
15              behaviour of the BS for 2nd Management connection
16              scheduling and SFID allocation as well as configuration
17              parameters of the CPS scheduler and AAS system."
18         REFERENCE
19             "Subclause 10.1 in IEEE Std 802.16-2004"
20         ::= { wmanIf2BsCps 2 }
21
22
23
24
25      wmanIf2BsConfigurationEntry OBJECT-TYPE
26          SYNTAX      WmanIf2BsConfigurationEntry
27          MAX-ACCESS   not-accessible
28          STATUS      current
29          DESCRIPTION
30              "This table is indexed by ifIndex with an ifType of
31              propBWA2Mp."
32          INDEX { ifIndex }
33          ::= { wmanIf2BsConfigurationTable 1 }
34
35
36
37      WmanIf2BsConfigurationEntry ::= SEQUENCE {
38          wmanIf2BsDcdInterval          INTEGER,
39          wmanIf2BsUcdInterval          INTEGER,
40          wmanIf2BsUcdTransition        INTEGER,
41          wmanIf2BsDcdTransition        INTEGER,
42          wmanIf2BsInitialRangingInterval INTEGER,
43          wmanIf2BsSsULMapProcTime      Unsigned32,
44          wmanIf2BsSsRangRespProcTime    Unsigned32,
45          wmanIf2BsT5Timeout             INTEGER,
46          wmanIf2BsT9Timeout             INTEGER,
47          wmanIf2BsT13Timeout            INTEGER,
48          wmanIf2BsT15Timeout            INTEGER,
49          wmanIf2BsT17Timeout            INTEGER,
50          wmanIf2BsT27IdleTimer          Unsigned32,
51          wmanIf2BsT27ActiveTimer        Unsigned32,
52          wmanIf2Bs2ndMgmtDlQoSProfileIndex INTEGER,
53          wmanIf2Bs2ndMgmtUlQoSProfileIndex INTEGER,
54          wmanIf2BsAutoSfidEnabled       INTEGER,
55          wmanIf2BsAutoSfidRangeMin      Unsigned32,
56          wmanIf2BsAutoSfidRangeMax      Unsigned32,
57          wmanIf2BsAasChanFbckReqFreq    INTEGER,
58          wmanIf2BsAasBeamSelectFreq     INTEGER,
59          wmanIf2BsAasChanFbckReqResolution INTEGER,
60          wmanIf2BsAasBeamReqResolution INTEGER,
61
62
63
64
65

```

```

1          wmanIf2BsAasNumOptDiversityZones          INTEGER,
2          wmanIf2BsResetSector                      INTEGER}
3
4
5  wmanIf2BsDcdInterval OBJECT-TYPE
6      SYNTAX      INTEGER (0..10000)
7      UNITS       "milliseconds"
8      MAX-ACCESS  read-write
9      STATUS      current
10     DESCRIPTION
11         "Time between transmission of DCD messages in ms."
12         ::= { wmanIf2BsConfigurationEntry 1 }
13
14
15  wmanIf2BsUcdInterval OBJECT-TYPE
16      SYNTAX      INTEGER (0..10000)
17      UNITS       "milliseconds"
18      MAX-ACCESS  read-write
19      STATUS      current
20      DESCRIPTION
21         "Time between transmission of UCD messages in ms."
22         ::= { wmanIf2BsConfigurationEntry 2 }
23
24
25
26  wmanIf2BsUcdTransition OBJECT-TYPE
27      SYNTAX      INTEGER (2..65535)
28      UNITS       "Number of MAC Frames"
29      MAX-ACCESS  read-write
30      STATUS      current
31      DESCRIPTION
32         "The time the BS shall wait after transmitting a UCD message
33         with an incremented Configuration Change Count before
34         issuing a UL-MAP message referring to
35         Uplink_Burst_Profiles defined in that UCD message."
36         ::= { wmanIf2BsConfigurationEntry 3 }
37
38
39
40
41  wmanIf2BsDcdTransition OBJECT-TYPE
42      SYNTAX      INTEGER (2..65535)
43      UNITS       "Number of MAC Frames"
44      MAX-ACCESS  read-write
45      STATUS      current
46      DESCRIPTION
47         "The time the BS shall wait after transmitting a DCD message
48         with an incremented Configuration Change Count before
49         issuing a DL-MAP message referring to
50         Downlink_Burst_Profiles defined in that DCD message."
51         ::= { wmanIf2BsConfigurationEntry 4 }
52
53
54
55  wmanIf2BsInitialRangingInterval OBJECT-TYPE
56      SYNTAX      INTEGER(0..2000)
57      UNITS       "milliseconds"
58      MAX-ACCESS  read-write
59      STATUS      current
60      DESCRIPTION
61         "Time between Initial Ranging regions assigned by the BS
62         in ms."
63         ::= { wmanIf2BsConfigurationEntry 5 }
64
65

```

```

1
2 wmanIf2BsSsULMapProcTime OBJECT-TYPE
3     SYNTAX      Unsigned32 (200 .. 4294967295)
4     UNITS       "micro seconds"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "Time provided between arrival of the last bit of a UL-MAP
9         at an SS and effectiveness of that map in us."
10    ::= { wmanIf2BsConfigurationEntry 6 }
11
12
13
14 wmanIf2BsSsRangRespProcTime OBJECT-TYPE
15     SYNTAX      Unsigned32 (10000 .. 4294967295)
16     UNITS       "micro seconds"
17     MAX-ACCESS  read-write
18     STATUS      current
19     DESCRIPTION
20         "Time allowed for an SS following receipt of a ranging
21         response before it is expected to reply to an invited
22         ranging request in us."
23    ::= { wmanIf2BsConfigurationEntry 7 }
24
25
26
27 wmanIf2BsT5Timeout OBJECT-TYPE
28     SYNTAX      INTEGER (0 .. 2000)
29     UNITS       "milliseconds"
30     MAX-ACCESS  read-write
31     STATUS      current
32     DESCRIPTION
33         "Wait for Uplink Channel Change Response in ms."
34    ::= { wmanIf2BsConfigurationEntry 8 }
35
36
37
38 wmanIf2BsT9Timeout OBJECT-TYPE
39     SYNTAX      INTEGER (300 .. 65535)
40     UNITS       "milliseconds"
41     MAX-ACCESS  read-write
42     STATUS      current
43     DESCRIPTION
44         "Registration Timeout, the time allowed between the BS
45         sending a RNG-RSP (success) to an SS, and receiving a
46         SBC-REQ from that same SS in ms."
47    ::= { wmanIf2BsConfigurationEntry 9 }
48
49
50
51 wmanIf2BsT13Timeout OBJECT-TYPE
52     SYNTAX      INTEGER (15 .. 65535)
53     UNITS       "minutes"
54     MAX-ACCESS  read-write
55     STATUS      current
56     DESCRIPTION
57         "The time allowed for an SS, following receipt of a
58         REG-RSP message to send a TFTP-CPLT message to the BS
59         in min."
60    ::= { wmanIf2BsConfigurationEntry 10 }
61
62
63
64 wmanIf2BsT15Timeout OBJECT-TYPE
65

```

```

1      SYNTAX      INTEGER (20 .. 65535)
2      UNITS       "milliseconds"
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "Wait for MCA-RSP in ms."
7      ::= { wmanIf2BsConfigurationEntry 11 }
8
9
10
11  wmanIf2BsT17Timeout OBJECT-TYPE
12      SYNTAX      INTEGER (5 .. 65535)
13      UNITS       "minutes"
14      MAX-ACCESS  read-write
15      STATUS      current
16      DESCRIPTION
17          "Time allowed for SS to complete SS Authorization and
18           Key Exchange in minutes."
19      ::= { wmanIf2BsConfigurationEntry 12 }
20
21
22
23  wmanIf2BsT27IdleTimer OBJECT-TYPE
24      SYNTAX      Unsigned32 (10000 .. 4294967295)
25      UNITS       "us"
26      MAX-ACCESS  read-write
27      STATUS      current
28      DESCRIPTION
29          "Maximum time between unicast grants to SS when BS believes
30           SS uplink transmission quality is good enough."
31      ::= { wmanIf2BsConfigurationEntry 13 }
32
33
34
35  wmanIf2BsT27ActiveTimer OBJECT-TYPE
36      SYNTAX      Unsigned32 (10000 .. 4294967295)
37      UNITS       "us"
38      MAX-ACCESS  read-write
39      STATUS      current
40      DESCRIPTION
41          "Maximum time between unicast grants to SS when BS believes
42           SS uplink transmission quality is not good enough."
43      ::= { wmanIf2BsConfigurationEntry 14 }
44
45
46
47
48  wmanIf2Bs2ndMgmtDlQoSProfileIndex OBJECT-TYPE
49      SYNTAX      INTEGER (1..65535)
50      MAX-ACCESS  read-write
51      STATUS      current
52      DESCRIPTION
53          "This object defines the index of a row in
54           wmanIf2BsServiceClassTable which is used to obtain all QoS
55           parameters required for the BS downlink scheduler to
56           properly allocate and manage the bandwidth and schedule
57           the 2nd Management Connection traffic. The 2nd Management
58           Connection traffic doesn't differ from Traffic Connection
59           traffic in the area of QoS management."
60      ::= { wmanIf2BsConfigurationEntry 15 }
61
62
63
64  wmanIf2Bs2ndMgmtUlQoSProfileIndex OBJECT-TYPE
65

```

```

1      SYNTAX      INTEGER (1..65535)
2      MAX-ACCESS  read-write
3      STATUS      current
4      DESCRIPTION
5
6          "This object defines the index of a row in
7          wmanIf2BsServiceClassTable which is used to obtain all QoS
8          parameters required for the BS uplink scheduler to
9          properly allocate and manage the bandwidth and schedule
10         the 2nd Management Connection traffic. The 2nd Management
11         Connection traffic doesn't differ from Traffic Connection
12         traffic in the area of QoS management."
13
14         ::= { wmanIf2BsConfigurationEntry 16 }
15
16 wmanIf2BsAutoSfidEnabled OBJECT-TYPE
17     SYNTAX      INTEGER {autoSfidDisabled(0),
18                       autoSfidEnabled(1)}
19     MAX-ACCESS  read-write
20     STATUS      current
21     DESCRIPTION
22
23         "This object defines whether the BS is allowed to
24         autonomously allocate SFIDs. When the object is set to
25         autoSfidEnabled, the BS is allowed to autonomously allocate
26         SFIDs from the range of allowed values defined by
27         wmanIf2BsConfigExtAutoSfidRangeMin and
28         wmanIf2BsConfigExtAutoSfidRangeMax. A SF is created
29         autonomously when it has not been provisioned in the
30         wmanIf2BsProvisionedSfTable and may be initiated by either
31         the SS or BS. The BS should always initiate SF creation
32         based on the provisioned Service flows configured in
33         wmanIf2BsProvisionedSfTable."
34
35     REFERENCE
36
37         "Subclause 11.13.1 in IEEE Std 802.16-2004"
38
39         ::= { wmanIf2BsConfigurationEntry 17 }
40
41 wmanIf2BsAutoSfidRangeMin OBJECT-TYPE
42     SYNTAX      Unsigned32 ( 1 .. 4294967295)
43     MAX-ACCESS  read-write
44     STATUS      current
45     DESCRIPTION
46
47         "This object defines the minimum value of the range of SFID
48         values allocated for the BS sector for the purpose of
49         autonomous creation of service flows. This value is used
50         when the object wmanIf2BsAutoSfidEnabled allows
51         autonomous creation of SFIDs."
52
53     REFERENCE
54
55         "Subclause 11.13.1 in IEEE Std 802.16-2004"
56
57         ::= { wmanIf2BsConfigurationEntry 18 }
58
59 wmanIf2BsAutoSfidRangeMax OBJECT-TYPE
60     SYNTAX      Unsigned32 ( 1 .. 4294967295)
61     MAX-ACCESS  read-write
62     STATUS      current
63     DESCRIPTION
64
65         "This object defines the maximum value of the range of SFID

```


values allocated for the BS sector for the purpose of autonomous creation of the service flows. This value is used when the object wmanIf2BsAutoSfidEnabled allows autonomous creation of SFIDs."

REFERENCE

"Subclause 11.13.1 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsConfigurationEntry 19 }

wmanIf2BsAasChanFbckReqFreq OBJECT-TYPE

SYNTAX INTEGER (5..10000)

UNITS "ms"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This object defines AAS channel feedback request frequency. It controls the frequency of downlink beam measurements. The relevant MAC messages are AAS-FBCK-REQ/RSP"

REFERENCE

"Subclause 6.3.2.3.40 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsConfigurationEntry 20 }

wmanIf2BsAasBeamSelectFreq OBJECT-TYPE

SYNTAX INTEGER (5..10000)

UNITS "ms"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This object defines AAS beam select frequency. It controls how often SS issues beam select messages. The relevant MAC message is AAS_Beam_Select"

REFERENCE

"Subclause 6.3.2.3.41 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsConfigurationEntry 21 }

wmanIf2BsAasChanFbckReqResolution OBJECT-TYPE

SYNTAX INTEGER { aasChanFbckRes00(0),
 aasChanFbckRes01(1),
 aasChanFbckRes10(2),
 aasChanFbckRes11(3) }

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This object defines AAS feedback request frequency measurements resolution. It is coded as follows:

aasChanFbckRes00 - every 4th carrier

(-100, -96, -92, .., 100)

aasChanFbckRes01 - every 8th carrier

(-100, -92, -84, .., 100)

aasChanFbckRes10 - every 16th carrier

(-100, -84, -68, .., 100)

aasChanFbckRes11 - every 32th carrier

(-100, -68, -36, .., 100)"

REFERENCE

"Subclause 8.3.6.4 in IEEE Std 802.16-2004"

```

1      ::= { wmanIf2BsConfigurationEntry 22 }
2
3
4  wmanIf2BsAasBeamReqResolution OBJECT-TYPE
5      SYNTAX      INTEGER {aasBeamReqRes000(0),
6                      aasBeamReqRes001(1),
7                      aasBeamReqRes010(2),
8                      aasBeamReqRes011(3),
9                      aasBeamReqRes100(4)}
10
11     MAX-ACCESS   read-write
12     STATUS       current
13     DESCRIPTION
14         "This object defines AAS beam select request resolution
15         parameter. It is coded as follows:
16         aasBeamReqRes000 - every 4th carrier
17         aasBeamReqRes001 - every 8th carrier
18         aasBeamReqRes010 - every 16th carrier
19         aasBeamReqRes011 - every 32th carrier
20         aasBeamReqRes100 - every 64th carrier"
21
22     REFERENCE
23         "Subclause 8.3.6.5 in IEEE Std 802.16-2004"
24     ::= { wmanIf2BsConfigurationEntry 23 }
25
26
27  wmanIf2BsAasNumOptDiversityZones OBJECT-TYPE
28     SYNTAX      INTEGER (0..65535)
29     MAX-ACCESS   read-write
30     STATUS       current
31     DESCRIPTION
32         "This object defines the number of optional diversity zones
33         transmitted in downlink."
34
35     REFERENCE
36         "Figure 209 in IEEE Std 802.16-2004"
37     ::= { wmanIf2BsConfigurationEntry 24 }
38
39
40  wmanIf2BsResetSector OBJECT-TYPE
41     SYNTAX      INTEGER {actionResetSectorNoAction(0),
42                      actionResetSector(1)}
43
44     MAX-ACCESS   read-write
45     STATUS       current
46     DESCRIPTION
47         "This object should be implemented as follows:
48         - When set to actionsResetSector value, instructs BS to
49         reset the sector identified by ifIndex. As a result of
50         this action the Phy and Mac of this sector should be
51         reinitialised.
52         - When set to value different than actionsResetSector it
53         should be ignored
54         - When read it should return actionsResetSectorNoAction"
55     ::= { wmanIf2BsConfigurationEntry 25 }
56
57
58
59
60     --
61     -- Base Station Channel Measurement Table
62     --
63  wmanIf2BsChannelMeasurementTable OBJECT-TYPE
64     SYNTAX      SEQUENCE OF WmanIf2BsChannelMeasurementEntry
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table contains channel measurement information as
5          derived from BS measurement of uplink signal from SS,
6          and the downlink signal as reported from SS using
7          REP-REQ/RSP messages. The table shall be maintained as
8          FIFO to store measurement samples that can be used to
9          create RSSI and CINR histogram report. When the
10         measurement entry for a SS reaches the limit, the oldest
11         entry shall be deleted as the new entry is added to the
12         table."
13
14     REFERENCE
15         "6.3.2.3.33 in IEEE Std 802.16-2004"
16     ::= { wmanIf2BsCps 3 }
17
18 wmanIf2BsChannelMeasurementEntry OBJECT-TYPE
19     SYNTAX      WmanIf2BsChannelMeasurementEntry
20     MAX-ACCESS  not-accessible
21     STATUS      current
22     DESCRIPTION
23         "Each entry in the table contains RSSI and CINR
24         signal quality measurement on signal received from the SS.
25         The primary index is the ifIndex with ifType of propBWA2Mp
26         identifying the BS sector. wmanIf2BsSsMacAddress identifies
27         the SS from which the signal was received.
28         wmanIf2BsChannelDirection is the index to the direction of
29         the channel. wmanIf2BsHistogramIndex is the index to
30         histogram samples. Since there is no time stamp in the
31         table, wmanIf2BsHistogramIndex should be increased
32         monotonically, and wraps around when it reaches the
33         implementation specific limit."
34     INDEX      { ifIndex,
35                 wmanIf2BsSsMacAddress,
36                 wmanIf2BsChannelDirection,
37                 wmanIf2BsHistogramIndex }
38     ::= { wmanIf2BsChannelMeasurementTable 1 }
39
40 WmanIf2BsChannelMeasurementEntry ::= SEQUENCE {
41     wmanIf2BsChannelDirection      INTEGER,
42     wmanIf2BsHistogramIndex        Unsigned32,
43     wmanIf2BsChannelNumber         WmanIf2ChannelNumber,
44     wmanIf2BsStartFrame            INTEGER,
45     wmanIf2BsDuration              INTEGER,
46     wmanIf2BsBasicReport           BITS,
47     wmanIf2BsMeanCinrReport        INTEGER,
48     wmanIf2BsMeanRssiReport        INTEGER,
49     wmanIf2BsStdDeviationCinrReport INTEGER,
50     wmanIf2BsStdDeviationRssiReport INTEGER }
51
52 wmanIf2BsChannelDirection OBJECT-TYPE
53     SYNTAX      INTEGER { downstream(1),
54                          upstream(2) }
55     MAX-ACCESS  not-accessible
56
57
58
59
60
61
62
63
64
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "wmanIf2BsChannelDirection identifies the direction of a
4          a channel where the measurement takes place."
5      ::= { wmanIf2BsChannelMeasurementEntry 1 }
6
7
8      wmanIf2BsHistogramIndex OBJECT-TYPE
9          SYNTAX      Unsigned32 (1 .. 4294967295)
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "wmanIf2BsHistogramIndex identifies the histogram samples
14             in the table for each subscriber station."
15         ::= { wmanIf2BsChannelMeasurementEntry 2 }
16
17
18         wmanIf2BsChannelNumber OBJECT-TYPE
19             SYNTAX      WmanIf2ChannelNumber
20             MAX-ACCESS  read-only
21             STATUS      current
22             DESCRIPTION
23                 "Physical channel number to be reported on is only
24                 applicable to license exempt band. For licensed band,
25                 this parameter should be null."
26             REFERENCE
27                 "Subclause 11.12 in IEEE Std 802.16-2004"
28             ::= { wmanIf2BsChannelMeasurementEntry 3 }
29
30
31         wmanIf2BsStartFrame OBJECT-TYPE
32             SYNTAX      INTEGER (0..65535)
33             MAX-ACCESS  read-only
34             STATUS      current
35             DESCRIPTION
36                 "Frame number in which measurement for this channel
37                 started."
38             REFERENCE
39                 "Subclause 11.12 in IEEE Std 802.16-2004"
40             ::= { wmanIf2BsChannelMeasurementEntry 4 }
41
42
43         wmanIf2BsDuration OBJECT-TYPE
44             SYNTAX      INTEGER (0 .. 16777215)
45             MAX-ACCESS  read-only
46             STATUS      current
47             DESCRIPTION
48                 "Cumulative measurement duration on the channel in
49                 multiples of Ts. For any value exceeding 0xFFFFF,
50                 report 0xFFFFF."
51             REFERENCE
52                 "Subclause 11.12 in IEEE Std 802.16-2004"
53             ::= { wmanIf2BsChannelMeasurementEntry 5 }
54
55
56         wmanIf2BsBasicReport OBJECT-TYPE
57             SYNTAX      BITS {wirelessHuman(0),
58                             unknownTransmission(1),
59                             primaryUser(2),
60

```

```

1           channelNotMeasured(3) }
2
3   MAX-ACCESS    read-only
4   STATUS        current
5   DESCRIPTION
6       "Bit #0: WirelessHUMAN detected on the channel
7       Bit #1: Unknown transmissions detected on the channel
8       Bit #2: Primary User detected on the channel
9       Bit #3: Unmeasured. Channel not measured"
10
11  REFERENCE
12      "Subclause 11.12 in IEEE Std 802.16-2004"
13  ::= { wmanIf2BsChannelMeasurementEntry 6 }
14
15  wmanIf2BsMeanCinrReport OBJECT-TYPE
16      SYNTAX      INTEGER (0 .. 41)
17      UNITS        "dB"
18      MAX-ACCESS  read-only
19      STATUS      current
20      DESCRIPTION
21          "Mean CINR report."
22      REFERENCE
23          "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
24      ::= { wmanIf2BsChannelMeasurementEntry 7 }
25
26  wmanIf2BsMeanRssiReport OBJECT-TYPE
27      SYNTAX      INTEGER (0 .. 83)
28      UNITS        "dBm"
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32          "Mean RSSI report."
33      REFERENCE
34          "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
35      ::= { wmanIf2BsChannelMeasurementEntry 8 }
36
37  wmanIf2BsStdDeviationCinrReport OBJECT-TYPE
38      SYNTAX      INTEGER (0 .. 41)
39      UNITS        "dB"
40      MAX-ACCESS  read-only
41      STATUS      current
42      DESCRIPTION
43          "Standard deviation CINR report."
44      REFERENCE
45          "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
46      ::= { wmanIf2BsChannelMeasurementEntry 9 }
47
48  wmanIf2BsStdDeviationRssiReport OBJECT-TYPE
49      SYNTAX      INTEGER (0 .. 83)
50      UNITS        "dB"
51      MAX-ACCESS  read-only
52      STATUS      current
53      DESCRIPTION
54          "Standard deviation RSSI report."
55      REFERENCE
56          "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"

```

```

1      ::= { wmanIf2BsChannelMeasurementEntry 10 }
2
3
4      --
5      -- Base Station capabilities
6      --
7      wmanIf2BsCapabilities OBJECT IDENTIFIER ::= { wmanIf2BsCps 4 }
8
9
10     wmanIf2BsSsReqCapabilitiesTable OBJECT-TYPE
11         SYNTAX      SEQUENCE OF WmanIf2BsSsReqCapabilitiesEntry
12         MAX-ACCESS  not-accessible
13         STATUS      current
14         DESCRIPTION
15             "This table contains the basic capability information of SSs
16             that have been reported by SSs to BS using RNG-REQ, SBC-REQ
17             and REG-REQ messages. Entries in this table should be
18             created when an SS registers with a BS."
19         ::= { wmanIf2BsCapabilities 1 }
20
21
22
23     wmanIf2BsSsReqCapabilitiesEntry OBJECT-TYPE
24         SYNTAX      WmanIf2BsSsReqCapabilitiesEntry
25         MAX-ACCESS  not-accessible
26         STATUS      current
27         DESCRIPTION
28             "This table provides one row for each SS that has been
29             registered in the BS. This table augments the table
30             wmanIf2BsRegisteredSsTable."
31         AUGMENTS { wmanIf2BsRegisteredSsEntry }
32         ::= { wmanIf2BsSsReqCapabilitiesTable 1 }
33
34
35
36     WmanIf2BsSsReqCapabilitiesEntry ::= SEQUENCE {
37         wmanIf2BsSsReqCapUplinkCidSupport      WmanIf2eNumOfCid,
38         wmanIf2BsSsReqCapArqSupport             WmanIf2ArqSupportType,
39         wmanIf2BsSsReqCapDsxFlowControl         WmanIf2MaxDsxFlowType,
40         wmanIf2BsSsReqCapMacCrcSupport          WmanIf2MacCrcSupport,
41         wmanIf2BsSsReqCapMcaFlowControl         WmanIf2MaxMcaFlowType,
42         wmanIf2BsSsReqCapMcpGroupCidSupport     WmanIf2MaxMcpGroupCid,
43         wmanIf2BsSsReqCapPkmFlowControl        WmanIf2MaxPkmFlowType,
44         wmanIf2BsSsReqCapAuthPolicyControl      WmanIf2AuthPolicyType,
45         wmanIf2BsSsReqCapMaxNumOfSupportedSA    WmanIf2MaxNumOfSaType,
46         wmanIf2BsSsReqCapIpVersion             WmanIf2IpVersionType,
47         wmanIf2BsSsReqCapMacCsSupportBitMap     WmanIf2MacCsBitMap,
48         wmanIf2BsSsReqCapMaxNumOfClassifier     WmanIf2MaxClassifiers,
49         wmanIf2BsSsReqCapPhsSupport            WmanIf2PhsSupportType,
50         wmanIf2BsSsReqCapBandwidthAllocSupport WmanIf2BwAllocSupport,
51         wmanIf2BsSsReqCapPduConstruction       WmanIf2PduConstruction,
52         wmanIf2BsSsReqCapTtgTransitionGap       WmanIf2SsTransitionGap,
53         wmanIf2BsSsReqCapRtgTransitionGap       WmanIf2SsTransitionGap}
54
55
56
57
58
59     wmanIf2BsSsReqCapUplinkCidSupport OBJECT-TYPE
60         SYNTAX      WmanIf2eNumOfCid
61         MAX-ACCESS  read-only
62         STATUS      current
63         DESCRIPTION
64             "This object shows the number of Uplink CIDs the SS can
65

```

```

1      support."
2      ::= { wmanIf2BsSsReqCapabilitiesEntry 1 }
3
4
5  wmanIf2BsSsReqCapArqSupport OBJECT-TYPE
6      SYNTAX      WmanIf2ArqSupportType
7      MAX-ACCESS  read-only
8      STATUS      current
9      DESCRIPTION
10         "This object indicates whether the SS supports ARQ."
11      ::= { wmanIf2BsSsReqCapabilitiesEntry 2 }
12
13
14  wmanIf2BsSsReqCapDsxFowControl OBJECT-TYPE
15      SYNTAX      WmanIf2MaxDsxFowType
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19         "This object specifies the maximum number of concurrent
20         DSA, DSC, or DSD transactions that SS is capable of having
21         outstanding."
22      DEFVAL      { 0 }
23      ::= { wmanIf2BsSsReqCapabilitiesEntry 3 }
24
25
26
27  wmanIf2BsSsReqCapMacCrcSupport OBJECT-TYPE
28      SYNTAX      WmanIf2MacCrcSupport
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32         "This object indicates whether or not the SS supports MAC
33         level CRC."
34      DEFVAL      { macCrcSupport }
35      ::= { wmanIf2BsSsReqCapabilitiesEntry 4 }
36
37
38
39  wmanIf2BsSsReqCapMcaFlowControl OBJECT-TYPE
40      SYNTAX      WmanIf2MaxMcaFlowType
41      MAX-ACCESS  read-only
42      STATUS      current
43      DESCRIPTION
44         "This object specifies the maximum number of concurrent MCA
45         transactions that SS is capable of having outstanding."
46      DEFVAL      { 0 }
47      ::= { wmanIf2BsSsReqCapabilitiesEntry 5 }
48
49
50
51  wmanIf2BsSsReqCapMcpGroupCidSupport OBJECT-TYPE
52      SYNTAX      WmanIf2MaxMcpGroupCid
53      MAX-ACCESS  read-only
54      STATUS      current
55      DESCRIPTION
56         "This object indicates the maximum number of
57         simultaneous Multicast Polling Groups the SS is
58         capable of belonging to."
59      DEFVAL      { 0 }
60      ::= { wmanIf2BsSsReqCapabilitiesEntry 6 }
61
62
63
64  wmanIf2BsSsReqCapPkmFlowControl OBJECT-TYPE
65

```

```

1      SYNTAX      WmanIf2MaxPkmFlowType
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "This object specifies the maximum number of concurrent PKM
6          transactions that SS is capable of having outstanding."
7      DEFVAL      { 0 }
8      ::= { wmanIf2BsSsReqCapabilitiesEntry 7 }
9
10
11
12  wmanIf2BsSsReqCapAuthPolicyControl OBJECT-TYPE
13      SYNTAX      WmanIf2AuthPolicyType
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "This object specifies authorization policy that SS is
18          capable of. A bit value of 0 = 'not supported', 1 =
19          'supported'. If this field is omitted, then both SS and
20          BS shall use the IEEE 802.16 security, constituting X.509
21          digital certificates and the RSA public key encryption
22          algorithm, as authorization policy."
23      DEFVAL      { 0 }
24      ::= { wmanIf2BsSsReqCapabilitiesEntry 8 }
25
26
27
28  wmanIf2BsSsReqCapMaxNumOfSupportedSA OBJECT-TYPE
29      SYNTAX      WmanIf2MaxNumOfSaType
30      MAX-ACCESS  read-only
31      STATUS      current
32      DESCRIPTION
33          "This field specifies the maximum number of supported
34          security associations of the SS."
35      DEFVAL      { 1 }
36      ::= { wmanIf2BsSsReqCapabilitiesEntry 9 }
37
38
39
40  wmanIf2BsSsReqCapIpVersion OBJECT-TYPE
41      SYNTAX      WmanIf2IpVersionType
42      MAX-ACCESS  read-only
43      STATUS      current
44      DESCRIPTION
45          "This object indicates the version of IP used on the 2nd
46          Management Connection. The value should be undefined
47          if the 2nd management CID doesn't exist."
48      DEFVAL      { 0 }
49      ::= { wmanIf2BsSsReqCapabilitiesEntry 10 }
50
51
52
53  wmanIf2BsSsReqCapMacCsSupportBitMap OBJECT-TYPE
54      SYNTAX      WmanIf2MacCsBitMap
55      MAX-ACCESS  read-only
56      STATUS      current
57      DESCRIPTION
58          "This object indicates SS reported set of MAC convergence
59          sublayer support. When a bit is set, it indicates
60          the corresponding CS feature is supported."
61      DEFVAL      { 0 }
62      ::= { wmanIf2BsSsReqCapabilitiesEntry 11 }
63
64
65  wmanIf2BsSsReqCapMaxNumOfClassifier OBJECT-TYPE
66      SYNTAX      WmanIf2MaxClassifiers

```



```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "This object indicates the maximum number of admitted
5          Classifiers that the SS can support."
6      DEFVAL        { 0 }
7      ::= { wmanIf2BsSsReqCapabilitiesEntry 12 }
8
9
10
11  wmanIf2BsSsReqCapPhsSupport OBJECT-TYPE
12      SYNTAX        WmanIf2PhsSupportType
13      MAX-ACCESS    read-only
14      STATUS        current
15      DESCRIPTION
16          "This object indicates indicates the level of SS support
17          for PHS."
18      DEFVAL        { noPhsSupport }
19      ::= { wmanIf2BsSsReqCapabilitiesEntry 13 }
20
21
22
23  wmanIf2BsSsReqCapBandwidthAllocSupport OBJECT-TYPE
24      SYNTAX        WmanIf2BwAllocSupport
25      MAX-ACCESS    read-only
26      STATUS        current
27      DESCRIPTION
28          "This field indicates the bandwidth allocation
29          capabilities of the SS. The usage is defined by
30          WmanIf2BwAllocSupport."
31      ::= { wmanIf2BsSsReqCapabilitiesEntry 14 }
32
33
34
35  wmanIf2BsSsReqCapPduConstruction OBJECT-TYPE
36      SYNTAX        WmanIf2PduConstruction
37      MAX-ACCESS    read-only
38      STATUS        current
39      DESCRIPTION
40          "This field indicates the SS's capabilities for
41          construction and transmission of MAC PDUs. The usage
42          is defined by WmanIf2PduConstruction."
43      ::= { wmanIf2BsSsReqCapabilitiesEntry 15 }
44
45
46
47  wmanIf2BsSsReqCapTtgTransitionGap OBJECT-TYPE
48      SYNTAX        WmanIf2SsTransitionGap
49      UNITS          "us"
50      MAX-ACCESS    read-only
51      STATUS        current
52      DESCRIPTION
53          "This field indicates the SS's transition speed SSTTG
54          for TDD and H-FDD SSs. The usage is defined by
55          WmanIf2SsTransitionGap."
56      ::= { wmanIf2BsSsReqCapabilitiesEntry 16 }
57
58
59
60  wmanIf2BsSsReqCapRtgTransitionGap OBJECT-TYPE
61      SYNTAX        WmanIf2SsTransitionGap
62      UNITS          "us"
63      MAX-ACCESS    read-only
64      STATUS        current
65

```

DESCRIPTION

"This field indicates the SS's transition speed SSRTG for TDD and H-FDD SSs. The usage is defined by WmanIf2SsTransitionGap."

::= { wmanIf2BsSsReqCapabilitiesEntry 17 }

wmanIf2BsSsRspCapabilitiesTable OBJECT-TYPE

SYNTAX SEQUENCE OF WmanIf2BsSsRspCapabilitiesEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table contains the basic capability information of SSs that have been negotiated and agreed between BS and SS via RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages.

This table augments the wmanIf2BsRegisteredSsTable."

REFERENCE

"Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"

::= { wmanIf2BsCapabilities 2 }

wmanIf2BsSsRspCapabilitiesEntry OBJECT-TYPE

SYNTAX WmanIf2BsSsRspCapabilitiesEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table provides one row for each SS that has been registered in the BS. This table augments the wmanIf2BsRegisteredSsTable. "

AUGMENTS { wmanIf2BsRegisteredSsEntry }

::= { wmanIf2BsSsRspCapabilitiesTable 1 }

WmanIf2BsSsRspCapabilitiesEntry ::= SEQUENCE {

wmanIf2BsSsRspCapUplinkCidSupport	WmanIf2eNumOfCid,
wmanIf2BsSsRspCapArqSupport	WmanIf2ArqSupportType,
wmanIf2BsSsRspCapDsxFowControl	WmanIf2MaxDsxFowType,
wmanIf2BsSsRspCapMacCrcSupport	WmanIf2MacCrcSupport,
wmanIf2BsSsRspCapMcaFlowControl	WmanIf2MaxMcaFlowType,
wmanIf2BsSsRspCapMcpGroupCidSupport	WmanIf2MaxMcpGroupCid,
wmanIf2BsSsRspCapPkmFlowControl	WmanIf2MaxPkmFlowType,
wmanIf2BsSsRspCapAuthPolicyControl	WmanIf2AuthPolicyType,
wmanIf2BsSsRspCapMaxNumOfSupportedSA	WmanIf2MaxNumOfSaType,
wmanIf2BsSsRspCapIpVersion	WmanIf2IpVersionType,
wmanIf2BsSsRspCapMacCsSupportBitMap	WmanIf2MacCsBitMap,
wmanIf2BsSsRspCapMaxNumOfClassifier	WmanIf2MaxClassifiers,
wmanIf2BsSsRspCapPhsSupport	WmanIf2PhsSupportType,
wmanIf2BsSsRspCapBandwidthAllocSupport	WmanIf2BwAllocSupport,
wmanIf2BsSsRspCapPduConstruction	WmanIf2PduConstruction,
wmanIf2BsSsRspCapTtgTransitionGap	WmanIf2SsTransitionGap,
wmanIf2BsSsRspCapRtgTransitionGap	WmanIf2SsTransitionGap}

wmanIf2BsSsRspCapUplinkCidSupport OBJECT-TYPE

SYNTAX WmanIf2eNumOfCid

MAX-ACCESS read-only

STATUS current

DESCRIPTION

```

1         "Negotiated number of Uplink CIDs the SS can support."
2         ::= { wmanIf2BsSsRspCapabilitiesEntry 1 }
3
4
5 wmanIf2BsSsRspCapArqSupport OBJECT-TYPE
6     SYNTAX      WmanIf2ArqSupportType
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10
11         "This object indicates whether the SS is allowed to use ARQ
12         as a result of the capabilities negotiation."
13     ::= { wmanIf2BsSsRspCapabilitiesEntry 2 }
14
15
16 wmanIf2BsSsRspCapDsxFowControl OBJECT-TYPE
17     SYNTAX      WmanIf2MaxDsxFowType
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21
22         "Negotiated maximum number of concurrent DSA, DSC, or DSD
23         transactions that may be outstanding."
24     ::= { wmanIf2BsSsRspCapabilitiesEntry 3 }
25
26
27 wmanIf2BsSsRspCapMacCrcSupport OBJECT-TYPE
28     SYNTAX      WmanIf2MacCrcSupport
29     MAX-ACCESS  read-only
30     STATUS      current
31     DESCRIPTION
32
33         "This object indicates whether or not the SS is allowed to
34         use MAC level CRC as a result of the capabilities
35         negotiation."
36     DEFVAL      { macCrcSupport }
37     ::= { wmanIf2BsSsRspCapabilitiesEntry 4 }
38
39
40 wmanIf2BsSsRspCapMcaFlowControl OBJECT-TYPE
41     SYNTAX      WmanIf2MaxMcaFlowType
42     MAX-ACCESS  read-only
43     STATUS      current
44     DESCRIPTION
45
46         "Negotiated maximum number of concurrent
47         MCA transactions that may be outstanding."
48     DEFVAL      { 0 }
49     ::= { wmanIf2BsSsRspCapabilitiesEntry 5 }
50
51
52 wmanIf2BsSsRspCapMcpGroupCidSupport OBJECT-TYPE
53     SYNTAX      WmanIf2MaxMcpGroupCid
54     MAX-ACCESS  read-only
55     STATUS      current
56     DESCRIPTION
57
58         "Negotiated maximum number of simultaneous Multicast
59         Polling Groups the SS is capable of belonging to."
60     DEFVAL      { 0 }
61     ::= { wmanIf2BsSsRspCapabilitiesEntry 6 }
62
63
64 wmanIf2BsSsRspCapPkmFlowControl OBJECT-TYPE
65     SYNTAX      WmanIf2MaxPkmFlowType

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "Negotiated maximum number of concurrent PKM
5          transactions that may be outstanding."
6      DEFVAL        { 0 }
7      ::= { wmanIf2BsSsRspCapabilitiesEntry 7 }
8
9
10
11  wmanIf2BsSsRspCapAuthPolicyControl OBJECT-TYPE
12      SYNTAX          WmanIf2AuthPolicyType
13      MAX-ACCESS      read-only
14      STATUS          current
15      DESCRIPTION
16          "This object specifies negotiated authorization policy.
17          A bit value of 0 = 'not supported', 1 = 'supported'. If
18          this field is omitted, then both SS and BS shall use the
19          IEEE 802.16 security, constituting X.509 digital
20          certificates and the RSA public key encryption
21          algorithm, as authorization policy."
22      ::= { wmanIf2BsSsRspCapabilitiesEntry 8 }
23
24
25
26  wmanIf2BsSsRspCapMaxNumOfSupportedSA OBJECT-TYPE
27      SYNTAX          WmanIf2MaxNumOfSaType
28      MAX-ACCESS      read-only
29      STATUS          current
30      DESCRIPTION
31          "Negotiated maximum number of supported security
32          association of the SS."
33      DEFVAL          { 1 }
34      ::= { wmanIf2BsSsRspCapabilitiesEntry 9 }
35
36
37
38  wmanIf2BsSsRspCapIpVersion OBJECT-TYPE
39      SYNTAX          WmanIf2IpVersionType
40      MAX-ACCESS      read-only
41      STATUS          current
42      DESCRIPTION
43          "Negotiated version of IP used on the 2nd Management
44          Connection. The value should be undefined if the 2nd
45          management CID doesn't exist."
46      ::= { wmanIf2BsSsRspCapabilitiesEntry 10 }
47
48
49
50  wmanIf2BsSsRspCapMacCsSupportBitMap OBJECT-TYPE
51      SYNTAX          WmanIf2MacCsBitMap
52      MAX-ACCESS      read-only
53      STATUS          current
54      DESCRIPTION
55          "Negotiated set of MAC convergence sublayer support.
56          When a bit is set, it indicates the corresponding CS
57          feature is supported."
58      ::= { wmanIf2BsSsRspCapabilitiesEntry 11 }
59
60
61
62  wmanIf2BsSsRspCapMaxNumOfClassifier OBJECT-TYPE
63      SYNTAX          WmanIf2MaxClassifiers
64      MAX-ACCESS      read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "Negotiated maximum number of admitted Classifiers
4              that the SS is allowed to have."
5      DEFVAL      { 0 }
6      ::= { wmanIf2BsSsRspCapabilitiesEntry 12 }
7
8
9
10     wmanIf2BsSsRspCapPhsSupport OBJECT-TYPE
11         SYNTAX      WmanIf2PhsSupportType
12         MAX-ACCESS  read-only
13         STATUS      current
14         DESCRIPTION
15             "This object indicates the negotiated level of PHS
16                 support."
17         DEFVAL      { noPhsSupport }
18         ::= { wmanIf2BsSsRspCapabilitiesEntry 13 }
19
20
21     wmanIf2BsSsRspCapBandwidthAllocSupport OBJECT-TYPE
22         SYNTAX      WmanIf2BwAllocSupport
23         MAX-ACCESS  read-only
24         STATUS      current
25         DESCRIPTION
26             "This field indicates negotiated properties of the SS
27                 for bandwidth allocation purposes. The usage is defined
28                 by WmanIf2BwAllocSupport."
29         ::= { wmanIf2BsSsRspCapabilitiesEntry 14 }
30
31
32
33     wmanIf2BsSsRspCapPduConstruction OBJECT-TYPE
34         SYNTAX      WmanIf2PduConstruction
35         MAX-ACCESS  read-only
36         STATUS      current
37         DESCRIPTION
38             "Specifies negotiated capabilities for construction and
39                 transmission of MAC PDUs. The usage is defined by
40                 WmanIf2PduConstruction."
41         ::= { wmanIf2BsSsRspCapabilitiesEntry 15 }
42
43
44
45     wmanIf2BsSsRspCapTtgTransitionGap OBJECT-TYPE
46         SYNTAX      WmanIf2SsTransitionGap
47         UNITS        "us"
48         MAX-ACCESS  read-only
49         STATUS      current
50         DESCRIPTION
51             "This field indicates the negotiated transition speed
52                 SSTTG for TDD and H-FDD SSs. The usage is defined by
53                 WmanIf2SsTransitionGap."
54         ::= { wmanIf2BsSsRspCapabilitiesEntry 16 }
55
56
57
58     wmanIf2BsSsRspCapRtgTransitionGap OBJECT-TYPE
59         SYNTAX      WmanIf2SsTransitionGap
60         UNITS        "us"
61         MAX-ACCESS  read-only
62         STATUS      current
63         DESCRIPTION
64
65

```

```

1         "This field indicates the negotiated transition speed
2         SSRTG for TDD and H-FDD SSs. The usage is defined by
3         WmanIf2SsTransitionGap."
4         ::= { wmanIf2BsSsRspCapabilitiesEntry 17 }
5
6
7 wmanIf2BsBasicCapabilitiesTable OBJECT-TYPE
8     SYNTAX      SEQUENCE OF WmanIf2BsBasicCapabilitiesEntry
9     MAX-ACCESS  not-accessible
10    STATUS      current
11    DESCRIPTION
12        "This table contains the basic capabilities of the BS as
13        implemented in BS hardware and software. These capabilities
14        along with the configuration for them
15        (wmanIf2BsCapabilitiesConfigTable) are used for negotiation
16        of basic capabilities with SS using RNG-RSP, SBC-RSP and
17        REG-RSP messages. The negotiated capabilities are obtained
18        by interSubclause of SS raw reported capabilities, BS raw
19        capabilities and BS configured capabilities. The objects in
20        the table have read-only access. The table is maintained
21        by BS."
22        ::= { wmanIf2BsCapabilities 3 }
23
24
25
26
27 wmanIf2BsBasicCapabilitiesEntry OBJECT-TYPE
28     SYNTAX      WmanIf2BsBasicCapabilitiesEntry
29     MAX-ACCESS  not-accessible
30     STATUS      current
31     DESCRIPTION
32         "This table provides one row for each BS sector and is
33         indexed by ifIndex."
34     INDEX { ifIndex }
35     ::= { wmanIf2BsBasicCapabilitiesTable 1 }
36
37
38
39
40 WmanIf2BsBasicCapabilitiesEntry ::= SEQUENCE {
41     wmanIf2BsCapUplinkCidSupport      WmanIf2eNumOfCid,
42     wmanIf2BsCapArqSupport            WmanIf2ArqSupportType,
43     wmanIf2BsCapDsxFlowControl        WmanIf2MaxDsxFlowType,
44     wmanIf2BsCapMacCrcSupport         WmanIf2MacCrcSupport,
45     wmanIf2BsCapMcaFlowControl        WmanIf2MaxMcaFlowType,
46     wmanIf2BsCapMcpGroupCidSupport    WmanIf2MaxMcpGroupCid,
47     wmanIf2BsCapPkmFlowControl        WmanIf2MaxPkmFlowType,
48     wmanIf2BsCapAuthPolicyControl     WmanIf2AuthPolicyType,
49     wmanIf2BsCapMaxNumOfSupportedSA   WmanIf2MaxNumOfSaType,
50     wmanIf2BsCapIpVersion             WmanIf2IpVersionType,
51     wmanIf2BsCapMacCsSupportBitMap    WmanIf2MacCsBitMap,
52     wmanIf2BsCapMaxNumOfClassifier    WmanIf2MaxClassifiers,
53     wmanIf2BsCapPhsSupport            WmanIf2PhsSupportType,
54     wmanIf2BsCapBandwidthAllocSupport WmanIf2BwAllocSupport,
55     wmanIf2BsCapPduConstruction       WmanIf2PduConstruction,
56     wmanIf2BsCapTtgTransitionGap      WmanIf2SsTransitionGap,
57     wmanIf2BsCapRtgTransitionGap      WmanIf2SsTransitionGap}
58
59
60
61
62 wmanIf2BsCapUplinkCidSupport OBJECT-TYPE
63     SYNTAX      WmanIf2eNumOfCid
64     MAX-ACCESS  read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This object shows the number of Uplink CIDs the BS can
4          support per SS."
5      ::= { wmanIf2BsBasicCapabilitiesEntry 1 }
6
7
8      wmanIf2BsCapArqSupport OBJECT-TYPE
9          SYNTAX      WmanIf2ArqSupportType
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "This object indicates whether the BS supports ARQ."
14         ::= { wmanIf2BsBasicCapabilitiesEntry 2 }
15
16
17         wmanIf2BsCapDsxFowControl OBJECT-TYPE
18             SYNTAX      WmanIf2MaxDsxFowType
19             MAX-ACCESS  read-only
20             STATUS      current
21             DESCRIPTION
22                 "This object specifies the maximum number of concurrent
23                 DSA, DSC, or DSD transactions that BS allows each SS to
24                 have outstanding."
25             DEFVAL      { 0 }
26             ::= { wmanIf2BsBasicCapabilitiesEntry 3 }
27
28
29         wmanIf2BsCapMacCrcSupport OBJECT-TYPE
30             SYNTAX      WmanIf2MacCrcSupport
31             MAX-ACCESS  read-only
32             STATUS      current
33             DESCRIPTION
34                 "This object indicates whether or not the BS supports MAC
35                 level CRC."
36             DEFVAL      { macCrcSupport }
37             ::= { wmanIf2BsBasicCapabilitiesEntry 4 }
38
39
40         wmanIf2BsCapMcaFlowControl OBJECT-TYPE
41             SYNTAX      WmanIf2MaxMcaFlowType
42             MAX-ACCESS  read-only
43             STATUS      current
44             DESCRIPTION
45                 "This object specifies the maximum number of concurrent
46                 MCA transactions that BS allows each SS to have."
47             DEFVAL      { 0 }
48             ::= { wmanIf2BsBasicCapabilitiesEntry 5 }
49
50
51         wmanIf2BsCapMcpGroupCidSupport OBJECT-TYPE
52             SYNTAX      WmanIf2MaxMcpGroupCid
53             MAX-ACCESS  read-only
54             STATUS      current
55             DESCRIPTION
56                 "This object indicates the maximum number of simultaneous
57                 Multicast Polling Groups the BS allows each SS to belong
58                 to."
59             DEFVAL      { 0 }
60
61
62
63
64
65

```

```

1      ::= { wmanIf2BsBasicCapabilitiesEntry 6 }
2
3
4  wmanIf2BsCapPkmFlowControl OBJECT-TYPE
5      SYNTAX      WmanIf2MaxPkmFlowType
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "This object specifies the maximum number of concurrent
10         PKM transactions that BS allows each SS to have."
11      DEFVAL      { 0 }
12      ::= { wmanIf2BsBasicCapabilitiesEntry 7 }
13
14
15  wmanIf2BsCapAuthPolicyControl OBJECT-TYPE
16      SYNTAX      WmanIf2AuthPolicyType
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20          "This object specifies authorization policy that BS is
21         capable of. A bit value of 0 = 'not supported', 1 =
22         'supported'. If this field is omitted, then both SS and
23         BS shall use the IEEE 802.16 security, constituting X.509
24         digital certificates and the RSA public key encryption
25         algorithm, as authorization policy."
26      ::= { wmanIf2BsBasicCapabilitiesEntry 8 }
27
28
29  wmanIf2BsCapMaxNumOfSupportedSA OBJECT-TYPE
30      SYNTAX      WmanIf2MaxNumOfSaType
31      MAX-ACCESS  read-only
32      STATUS      current
33      DESCRIPTION
34          "This field specifies maximum number of supported security
35         associations per SS that the BS allows."
36      DEFVAL      { 1 }
37      ::= { wmanIf2BsBasicCapabilitiesEntry 9 }
38
39
40  wmanIf2BsCapIpVersion OBJECT-TYPE
41      SYNTAX      WmanIf2IpVersionType
42      MAX-ACCESS  read-only
43      STATUS      current
44      DESCRIPTION
45          "This object indicates the version of IP BS allows each SS
46         to use on the 2nd Management Connection. The value
47         'undefined' should not be used for this field."
48      REFERENCE
49          "Subclause 11.7.4 in IEEE Std 802.16-2004"
50      ::= { wmanIf2BsBasicCapabilitiesEntry 10 }
51
52
53  wmanIf2BsCapMacCsSupportBitMap OBJECT-TYPE
54      SYNTAX      WmanIf2MacCsBitMap
55      MAX-ACCESS  read-only
56      STATUS      current
57      DESCRIPTION
58          "This object indicates BS set of MAC convergence
59         sublayer support. When a bit is set, it indicates
60

```



```

1         the corresponding CS feature is supported."
2         ::= { wmanIf2BsBasicCapabilitiesEntry 11 }
3
4
5 wmanIf2BsCapMaxNumOfClassifier OBJECT-TYPE
6     SYNTAX      WmanIf2MaxClassifiers
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10
11         "This object indicates the maximum number of admitted
12         Classifiers per SS that the BS allows."
13     DEFVAL      { 0 }
14     ::= { wmanIf2BsBasicCapabilitiesEntry 12 }
15
16
17 wmanIf2BsCapPhsSupport OBJECT-TYPE
18     SYNTAX      WmanIf2PhsSupportType
19     MAX-ACCESS  read-only
20     STATUS      current
21     DESCRIPTION
22
23         "This object indicates the level of BS support for PHS.
24         The usage is defined by WmanIf2PhsSupportType."
25     DEFVAL      { noPhsSupport }
26     ::= { wmanIf2BsBasicCapabilitiesEntry 13 }
27
28
29 wmanIf2BsCapBandwidthAllocSupport OBJECT-TYPE
30     SYNTAX      WmanIf2BwAllocSupport
31     MAX-ACCESS  read-only
32     STATUS      current
33     DESCRIPTION
34
35         "This field indicates the bandwidth allocation properties
36         that the BS permits SSs to use. The usage is defined by
37         WmanIf2BwAllocSupport."
38     ::= { wmanIf2BsBasicCapabilitiesEntry 14 }
39
40
41 wmanIf2BsCapPduConstruction OBJECT-TYPE
42     SYNTAX      WmanIf2PduConstruction
43     MAX-ACCESS  read-only
44     STATUS      current
45     DESCRIPTION
46
47         "Specifies the capabilities for construction and
48         transmission of MAC PDUs allowed by the BS. The usage is
49         defined by WmanIf2PduConstruction."
50     ::= { wmanIf2BsBasicCapabilitiesEntry 15 }
51
52
53 wmanIf2BsCapTtgTransitionGap OBJECT-TYPE
54     SYNTAX      WmanIf2SsTransitionGap
55     UNITS        "us"
56     MAX-ACCESS  read-only
57     STATUS      current
58     DESCRIPTION
59
60         "This field indicates the transition speed SSTTG for TDD
61         and H-FDD SSs allowed by the BS. The usage is defined by
62         WmanIf2SsTransitionGap."
63     ::= { wmanIf2BsBasicCapabilitiesEntry 16 }
64
65

```

```

1  wmanIf2BsCapRtgTransitionGap OBJECT-TYPE
2      SYNTAX          WmanIf2SsTransitionGap
3      UNITS            "us"
4      MAX-ACCESS      read-only
5      STATUS          current
6      DESCRIPTION
7          "This field indicates the transition speed SSRTG for TDD
8            and H-FDD SSs allowed by the BS. The usage is defined
9            by WmanIf2SsTransitionGap."
10         ::= { wmanIf2BsBasicCapabilitiesEntry 17 }
11
12  wmanIf2BsCapabilitiesConfigTable OBJECT-TYPE
13      SYNTAX          SEQUENCE OF WmanIf2BsCapabilitiesConfigEntry
14      MAX-ACCESS      not-accessible
15      STATUS          current
16      DESCRIPTION
17          "This table contains the configuration for basic
18            capabilities of BS. The table is intended to be used to
19            restrict the Capabilities implemented by BS, for example in
20            order to comply with local regulatory requirements. The BS
21            should use the configuration along with the implemented
22            Capabilities (wmanIf2BsBasicCapabilitiesTable) for
23            negotiation of basic capabilities with SS using RNG-RSP,
24            SBC-RSP and REG-RSP messages. The negotiated capabilities
25            are obtained by interSubclause of SS reported capabilities,
26            BS raw capabilities and BS configured capabilities. The
27            objects in the table have read-write access. The rows are
28            created by BS as a copy of wmanIf2BsBasicCapabilitiesTable
29            and can be modified by NMS."
30         ::= { wmanIf2BsCapabilities 4 }
31
32  wmanIf2BsCapabilitiesConfigEntry OBJECT-TYPE
33      SYNTAX          WmanIf2BsCapabilitiesConfigEntry
34      MAX-ACCESS      not-accessible
35      STATUS          current
36      DESCRIPTION
37          "This table provides one row for each BS sector and is
38            indexed by ifIndex."
39      INDEX { ifIndex }
40      ::= { wmanIf2BsCapabilitiesConfigTable 1 }
41
42  WmanIf2BsCapabilitiesConfigEntry ::= SEQUENCE {
43      wmanIf2BsCapCfgUplinkCidSupport      WmanIf2eNumOfCid,
44      wmanIf2BsCapCfgArqSupport            WmanIf2ArqSupportType,
45      wmanIf2BsCapCfgDsxFlowControl        WmanIf2MaxDsxFlowType,
46      wmanIf2BsCapCfgMacCrcSupport         WmanIf2MacCrcSupport,
47      wmanIf2BsCapCfgMcaFlowControl        WmanIf2MaxMcaFlowType,
48      wmanIf2BsCapCfgMcpGroupCidSupport    WmanIf2MaxMcpGroupCid,
49      wmanIf2BsCapCfgPkmFlowControl        WmanIf2MaxPkmFlowType,
50      wmanIf2BsCapCfgAuthPolicyControl     WmanIf2AuthPolicyType,
51      wmanIf2BsCapCfgMaxNumOfSupportedSA   WmanIf2MaxNumOfSaType,
52      wmanIf2BsCapCfgIpVersion             WmanIf2IpVersionType,
53      wmanIf2BsCapCfgMacCsSupportBitMap    WmanIf2MacCsBitMap,
54      wmanIf2BsCapCfgMaxNumOfClassifier    WmanIf2MaxClassifiers,

```

```

1      wmanIf2BsCapCfgPhsSupport                WmanIf2PhsSupportType,
2      wmanIf2BsCapCfgBandwidthAllocSupport     WmanIf2BwAllocSupport,
3      wmanIf2BsCapCfgPduConstruction           WmanIf2PduConstruction,
4      wmanIf2BsCapCfgTtgTransitionGap           WmanIf2SsTransitionGap,
5      wmanIf2BsCapCfgRtgTransitionGap           WmanIf2SsTransitionGap}
6
7
8      wmanIf2BsCapCfgUplinkCidSupport OBJECT-TYPE
9          SYNTAX      WmanIf2eNumOfCid
10         MAX-ACCESS  read-write
11         STATUS      current
12         DESCRIPTION
13             "This object shows the configured number of Uplink CIDs the
14             BS can support per SS."
15         ::= { wmanIf2BsCapabilitiesConfigEntry 1 }
16
17
18      wmanIf2BsCapCfgArqSupport OBJECT-TYPE
19          SYNTAX      WmanIf2ArqSupportType
20          MAX-ACCESS  read-write
21          STATUS      current
22          DESCRIPTION
23              "This object indicates whether the BS is configured to
24              support ARQ."
25          ::= { wmanIf2BsCapabilitiesConfigEntry 2 }
26
27
28      wmanIf2BsCapCfgDsxFowControl OBJECT-TYPE
29          SYNTAX      WmanIf2MaxDsxFowType
30          MAX-ACCESS  read-write
31          STATUS      current
32          DESCRIPTION
33              "This object specifies the configured maximum number of
34              concurrent DSA, DSC, or DSD transactions that BS allows
35              each SS to have outstanding."
36          DEFVAL      { 0 }
37          ::= { wmanIf2BsCapabilitiesConfigEntry 3 }
38
39
40      wmanIf2BsCapCfgMacCrcSupport OBJECT-TYPE
41          SYNTAX      WmanIf2MacCrcSupport
42          MAX-ACCESS  read-write
43          STATUS      current
44          DESCRIPTION
45              "This object indicates whether BS is configured to support
46              MAC level CRC."
47          DEFVAL      { macCrcSupport }
48          ::= { wmanIf2BsCapabilitiesConfigEntry 4 }
49
50
51      wmanIf2BsCapCfgMcaFlowControl OBJECT-TYPE
52          SYNTAX      WmanIf2MaxMcaFlowType
53          MAX-ACCESS  read-write
54          STATUS      current
55          DESCRIPTION
56              "This object specifies the maximum number of concurrent
57              MCA transactions that BS is configured to allow each SS to
58              have."
59          DEFVAL      { 0 }
60
61
62
63
64
65

```

```

1      ::= { wmanIf2BsCapabilitiesConfigEntry 5 }
2
3
4  wmanIf2BsCapCfgMcpGroupCidSupport OBJECT-TYPE
5      SYNTAX      WmanIf2MaxMcpGroupCid
6      MAX-ACCESS  read-write
7      STATUS      current
8      DESCRIPTION
9          "This object indicates the maximum number of simultaneous
10         Multicast Polling Groups the BS is configured to allow
11         each SS to belong to."
12
13     DEFVAL      { 0 }
14     ::= { wmanIf2BsCapabilitiesConfigEntry 6 }
15
16
17  wmanIf2BsCapCfgPkmFlowControl OBJECT-TYPE
18      SYNTAX      WmanIf2MaxPkmFlowType
19      MAX-ACCESS  read-write
20      STATUS      current
21      DESCRIPTION
22          "This object specifies the maximum number of concurrent
23         PKM transactions that BS is configured to allow each SS
24         to have."
25
26     DEFVAL      { 0 }
27     ::= { wmanIf2BsCapabilitiesConfigEntry 7 }
28
29
30  wmanIf2BsCapCfgAuthPolicyControl OBJECT-TYPE
31      SYNTAX      WmanIf2AuthPolicyType
32      MAX-ACCESS  read-write
33      STATUS      current
34      DESCRIPTION
35          "This object specifies authorization policy that BS is
36         configured to be capable of. A bit value of 0 = 'not
37         supported', 1 = 'supported'. If this field is omitted,
38         then both SS and BS shall use the IEEE 802.16 security,
39         constituting X.509 digital certificates and the RSA
40         public key encryption algorithm, as authorization policy."
41
42     ::= { wmanIf2BsCapabilitiesConfigEntry 8 }
43
44
45  wmanIf2BsCapCfgMaxNumOfSupportedSA OBJECT-TYPE
46      SYNTAX      WmanIf2MaxNumOfSaType
47      MAX-ACCESS  read-write
48      STATUS      current
49      DESCRIPTION
50          "This field specifies configured maximum number of supported
51         security association per SS."
52
53     DEFVAL      { 1 }
54     ::= { wmanIf2BsCapabilitiesConfigEntry 9 }
55
56
57  wmanIf2BsCapCfgIpVersion OBJECT-TYPE
58      SYNTAX      WmanIf2IpVersionType
59      MAX-ACCESS  read-write
60      STATUS      current
61      DESCRIPTION
62          "This object indicates the configured version of IP that the
63         BS allows each SS to use on the 2nd Management Connection."
64
65

```

```

1         The value 'undefined' should not be used in this field."
2     ::= { wmanIf2BsCapabilitiesConfigEntry 10 }
3
4
5 wmanIf2BsCapCfgMacCsSupportBitMap OBJECT-TYPE
6     SYNTAX      WmanIf2MacCsBitMap
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "This object indicates BS configured set of MAC convergence
11        sublayer support. When a bit is set, it indicates
12        the corresponding CS feature is supported."
13    ::= { wmanIf2BsCapabilitiesConfigEntry 11 }
14
15
16 wmanIf2BsCapCfgMaxNumOfClassifier OBJECT-TYPE
17     SYNTAX      WmanIf2MaxClassifiers
18     MAX-ACCESS  read-write
19     STATUS      current
20     DESCRIPTION
21        "This object indicates the configured maximum number of
22        admitted Classifiers per SS that the BS can support."
23    DEFVAL      { 0 }
24    ::= { wmanIf2BsCapabilitiesConfigEntry 12 }
25
26
27
28 wmanIf2BsCapCfgPhsSupport OBJECT-TYPE
29     SYNTAX      WmanIf2PhsSupportType
30     MAX-ACCESS  read-write
31     STATUS      current
32     DESCRIPTION
33        "This object indicates the configured level of BS support
34        for PHS."
35    DEFVAL      { noPhsSupport }
36    ::= { wmanIf2BsCapabilitiesConfigEntry 13 }
37
38
39
40 wmanIf2BsCapCfgBandwidthAllocSupport OBJECT-TYPE
41     SYNTAX      WmanIf2BwAllocSupport
42     MAX-ACCESS  read-write
43     STATUS      current
44     DESCRIPTION
45        "This field indicates configured properties of the BS for
46        bandwidth allocation purposes. The usage is defined by
47        WmanIf2CapBwAllocSupport."
48    ::= { wmanIf2BsCapabilitiesConfigEntry 14 }
49
50
51
52 wmanIf2BsCapCfgPduConstruction OBJECT-TYPE
53     SYNTAX      WmanIf2PduConstruction
54     MAX-ACCESS  read-write
55     STATUS      current
56     DESCRIPTION
57        "Specifies configured capabilities for construction and
58        transmission of MAC PDUs. The usage is defined by
59        WmanIf2PduConstruction."
60    ::= { wmanIf2BsCapabilitiesConfigEntry 15 }
61
62
63
64 wmanIf2BsCapCfgTtgTransitionGap OBJECT-TYPE
65

```

```

1      SYNTAX      WmanIf2SsTransitionGap
2      UNITS       "us"
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "This field indicates the configured transition speed
7          SSTTG for TDD and H-FDD SSs. The usage is defined by
8          WmanIf2SsTransitionGap."
9      ::= { wmanIf2BsCapabilitiesConfigEntry 16 }
10
11
12
13  wmanIf2BsCapCfgRtgTransitionGap OBJECT-TYPE
14      SYNTAX      WmanIf2SsTransitionGap
15      UNITS       "us"
16      MAX-ACCESS  read-write
17      STATUS      current
18      DESCRIPTION
19          "This field indicates the configured transition speed
20          SSRTG for TDD and H-FDD SSs. The usage is defined by
21          WmanIf2SsTransitionGap."
22      ::= { wmanIf2BsCapabilitiesConfigEntry 17 }
23
24
25
26  wmanIf2eBsSsReqCapabilitiesTable OBJECT-TYPE
27      SYNTAX      SEQUENCE OF WmanIf2eBsSsReqCapabilitiesEntry
28      MAX-ACCESS  not-accessible
29      STATUS      current
30      DESCRIPTION
31          "This table augments wmanIf2BsRegisteredSsTable to include
32          new capabilities as introduced in IEEE 802.16e 2005
33          standard."
34      ::= { wmanIf2BsCapabilities 5 }
35
36
37
38  wmanIf2eBsSsReqCapabilitiesEntry OBJECT-TYPE
39      SYNTAX      WmanIf2eBsSsReqCapabilitiesEntry
40      MAX-ACCESS  not-accessible
41      STATUS      current
42      DESCRIPTION
43          "This table provides one row for each MS that has been
44          registered in the BS. This table augments the table
45          wmanIf2BsRegisteredSsTable."
46      AUGMENTS { wmanIf2BsRegisteredSsEntry }
47      ::= { wmanIf2eBsSsReqCapabilitiesTable 1 }
48
49
50
51  WmanIf2eBsSsReqCapabilitiesEntry ::= SEQUENCE {
52      wmanIf2eBsSsReqCapDownlinkCidSupport  WmanIf2eNumOfCid,
53      wmanIf2eBsSsReqCapPackingSupport      WmanIf2ePackingSupport,
54      wmanIf2eBsSsReqCapExtendedRtpsSupport WmanIf2eExtRtpsSupport,
55      wmanIf2eBsSsReqCapMaxNumBurstToMs    INTEGER,
56      wmanIf2eBsSsReqCapIpAddrAllocMethod  WmanIf2eIpAllocMethod,
57      wmanIf2eBsSsReqCapHandoverSupported  WmanIf2eHandoverType,
58      wmanIf2eBsSsReqCapHoProcessTimer     Unsigned32,
59      wmanIf2eBsSsReqCapIdleModeTimeout    Unsigned32,
60      wmanIf2eBsSsReqCapArqAckType         WmanIf2eArqAckType,
61      wmanIf2eBsSsReqCapMacHeader          WmanIf2eMacHeaderSupp}
62
63
64
65

```

```

1  wmanIf2eBsSsReqCapDownlinkCidSupport OBJECT-TYPE
2      SYNTAX          WmanIf2eNumOfCid
3      MAX-ACCESS      read-only
4      STATUS          current
5      DESCRIPTION
6          "This object shows the number of Downlink CIDs the SS can
7          support."
8      ::= { wmanIf2eBsSsReqCapabilitiesEntry 1 }
9
10
11  wmanIf2eBsSsReqCapPackingSupport OBJECT-TYPE
12      SYNTAX          WmanIf2ePackingSupport
13      MAX-ACCESS      read-only
14      STATUS          current
15      DESCRIPTION
16          "Indicates the availability of MS support for Packing."
17      REFERENCE
18          "Subclause 11.7.8.11 in IEEE Std 802.16e-2005"
19      ::= { wmanIf2eBsSsReqCapabilitiesEntry 2 }
20
21
22  wmanIf2eBsSsReqCapExtendedRtptsSupport OBJECT-TYPE
23      SYNTAX          WmanIf2eExtRtptsSupport
24      MAX-ACCESS      read-only
25      STATUS          current
26      DESCRIPTION
27          "Indicates the availability of MS support for extended
28          rtPs."
29      REFERENCE
30          "Subclause 11.7.8.12 in IEEE Std 802.16e-2005"
31      ::= { wmanIf2eBsSsReqCapabilitiesEntry 3 }
32
33
34  wmanIf2eBsSsReqCapMaxNumBurstToMs OBJECT-TYPE
35      SYNTAX          INTEGER (1..16)
36      MAX-ACCESS      read-only
37      STATUS          current
38      DESCRIPTION
39          "Maximum number of bursts transmitted concurrently to the MS
40          , including all bursts without CID or with CIDs matching
41          the MS's CIDs."
42      REFERENCE
43          "Subclause 11.7.8.13 in IEEE Std 802.16e-2005"
44      ::= { wmanIf2eBsSsReqCapabilitiesEntry 4 }
45
46
47  wmanIf2eBsSsReqCapIpAddrAllocMethod OBJECT-TYPE
48      SYNTAX          WmanIf2eIpAllocMethod
49      MAX-ACCESS      read-only
50      STATUS          current
51      DESCRIPTION
52          "Indicates the method of allocating IP address for the
53          secondary management connection."
54      REFERENCE
55          "Subclause 11.7.11 in IEEE Std 802.16e-2005"
56      ::= { wmanIf2eBsSsReqCapabilitiesEntry 5 }
57
58
59  wmanIf2eBsSsReqCapHandoverSupported OBJECT-TYPE
60
61
62
63
64
65

```

```

1      SYNTAX      WmanIf2eHandoverType
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "Indicates what type(s) of Handover the BS or MS supports."
6      REFERENCE
7          "Subclause 11.7.12 in IEEE Std 802.16e-2005"
8      ::= { wmanIf2eBsSsReqCapabilitiesEntry 6 }
9
10
11
12  wmanIf2eBsSsReqCapHoProcessTimer OBJECT-TYPE
13      SYNTAX      Unsigned32
14      UNITS        "frames"
15      MAX-ACCESS  read-only
16      STATUS      current
17      DESCRIPTION
18          "The duration in frames the MS shall wait until receipt of
19           the next unsolicited network re-entry MAC management
20           message as indicated in the HO Process Optimization
21           element of the RNG-RSP message."
22      REFERENCE
23          "Subclause 11.7.13.2 in IEEE Std 802.16e-2005"
24      ::= { wmanIf2eBsSsReqCapabilitiesEntry 7 }
25
26
27
28  wmanIf2eBsSsReqCapIdleModeTimeout OBJECT-TYPE
29      SYNTAX      Unsigned32
30      UNITS        "seconds"
31      MAX-ACCESS  read-only
32      STATUS      current
33      DESCRIPTION
34          "Max time interval between MS Idle Mode Location Updates."
35      REFERENCE
36          "Subclause 11.7.20.1 in IEEE Std 802.16e-2005"
37      DEFVAL      { 4096 }
38      ::= { wmanIf2eBsSsReqCapabilitiesEntry 8 }
39
40
41
42  wmanIf2eBsSsReqCapArqAckType OBJECT-TYPE
43      SYNTAX      WmanIf2eArqAckType
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47          "The value of this parameter specifies the ARQ ACK type
48           supported by the MS."
49      REFERENCE
50          "Subclause 11.7.23 in IEEE Std 802.16e-2005"
51      ::= { wmanIf2eBsSsReqCapabilitiesEntry 9 }
52
53
54
55  wmanIf2eBsSsReqCapMacHeader OBJECT-TYPE
56      SYNTAX      WmanIf2eMacHeaderSupp
57      MAX-ACCESS  read-only
58      STATUS      current
59      DESCRIPTION
60          "Indicates whether or not the MS and BS support various
61           types of MAC header and extended subheaders."
62      REFERENCE
63
64
65

```



```

1         "Subclause 11.7.25 in IEEE Std 802.16e-2005"
2         ::= { wmanIf2eBsSsReqCapabilitiesEntry 10 }
3
4
5 wmanIf2eBsSsRspCapabilitiesTable OBJECT-TYPE
6     SYNTAX      SEQUENCE OF WmanIf2eBsSsRspCapabilitiesEntry
7     MAX-ACCESS  not-accessible
8     STATUS      current
9     DESCRIPTION
10
11         "This table contains the basic capability information of SSs
12         that have been negotiated and agreed between BS and SS via
13         RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages.
14         This table augments the wmanIf2BsRegisteredSsTable."
15     REFERENCE
16
17         "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
18     ::= { wmanIf2BsCapabilities 6 }
19
20 wmanIf2eBsSsRspCapabilitiesEntry OBJECT-TYPE
21     SYNTAX      WmanIf2eBsSsRspCapabilitiesEntry
22     MAX-ACCESS  not-accessible
23     STATUS      current
24     DESCRIPTION
25
26         "This table provides one row for each SS that has been
27         registered in the BS. This table augments the
28         wmanIf2BsRegisteredSsTable. "
29     AUGMENTS { wmanIf2BsRegisteredSsEntry }
30     ::= { wmanIf2eBsSsRspCapabilitiesTable 1 }
31
32
33 WmanIf2eBsSsRspCapabilitiesEntry ::= SEQUENCE {
34     wmanIf2eBsSsRspCapDownlinkCidSupport  WmanIf2eNumOfCid,
35     wmanIf2eBsSsRspCapPackingSupport      WmanIf2ePackingSupport,
36     wmanIf2eBsSsRspCapExtendedRtpsSupport WmanIf2eExtRtpsSupport,
37     wmanIf2eBsSsRspCapMaxNumBurstToMs     INTEGER,
38     wmanIf2eBsSsRspCapIpAddrAllocMethod   WmanIf2eIpAllocMethod,
39     wmanIf2eBsSsRspCapHandoverSupported   WmanIf2eHandoverType,
40     wmanIf2eBsSsRspCapHoProcessTimer      Unsigned32,
41     wmanIf2eBsSsRspCapIdleModeTimeout     Unsigned32,
42     wmanIf2eBsSsRspCapArqAckType          WmanIf2eArqAckType,
43     wmanIf2eBsSsRspCapMacHeader           WmanIf2eMacHeaderSupp}
44
45
46 wmanIf2eBsSsRspCapDownlinkCidSupport OBJECT-TYPE
47     SYNTAX      WmanIf2eNumOfCid
48     MAX-ACCESS  read-only
49     STATUS      current
50     DESCRIPTION
51
52         "This object shows the number of Downlink CIDs the SS can
53         support."
54     ::= { wmanIf2eBsSsRspCapabilitiesEntry 1 }
55
56
57 wmanIf2eBsSsRspCapPackingSupport OBJECT-TYPE
58     SYNTAX      WmanIf2ePackingSupport
59     MAX-ACCESS  read-only
60     STATUS      current
61     DESCRIPTION
62
63         "Indicates the availability of MS support for Packing."
64
65

```

```

1      REFERENCE
2          "Subclause 11.7.8.11 in IEEE Std 802.16e-2005"
3      ::= { wmanIf2eBsSsRspCapabilitiesEntry 2 }
4
5
6      wmanIf2eBsSsRspCapExtendedRtPsSupport OBJECT-TYPE
7          SYNTAX      WmanIf2eExtRtPsSupport
8          MAX-ACCESS  read-only
9          STATUS      current
10         DESCRIPTION
11             "Indicates the availability of MS support for extended
12              rtPs."
13         REFERENCE
14             "Subclause 11.7.8.12 in IEEE Std 802.16e-2005"
15         ::= { wmanIf2eBsSsRspCapabilitiesEntry 3 }
16
17
18
19      wmanIf2eBsSsRspCapMaxNumBurstToMs OBJECT-TYPE
20          SYNTAX      INTEGER (1..16)
21          MAX-ACCESS  read-only
22          STATUS      current
23          DESCRIPTION
24              "Maximum number of bursts transmitted concurrently to the MS
25               , including all bursts without CID or with CIDs matching
26               the MS CIDs."
27          REFERENCE
28              "Subclause 11.7.8.13 in IEEE Std 802.16e-2005"
29          ::= { wmanIf2eBsSsRspCapabilitiesEntry 4 }
30
31
32
33      wmanIf2eBsSsRspCapIpAddrAllocMethod OBJECT-TYPE
34          SYNTAX      WmanIf2eIpAllocMethod
35          MAX-ACCESS  read-only
36          STATUS      current
37          DESCRIPTION
38              "Indicates the method of allocating IP address for the
39               secondary management connection."
40          REFERENCE
41              "Subclause 11.7.11 in IEEE Std 802.16e-2005"
42          ::= { wmanIf2eBsSsRspCapabilitiesEntry 5 }
43
44
45
46
47      wmanIf2eBsSsRspCapHandoverSupported OBJECT-TYPE
48          SYNTAX      WmanIf2eHandoverType
49          MAX-ACCESS  read-only
50          STATUS      current
51          DESCRIPTION
52              "Indicates what type(s) of Handover the BS or MS supports."
53          REFERENCE
54              "Subclause 11.7.12 in IEEE Std 802.16e-2005"
55          ::= { wmanIf2eBsSsRspCapabilitiesEntry 6 }
56
57
58
59      wmanIf2eBsSsRspCapHoProcessTimer OBJECT-TYPE
60          SYNTAX      Unsigned32
61          UNITS       "frames"
62          MAX-ACCESS  read-only
63          STATUS      current
64          DESCRIPTION
65

```

"The duration in frames the MS shall wait until receipt of the next unsolicited network re-entry MAC management message as indicated in the HO Process Optimization element of the RNG-RSP message."

REFERENCE

"Subclause 11.7.13.2 in IEEE Std 802.16e-2005"
 ::= { wmanIf2eBsSsRspCapabilitiesEntry 7 }

wmanIf2eBsSsRspCapIdleModeTimeout OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Max time interval between MS Idle Mode Location Updates."

REFERENCE

"Subclause 11.7.20.1 in IEEE Std 802.16e-2005"
 DEFVAL { 4096 }
 ::= { wmanIf2eBsSsRspCapabilitiesEntry 8 }

wmanIf2eBsSsRspCapArqAckType OBJECT-TYPE

SYNTAX WmanIf2eArqAckType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of this parameter specifies the ARQ ACK type supported by the MS."

REFERENCE

"Subclause 11.7.23 in IEEE Std 802.16e-2005"
 ::= { wmanIf2eBsSsRspCapabilitiesEntry 9 }

wmanIf2eBsSsRspCapMacHeader OBJECT-TYPE

SYNTAX WmanIf2eMacHeaderSupp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicates whether or not the MS and BS support various types of MAC header and extended subheaders."

REFERENCE

"Subclause 11.7.25 in IEEE Std 802.16e-2005"
 ::= { wmanIf2eBsSsRspCapabilitiesEntry 10 }

wmanIf2eBsBasicCapabilitiesTable OBJECT-TYPE

SYNTAX SEQUENCE OF WmanIf2eBsBasicCapabilitiesEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table contains the basic capabilities of the BS as implemented in BS hardware and software. These capabilities along with the configuration for them (wmanIf2eBsCapabilitiesConfigTable) are used for negotiation of basic capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP messages. The negotiated capabilities are obtained by interSubclause of SS raw reported capabilities, BS raw

```

1         capabilities and BS configured capabilities. The objects in
2         the table have read-only access. The table is maintained
3         by BS."
4     ::= { wmanIf2BsCapabilities 7 }
5
6
7 wmanIf2eBsBasicCapabilitiesEntry OBJECT-TYPE
8     SYNTAX      WmanIf2eBsBasicCapabilitiesEntry
9     MAX-ACCESS  not-accessible
10    STATUS      current
11    DESCRIPTION
12        "This table provides one row for each BS sector and is
13        indexed by ifIndex."
14    INDEX { ifIndex }
15    ::= { wmanIf2eBsBasicCapabilitiesTable 1 }
16
17
18
19 WmanIf2eBsBasicCapabilitiesEntry ::= SEQUENCE {
20     wmanIf2eBsCapDownlinkCidSupport      WmanIf2eNumOfCid,
21     wmanIf2eBsCapPackingSupport          WmanIf2ePackingSupport,
22     wmanIf2eBsCapExtendedRtpsSupport     WmanIf2eExtRtpsSupport,
23     wmanIf2eBsCapMaxNumBurstToMs        INTEGER,
24     wmanIf2eBsCapIpAddrAllocMethod      WmanIf2eIpAllocMethod,
25     wmanIf2eBsCapHandoverSupported      WmanIf2eHandoverType,
26     wmanIf2eBsCapHoProcessTimer         Unsigned32,
27     wmanIf2eBsCapIdleModeTimeout        Unsigned32,
28     wmanIf2eBsCapArqAckType             WmanIf2eArqAckType,
29     wmanIf2eBsCapMacHeader              WmanIf2eMacHeaderSupp}
30
31
32
33 wmanIf2eBsCapDownlinkCidSupport OBJECT-TYPE
34     SYNTAX      WmanIf2eNumOfCid
35     MAX-ACCESS  read-only
36     STATUS      current
37     DESCRIPTION
38         "This object shows the number of Downlink CIDs the SS can
39         support."
40     ::= { wmanIf2eBsBasicCapabilitiesEntry 1 }
41
42
43
44 wmanIf2eBsCapPackingSupport OBJECT-TYPE
45     SYNTAX      WmanIf2ePackingSupport
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49         "Indicates the availability of MS support for Packing."
50     REFERENCE
51         "Subclause 11.7.8.11 in IEEE Std 802.16e-2005"
52     ::= { wmanIf2eBsBasicCapabilitiesEntry 2 }
53
54
55
56 wmanIf2eBsCapExtendedRtpsSupport OBJECT-TYPE
57     SYNTAX      WmanIf2eExtRtpsSupport
58     MAX-ACCESS  read-only
59     STATUS      current
60     DESCRIPTION
61         "Indicates the availability of MS support for extended
62         rtps."
63     REFERENCE
64
65

```

```

1         "Subclause 11.7.8.12 in IEEE Std 802.16e-2005"
2         ::= { wmanIf2eBsBasicCapabilitiesEntry 3 }
3
4
5 wmanIf2eBsCapMaxNumBurstToMs OBJECT-TYPE
6     SYNTAX      INTEGER (1..16)
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "Maximum number of bursts transmitted concurrently to the MS
11         , including all bursts without CID or with CIDs matching
12         the MS CIDs."
13
14     REFERENCE
15        "Subclause 11.7.8.13 in IEEE Std 802.16e-2005"
16        ::= { wmanIf2eBsBasicCapabilitiesEntry 4 }
17
18
19 wmanIf2eBsCapIpAddrAllocMethod OBJECT-TYPE
20     SYNTAX      WmanIf2eIpAllocMethod
21     MAX-ACCESS  read-only
22     STATUS      current
23     DESCRIPTION
24        "Indicates the method of allocating IP address for the
25         secondary management connection."
26
27     REFERENCE
28        "Subclause 11.7.11 in IEEE Std 802.16e-2005"
29        ::= { wmanIf2eBsBasicCapabilitiesEntry 5 }
30
31
32 wmanIf2eBsCapHandoverSupported OBJECT-TYPE
33     SYNTAX      WmanIf2eHandoverType
34     MAX-ACCESS  read-only
35     STATUS      current
36     DESCRIPTION
37        "Indicates what type(s) of Handover the BS or MS supports."
38
39     REFERENCE
40        "Subclause 11.7.12 in IEEE Std 802.16e-2005"
41        ::= { wmanIf2eBsBasicCapabilitiesEntry 6 }
42
43
44 wmanIf2eBsCapHoProcessTimer OBJECT-TYPE
45     SYNTAX      Unsigned32
46     UNITS       "frames"
47     MAX-ACCESS  read-only
48     STATUS      current
49     DESCRIPTION
50        "The duration in frames the MS shall wait until receipt of
51         the next unsolicited network re-entry MAC management
52         message as indicated in the HO Process Optimization
53         element of the RNG-RSP message."
54
55     REFERENCE
56        "Subclause 11.7.13.2 in IEEE Std 802.16e-2005"
57        ::= { wmanIf2eBsBasicCapabilitiesEntry 7 }
58
59
60
61 wmanIf2eBsCapIdleModeTimeout OBJECT-TYPE
62     SYNTAX      Unsigned32
63     UNITS       "seconds"
64     MAX-ACCESS  read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "Max time interval between MS Idle Mode Location Updates."
4      REFERENCE
5          "Subclause 11.7.20.1 in IEEE Std 802.16e-2005"
6      DEFVAL      { 4096 }
7      ::= { wmanIf2eBsBasicCapabilitiesEntry 8 }
8
9
10
11  wmanIf2eBsCapArqAckType OBJECT-TYPE
12      SYNTAX      WmanIf2eArqAckType
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "The value of this parameter specifies the ARQ ACK type
17           supported by the MS."
18      REFERENCE
19          "Subclause 11.7.23 in IEEE Std 802.16e-2005"
20      ::= { wmanIf2eBsBasicCapabilitiesEntry 9 }
21
22
23
24  wmanIf2eBsCapMacHeader OBJECT-TYPE
25      SYNTAX      WmanIf2eMacHeaderSupp
26      MAX-ACCESS  read-only
27      STATUS      current
28      DESCRIPTION
29          "Indicates whether or not the MS and BS support various
30           types of MAC header and extended subheaders."
31      REFERENCE
32          "Subclause 11.7.25 in IEEE Std 802.16e-2005"
33      ::= { wmanIf2eBsBasicCapabilitiesEntry 10 }
34
35
36
37  wmanIf2eBsCapabilitiesConfigTable OBJECT-TYPE
38      SYNTAX      SEQUENCE OF WmanIf2eBsCapabilitiesConfigEntry
39      MAX-ACCESS  not-accessible
40      STATUS      current
41      DESCRIPTION
42          "This table contains the configuration for basic
43           capabilities of BS. The table is intended to be used to
44           restrict the Capabilities implemented by BS, for example in
45           order to comply with local regulatory requirements. The BS
46           should use the configuration along with the implemented
47           Capabilities (wmanIf2eBsBasicCapabilitiesTable) for
48           negotiation of basic capabilities with SS using RNG-RSP,
49           SBC-RSP and REG-RSP messages. The negotiated capabilities
50           are obtained by interSubclause of SS reported capabilities,
51           BS raw capabilities and BS configured capabilities. The
52           objects in the table have read-write access. The rows are
53           created by BS as a copy of wmanIf2eBsBasicCapabilitiesTable
54           and can be modified by NMS."
55      ::= { wmanIf2BsCapabilities 8 }
56
57
58
59
60
61  wmanIf2eBsCapabilitiesConfigEntry OBJECT-TYPE
62      SYNTAX      WmanIf2eBsCapabilitiesConfigEntry
63      MAX-ACCESS  not-accessible
64      STATUS      current
65

```

```

1      DESCRIPTION
2          "This table provides one row for each BS sector and is
3          indexed by ifIndex."
4      INDEX { ifIndex }
5      ::= { wmanIf2eBsCapabilitiesConfigTable 1 }
6
7
8      WmanIf2eBsCapabilitiesConfigEntry ::= SEQUENCE {
9          wmanIf2eBsCapCfgDownlinkCidSupport      WmanIf2eNumOfCid,
10         wmanIf2eBsCapCfgPackingSupport           WmanIf2ePackingSupport,
11         wmanIf2eBsCapCfgExtendedRtpsSupport      WmanIf2eExtRtpsSupport,
12         wmanIf2eBsCapCfgMaxNumBurstToMs          INTEGER,
13         wmanIf2eBsCapCfgIpAddrAllocMethod        WmanIf2eIpAllocMethod,
14         wmanIf2eBsCapCfgHandoverSupported        WmanIf2eHandoverType,
15         wmanIf2eBsCapCfgHoProcessTimer           Unsigned32,
16         wmanIf2eBsCapCfgIdleModeTimeout          Unsigned32,
17         wmanIf2eBsCapCfgArqAckType               WmanIf2eArqAckType,
18         wmanIf2eBsCapCfgMacHeader                WmanIf2eMacHeaderSupp}
19
20
21
22
23     wmanIf2eBsCapCfgDownlinkCidSupport OBJECT-TYPE
24         SYNTAX      WmanIf2eNumOfCid
25         MAX-ACCESS   read-write
26         STATUS       current
27         DESCRIPTION
28             "This object shows the number of Downlink CIDs the SS can
29             support."
30         ::= { wmanIf2eBsCapabilitiesConfigEntry 1 }
31
32
33
34     wmanIf2eBsCapCfgPackingSupport OBJECT-TYPE
35         SYNTAX      WmanIf2ePackingSupport
36         MAX-ACCESS   read-only
37         STATUS       current
38         DESCRIPTION
39             "Indicates the availability of MS support for Packing."
40         REFERENCE
41             "Subclause 11.7.8.11 in IEEE Std 802.16e-2005"
42         ::= { wmanIf2eBsCapabilitiesConfigEntry 2 }
43
44
45
46     wmanIf2eBsCapCfgExtendedRtpsSupport OBJECT-TYPE
47         SYNTAX      WmanIf2eExtRtpsSupport
48         MAX-ACCESS   read-write
49         STATUS       current
50         DESCRIPTION
51             "Indicates the availability of MS support for extended
52             rtps."
53         REFERENCE
54             "Subclause 11.7.8.12 in IEEE Std 802.16e-2005"
55         ::= { wmanIf2eBsCapabilitiesConfigEntry 3 }
56
57
58
59     wmanIf2eBsCapCfgMaxNumBurstToMs OBJECT-TYPE
60         SYNTAX      INTEGER (1..16)
61         MAX-ACCESS   read-write
62         STATUS       current
63         DESCRIPTION
64             "Maximum number of bursts transmitted concurrently to the MS
65

```

```

1         , including all bursts without CID or with CIDs matching
2         the MS CIDs."
3     REFERENCE
4         "Subclause 11.7.8.13 in IEEE Std 802.16e-2005"
5         ::= { wmanIf2eBsCapabilitiesConfigEntry 4 }
6
7
8     wmanIf2eBsCapCfgIpAddrAllocMethod OBJECT-TYPE
9         SYNTAX      WmanIf2eIpAllocMethod
10        MAX-ACCESS   read-write
11        STATUS       current
12        DESCRIPTION
13            "Indicates the method of allocating IP address for the
14             secondary management connection."
15        REFERENCE
16            "Subclause 11.7.11 in IEEE Std 802.16e-2005"
17            ::= { wmanIf2eBsCapabilitiesConfigEntry 5 }
18
19
20
21    wmanIf2eBsCapCfgHandoverSupported OBJECT-TYPE
22        SYNTAX      WmanIf2eHandoverType
23        MAX-ACCESS   read-write
24        STATUS       current
25        DESCRIPTION
26            "Indicates what type(s) of Handover the BS or MS supports."
27        REFERENCE
28            "Subclause 11.7.12 in IEEE Std 802.16e-2005"
29            ::= { wmanIf2eBsCapabilitiesConfigEntry 6 }
30
31
32
33    wmanIf2eBsCapCfgHoProcessTimer OBJECT-TYPE
34        SYNTAX      Unsigned32
35        UNITS        "frames"
36        MAX-ACCESS   read-write
37        STATUS       current
38        DESCRIPTION
39            "The duration in frames the MS shall wait until receipt of
40             the next unsolicited network re-entry MAC management
41             message as indicated in the HO Process Optimization
42             element of the RNG-RSP message."
43        REFERENCE
44            "Subclause 11.7.13.2 in IEEE Std 802.16e-2005"
45            ::= { wmanIf2eBsCapabilitiesConfigEntry 7 }
46
47
48
49
50    wmanIf2eBsCapCfgIdleModeTimeout OBJECT-TYPE
51        SYNTAX      Unsigned32
52        UNITS        "seconds"
53        MAX-ACCESS   read-write
54        STATUS       current
55        DESCRIPTION
56            "Max time interval between MS Idle Mode Location Updates."
57        REFERENCE
58            "Subclause 11.7.20.1 in IEEE Std 802.16e-2005"
59            DEFVAL    { 4096 }
60            ::= { wmanIf2eBsCapabilitiesConfigEntry 8 }
61
62
63
64    wmanIf2eBsCapCfgArqAckType OBJECT-TYPE
65

```



```

1      SYNTAX      WmanIf2eArqAckType
2      MAX-ACCESS  read-write
3      STATUS      current
4      DESCRIPTION
5          "The value of this parameter specifies the ARQ ACK type
6          supported by the MS."
7      REFERENCE
8          "Subclause 11.7.23 in IEEE Std 802.16e-2005"
9      ::= { wmanIf2eBsCapabilitiesConfigEntry 9 }
10
11
12
13  wmanIf2eBsCapCfgMacHeader OBJECT-TYPE
14      SYNTAX      WmanIf2eMacHeaderSupp
15      MAX-ACCESS  read-write
16      STATUS      current
17      DESCRIPTION
18          "Indicates whether or not the MS and BS support various
19          types of MAC header and extended subheaders."
20      REFERENCE
21          "Subclause 11.7.25 in IEEE Std 802.16e-2005"
22      ::= { wmanIf2eBsCapabilitiesConfigEntry 10 }
23
24
25
26  wmanIf2BsSsActionsTable OBJECT-TYPE
27      SYNTAX      SEQUENCE OF WmanIf2BsSsActionsEntry
28      MAX-ACCESS  not-accessible
29      STATUS      current
30      DESCRIPTION
31          "This table contains all the actions specified for SSs in
32          the standard. The actions are routed down to SS using
33          unsolicited MAC messages: REG-RSP, DREG-REQ and RES-CMD.
34          The table also contains the parameters of the actions in
35          cases where they are specified by the standard."
36      ::= { wmanIf2BsCps 5 }
37
38
39
40
41  wmanIf2BsSsActionsEntry OBJECT-TYPE
42      SYNTAX      WmanIf2BsSsActionsEntry
43      MAX-ACCESS  not-accessible
44      STATUS      current
45      DESCRIPTION
46          "This table is indexed by wmanIf2BsSsActionsMacAddress. The
47          action can be requested for SS in any state not only those
48          registered. However BS will decide whether the action is
49          applicable to the SS based on its current state and execute
50          it or skip it as defined in each action definition."
51      INDEX { wmanIf2BsSsActionsMacAddress }
52      ::= { wmanIf2BsSsActionsTable 1 }
53
54
55
56  WmanIf2BsSsActionsEntry ::= SEQUENCE {
57      wmanIf2BsSsActionsMacAddress      MacAddress,
58      wmanIf2BsSsActionsResetSs         INTEGER,
59      wmanIf2BsSsActionsAbortSs         INTEGER,
60      wmanIf2BsSsActionsOverrideDnFreq  Unsigned32,
61      wmanIf2BsSsActionsOverrideChannelId  INTEGER,
62      wmanIf2BsSsActionsDeReRegSs       INTEGER,
63      wmanIf2BsSsActionsDeReRegSsCode   INTEGER,
64
65

```

```

1          wmanIf2BsSsActionsRowStatus          RowStatus}
2
3
4 wmanIf2BsSsActionsMacAddress OBJECT-TYPE
5     SYNTAX      MacAddress
6     MAX-ACCESS  not-accessible
7     STATUS      current
8     DESCRIPTION
9         "This object uniquely identifies the SS as an action
10        target."
11
12     ::= { wmanIf2BsSsActionsEntry 1 }
13
14 wmanIf2BsSsActionsResetSs OBJECT-TYPE
15     SYNTAX      INTEGER {actionsResetSsNoAction(0),
16                      actionsResetSs(1)}
17
18     MAX-ACCESS  read-create
19     STATUS      current
20     DESCRIPTION
21         "This object should be implemented as follows:
22         - When set to actionsResetSs value, instructs BS to send
23           RES-CMD to SS
24         - When set to value different than actionsResetSs it
25           should be ignored
26         - When read it should return actionsResetSsNoAction
27         The RES-CMD message shall be transmitted by the BS on an
28         SS Basic CID to force the SS to reset itself,
29         reinitialize its MAC, and repeat initial system access."
30
31     REFERENCE
32         "Subclause 6.3.2.3.22 in IEEE Std 802.16-2004"
33
34     ::= { wmanIf2BsSsActionsEntry 2 }
35
36
37 wmanIf2BsSsActionsAbortSs OBJECT-TYPE
38     SYNTAX      INTEGER {actionsAbortSsNoAction(0),
39                      actionsAbortSs(1),
40                      actionAbortSsParams(2)}
41
42
43     MAX-ACCESS  read-create
44     STATUS      current
45     DESCRIPTION
46         "This object should be implemented as follows:
47         - When set to actionsAbortSs value, it instructs BS to send
48           unsolicited RNG-RSP with Ranging Status equal to 'abort'
49           without override parameters
50         - When set to actionAbortSsParams value, it instructs BS to
51           send unsolicited RNG-RSP with Ranging Status equal to
52           'abort' and with 'Downlink Frequency Override' and
53           'Uplink Channel ID Override' parameters.
54         - When set to any other value it should be ignored
55         - When read it should returned actionsAbortSsNoAction"
56
57     REFERENCE
58         "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
59
60     ::= { wmanIf2BsSsActionsEntry 3 }
61
62
63 wmanIf2BsSsActionsOverrideDnFreq OBJECT-TYPE
64     SYNTAX      Unsigned32
65

```

```

1      UNITS          "kHz"
2      MAX-ACCESS    read-create
3      STATUS        current
4      DESCRIPTION
5
6          "This object is used as a parameter of the AbortSs action
7          with the code actionAbortSsParams. It is used for licensed
8          bands only. It defines the Center frequency, in kHz, of
9          new downlink channel where the SS should redo initial
10         ranging."
11
12     REFERENCE
13         "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
14     ::= { wmanIf2BsSsActionsEntry 4 }
15
16 wmanIf2BsSsActionsOverrideChannelId OBJECT-TYPE
17     SYNTAX      INTEGER (0..199)
18     MAX-ACCESS  read-create
19     STATUS      current
20     DESCRIPTION
21
22         "This object is used as a parameter of the AbortSs action
23         with the code actionAbortSsParams. It is coded as follows:
24         - Licensed bands: The identifier of the uplink channel
25         with which the SS is to redo initial ranging (not used
26         with PHYs without channelized uplinks).
27         - License-exempt bands: The Channel Nr (see 8.5.1) where
28         the SS should redo initial ranging."
29
30     REFERENCE
31         "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
32     ::= { wmanIf2BsSsActionsEntry 5 }
33
34 wmanIf2BsSsActionsDeReRegSs OBJECT-TYPE
35     SYNTAX      INTEGER {actionsDeReRegSsNoAction(0),
36                     actionsDeReRegSs(1)}
37     MAX-ACCESS  read-create
38     STATUS      current
39     DESCRIPTION
40
41         "This object should be implemented as follows:
42         - When set to actionsDeReRegSs value, instructs BS to
43         send DREG-CMD to SS with specified action code
44         - When set to value different than actionsDeReRegSs it
45         should be ignored
46         - When read it should return actionsDeReRegSsNoAction
47         The DREG-CMD message shall be transmitted by the BS on an
48         SS Basic CID to force the SS to change its access state.
49         Upon receiving a DREG-CMD, the SS shall take the action
50         indicated by the action code defined by
51         wmanIf2BsSsActionsDeReRegSsCode."
52
53     REFERENCE
54         "Subclause 6.3.2.3.26 in IEEE Std 802.16-2004"
55     ::= { wmanIf2BsSsActionsEntry 6 }
56
57 wmanIf2BsSsActionsDeReRegSsCode OBJECT-TYPE
58     SYNTAX      INTEGER {actionsDeReRegSsCodeChangeChan(0),
59                     actionsDeReRegSsCodeNoTransmit(1),
60                     actionsDeReRegSsCodeLtdTransmit(2),

```

```

1          actionsDeReRegSsCodeResume(3) }
2
3      MAX-ACCESS    read-create
4      STATUS        current
5      DESCRIPTION
6          "This object defines the action code for
7          wmanIf2BsSsActionsDeReRegSs action. The codes are defined
8          as follows:
9          actionsDeReRegSsCodeChangeChan - SS shall leave the
10         current channel and attempt to access another channel.
11         actionsDeReRegSsCodeNoTransmit - SS shall listen to the
12         current channel but shall not transmit until an
13         RES-CMD message or DREG_CMD with an Action Code that
14         allows transmission is received.
15         actionsDeReRegSsCodeLtdTransmit - SS shall listen to the
16         current channel but only transmit on the Basic,
17         Primary Management and 2nd Management Connections.
18         actionsDeReRegSsCodeResume - SS shall return to normal
19         operation and may transmit on any of its active
20         connections."
21
22      REFERENCE
23          "Subclause 6.3.2.3.26, Table 55 in IEEE Std 802.16-2004"
24      ::= { wmanIf2BsSsActionsEntry 7 }
25
26
27
28
29      wmanIf2BsSsActionsRowStatus OBJECT-TYPE
30          SYNTAX      RowStatus
31          MAX-ACCESS    read-create
32          STATUS        current
33          DESCRIPTION
34              "This object is used to ensure that the write operation to
35              multiple columns is guaranteed to be treated as atomic
36              operation by agent."
37          ::= { wmanIf2BsSsActionsEntry 8 }
38
39
40
41      --
42      -- wmanIf2BsSsPowerSavingStatusTable contains the power saving status
43      --
44      wmanIf2BsSsPowerSavingStatusTable OBJECT-TYPE
45          SYNTAX      SEQUENCE OF WmanIf2BsSsPowerSavingStatusEntry
46          MAX-ACCESS    not-accessible
47          STATUS        current
48          DESCRIPTION
49              "This table contains the power saving status for each CID
50              in an MS. When the BS roams to a different BS, all entries
51              associated with such MS will be deleted."
52          ::= { wmanIf2BsCps 6 }
53
54
55
56      wmanIf2BsSsPowerSavingStatusEntry OBJECT-TYPE
57          SYNTAX      WmanIf2BsSsPowerSavingStatusEntry
58          MAX-ACCESS    not-accessible
59          STATUS        current
60          DESCRIPTION
61              "This table provides one row for each CID in an MS, and
62              is indexed by ifIndex, wmanIf2BsSsMacAddress, and
63              wmanIf2BsSsCid."
64
65

```

```

1      INDEX      { ifIndex,
2                  wmanIf2BsSsMacAddress,
3                  wmanIf2BsSsCid }
4      ::= { wmanIf2BsSsPowerSavingStatusTable 1 }
5
6
7      WmanIf2BsSsPowerSavingStatusEntry ::= SEQUENCE {
8          wmanIf2BsSsCid                      WmanIf2CidType,
9          wmanIf2BsSsPowerSavingClassId       WmanIf2PsClassId}
10
11
12      wmanIf2BsSsCid OBJECT-TYPE
13          SYNTAX      WmanIf2CidType
14          MAX-ACCESS  read-only
15          STATUS      current
16          DESCRIPTION
17              "A 16 bit channel identifier to identify a connection."
18          ::= { wmanIf2BsSsPowerSavingStatusEntry 1 }
19
20
21      wmanIf2BsSsPowerSavingClassId OBJECT-TYPE
22          SYNTAX      WmanIf2PsClassId
23          MAX-ACCESS  read-only
24          STATUS      current
25          DESCRIPTION
26              "wmanIf2BsSsPowerSavingClassId identifies the power
27              saving class associated with this CID. It maps to an
28              entry in wmanIf2BsSsPowerSavingClassesTable."
29          ::= { wmanIf2BsSsPowerSavingStatusEntry 2 }
30
31
32      --
33      -- wmanIf2BsSsPowerSavingClassesTable contains the power saving classes
34      -- information
35      --
36
37      wmanIf2BsSsPowerSavingClassesTable OBJECT-TYPE
38          SYNTAX      SEQUENCE OF WmanIf2BsSsPowerSavingClassesEntry
39          MAX-ACCESS  not-accessible
40          STATUS      current
41          DESCRIPTION
42              "This table contains the power saving classes definitions,
43              and activation / deactivation information that are provided
44              by MOB_SLP-REQ and MOB_SLP-RSP messages. When the BS roams
45              to a different BS, all entries associated with such MS will
46              be deleted."
47          ::= { wmanIf2BsCps 7 }
48
49
50
51      wmanIf2BsSsPowerSavingClassesEntry OBJECT-TYPE
52          SYNTAX      WmanIf2BsSsPowerSavingClassesEntry
53          MAX-ACCESS  not-accessible
54          STATUS      current
55          DESCRIPTION
56              "This table is indexed by ifIndex, wmanIf2BsSsMacAddress,
57              and wmanIf2BsSsPsClassesId. It is intended to support both
58              unicast and multicast service flows.
59              wmanIf2BsSsMacAddress contains the MAC address of the MS
60              to which the power saving classes are associated."
61          INDEX { ifIndex,

```

```

1         wmanIf2BsSsMacAddress,
2         wmanIf2BsSsPsClassId }
3     ::= { wmanIf2BsSsPowerSavingClassesTable 1 }
4
5
6 WmanIf2BsSsPowerSavingClassesEntry ::= SEQUENCE {
7     wmanIf2BsSsPsClassId          WmanIf2PsClassId,
8     wmanIf2BsSsStartFrameNumber   INTEGER,
9     wmanIf2BsSsPowerSavingClassType WmanIf2ePsClassType,
10    wmanIf2BsSsPsClassCidDirection WmanIf2ePsClassCidDir,
11    wmanIf2BsSsTrafficTriggeredWakening INTEGER,
12    wmanIf2BsSsInitialSleepWindow    INTEGER,
13    wmanIf2BsSsFinalSleepWindowBase  INTEGER,
14    wmanIf2BsSsFinalSleepWindowExponent INTEGER,
15    wmanIf2BsSsListeningWindow       INTEGER,
16    wmanIf2BsSsPowerSavingMode       WmanIf2PowerSavingMode,
17    wmanIf2BsSsSlpId                 INTEGER}
18
19
20
21 wmanIf2BsSsPsClassId OBJECT-TYPE
22     SYNTAX          WmanIf2PsClassId
23     MAX-ACCESS      not-accessible
24     STATUS          current
25     DESCRIPTION
26         "This object uniquely identifies the power saving classes
27         in a MS."
28     ::= { wmanIf2BsSsPowerSavingClassesEntry 1 }
29
30
31
32 wmanIf2BsSsStartFrameNumber OBJECT-TYPE
33     SYNTAX          INTEGER
34     MAX-ACCESS      read-write
35     STATUS          current
36     DESCRIPTION
37         "Start frame number for first sleep window."
38     REFERENCE
39         "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
40     ::= { wmanIf2BsSsPowerSavingClassesEntry 2 }
41
42
43
44 wmanIf2BsSsPowerSavingClassType OBJECT-TYPE
45     SYNTAX          WmanIf2ePsClassType
46     MAX-ACCESS      read-write
47     STATUS          current
48     DESCRIPTION
49         "Power saving classes type I - BE & NRT-VR,
50         Power saving classes type II - UGS & RT-VR,
51         Power saving classes type III - multicast, management CID"
52     REFERENCE
53         "Subclause 6.3.21.2-4, in IEEE Std 802.16e-2005"
54     ::= { wmanIf2BsSsPowerSavingClassesEntry 3 }
55
56
57
58 wmanIf2BsSsPsClassCidDirection OBJECT-TYPE
59     SYNTAX          WmanIf2ePsClassCidDir
60     MAX-ACCESS      read-write
61     STATUS          current
62     DESCRIPTION
63         "The direction of power saving class's CIDs."
64
65

```

REFERENCE

"Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 4 }

wmanIf2BsSsTrafficTriggeredWakening OBJECT-TYPE

SYNTAX INTEGER (0..1)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"0 = Power Saving Class shall not be deactivated if
 traffic appears at the connection as per 6.3.19.2.
 1 = Power Saving Class shall be deactivated if
 traffic appears at the connection as 6.3.19.2."

REFERENCE

"Subclause 6.3.19.2, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 5 }

wmanIf2BsSsInitialSleepWindow OBJECT-TYPE

SYNTAX INTEGER (0..255)

UNITS "frame"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The initial duration for the sleep window. It is not
 relevant for Power Saving Class type III, and shall
 return '0'."

REFERENCE

"Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 6 }

wmanIf2BsSsFinalSleepWindowBase OBJECT-TYPE

SYNTAX INTEGER (0..1023)

UNITS "frame"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The final value for the sleep interval. It is not
 relevant for Power Saving Class type II, and shall
 return '0'. For Power Saving Class type III, it is the
 base for duration of single sleep window request."

REFERENCE

"Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 7 }

wmanIf2BsSsFinalSleepWindowExponent OBJECT-TYPE

SYNTAX INTEGER (0..7)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The factor by which the final-sleep window base is
 multiplied in order to calculate the final-sleep window.
 The following formula is used:
 final-sleep window = final-sleep window base x
 $2^{(\text{final-sleep window exponent})}$ "

For Power Saving Class type III, it is the exponent for the duration of single sleep window request."

REFERENCE

"Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 8 }

wmanIf2BsSsListeningWindow OBJECT-TYPE

SYNTAX INTEGER (0..255)

UNITS "frame"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The Duration of MS listening window. It is not relevant for Power Saving Class type III, and shall return '0'."

REFERENCE

"Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 9 }

wmanIf2BsSsPowerSavingMode OBJECT-TYPE

SYNTAX WmanIf2PowerSavingMode

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Indicate whether the power saving class mode of such CID is active or not.
 wmanIf2BsSsPowerSavingMode = Sleep_Approved && Operation."

REFERENCE

"Subclause 6.3.2.3.45, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 10 }

wmanIf2BsSsSlpId OBJECT-TYPE

SYNTAX INTEGER (0..1023)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"wmanIf2BsSsSlpId is assigned by the BS whenever an MS is instructed to enter sleep mode. This number shall be unique among all MSs that are in sleep mode."

REFERENCE

"Subclause 6.3.2.3.45, in IEEE Std 802.16e-2005"
 ::= { wmanIf2BsSsPowerSavingClassesEntry 11 }

--
 -- Mobile Station Sleep Mode Statistics Table

wmanIf2BsSsSleepModeStatisticsTable OBJECT-TYPE

SYNTAX SEQUENCE OF WmanIf2BsSsSleepModeStatisticsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table contains the sleep mode statistic for MS. This table shall be maintained as FIFO to store the sleep mode statistics over a period of time that is subject to

implementation. This statistics information can be to monitor, fine tuning, or debugging the power saving performance of each MS. When the statistics entry for an MS reaches the limit, it wraps around to the beginning, and overwrites the oldest entry with the new entry. When the BS roams to a different BS, all entries associated with such MS will be deleted."

REFERENCE

"6.3.21 in IEEE Std 802.16e-2005"

::= { wmanIf2BsCps 8 }

wmanIf2BsSsSleepModeStatisticsEntry OBJECT-TYPE

SYNTAX WmanIf2BsSsSleepModeStatisticsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Each entry in the table contains the event of an MS entering the sleep mode. It is indexed by ifIndex, wmanIf2BsSsMacAddress, and wmanIf2BsSsStatisticsIndex. wmanIf2BsSsStatisticsIndex is the index to sleep mode event entry in the table, and should be increased monotonically, and wraps around when it reaches the implementation specific limit. A time stamp is provided in each entry to indicate when the sleep mode event took place."

INDEX { ifIndex,
wmanIf2BsSsMacAddress,
wmanIf2BsSsCid,
wmanIf2BsSsStatisticsIndex }
::= { wmanIf2BsSsSleepModeStatisticsTable 1 }

WmanIf2BsSsSleepModeStatisticsEntry ::= SEQUENCE {

wmanIf2BsSsStatisticsIndex Unsigned32,
wmanIf2BsSsSleepWindowStarted Unsigned32,
wmanIf2BsSsListeningWindowStarted Unsigned32,
wmanIf2BsSsPendingMsdu INTEGER,
wmanIf2BsSsSleepWindowTimeStamp DateAndTime }

wmanIf2BsSsStatisticsIndex OBJECT-TYPE

SYNTAX Unsigned32 (1 .. 4294967295)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"wmanIf2BsSsStatisticsIndex identifies the entry in the table where the latest sleep mode event took place."

::= { wmanIf2BsSsSleepModeStatisticsEntry 1 }

wmanIf2BsSsSleepWindowStarted OBJECT-TYPE

SYNTAX Unsigned32 (1 .. 166777215)

UNITS "frame"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"wmanIf2BsSsSleepWindowStarted identifies when the sleep mode is activated."

```

1          wmanIf2BsSsSleepWindowStarted = current frame number +
2          Start_frame_number.
3          The frame number is provided in the DL-MAP, and is
4          incremented by 1 MOD 2^24 each frame."
5          ::= { wmanIf2BsSsSleepModeStatisticsEntry 2 }
6
7
8  wmanIf2BsSsListeningWindowStarted OBJECT-TYPE
9      SYNTAX      Unsigned32 (1 .. 166777215)
10     UNITS       "frame"
11     MAX-ACCESS  read-only
12     STATUS      current
13     DESCRIPTION
14         "wmanIf2BsSsListeningWindowStarted identifies when the sleep
15         mode is deactivated.
16         wmanIf2BsSsListeningWindowStarted =
17         wmanIf2BsSsListeningWindowStarted + sleep window
18         The frame number is provided in the DL-MAP, and is
19         incremented by 1 MOD 2^24 each frame."
20         ::= { wmanIf2BsSsSleepModeStatisticsEntry 3 }
21
22
23  wmanIf2BsSsPendingMsdu OBJECT-TYPE
24      SYNTAX      INTEGER
25      MAX-ACCESS  read-only
26      STATUS      current
27      DESCRIPTION
28          "Indicate the number of MAC SDU that are received from the
29          network during the sleep window."
30          ::= { wmanIf2BsSsSleepModeStatisticsEntry 4 }
31
32
33  wmanIf2BsSsSleepWindowTimeStamp OBJECT-TYPE
34      SYNTAX      DateAndTime
35      MAX-ACCESS  read-only
36      STATUS      current
37      DESCRIPTION
38          "This is the time when sleep window is started in seconds.
39          The definition of time is as in IETF RFC 868."
40          ::= { wmanIf2BsSsSleepModeStatisticsEntry 5 }
41
42
43  --
44  -- Base station PKM group
45  -- wmanIf2BsPkmObjects contain the Base Station Privacy Sublayer objects
46  --
47  wmanIf2BsPkmObjects OBJECT IDENTIFIER ::= { wmanIf2BsObjects 3 }
48
49
50  --
51  -- Table wmanIf2BsPkmBaseTable
52  --
53  wmanIf2BsPkmBaseTable OBJECT-TYPE
54      SYNTAX      SEQUENCE OF WmanIf2BsPkmBaseEntry
55      MAX-ACCESS  not-accessible
56      STATUS      current
57      DESCRIPTION
58          "This table describes the basic PKM attributes of each Base
59          Station wireless interface."
60
61
62
63
64
65

```

```

1      ::= { wmanIf2BsPkmObjects 1 }
2
3
4  wmanIf2BsPkmBaseEntry OBJECT-TYPE
5      SYNTAX      WmanIf2BsPkmBaseEntry
6      MAX-ACCESS  not-accessible
7      STATUS      current
8      DESCRIPTION
9          "Each entry contains objects describing attributes of one
10         BS wireless interface."
11      INDEX      { ifIndex }
12      ::= { wmanIf2BsPkmBaseTable 1 }
13
14
15  WmanIf2BsPkmBaseEntry ::= SEQUENCE {
16      wmanIf2BsPkmDefaultAuthLifetime      Integer32,
17      wmanIf2BsPkmDefaultTekLifetime        Integer32,
18      wmanIf2BsPkmDefaultSelfSigManufCertTrust INTEGER,
19      wmanIf2BsPkmCheckCertValidityPeriods  TruthValue,
20      wmanIf2BsPkmAuthentInfos              Counter32,
21      wmanIf2BsPkmAuthRequests              Counter32,
22      wmanIf2BsPkmAuthReplies               Counter32,
23      wmanIf2BsPkmAuthRejects               Counter32,
24      wmanIf2BsPkmAuthInvalids              Counter32,
25      wmanIf2BsPkmAuthGraceTime             Integer32,
26      wmanIf2BsPkmTekGraceTime              Integer32,
27      wmanIf2BsPkmAuthWaitTimeout           Integer32,
28      wmanIf2BsPkmReauthWaitTimeout         Integer32,
29      wmanIf2BsPkmOpWaitTimeout             Integer32,
30      wmanIf2BsPkmRekeyWaitTimeout          Integer32,
31      wmanIf2BsPkmAuthRejectWaitTimeout     Integer32}
32
33
34  wmanIf2BsPkmDefaultAuthLifetime OBJECT-TYPE
35      SYNTAX      Integer32 (86400..604800)
36      UNITS       "seconds"
37      MAX-ACCESS  read-write
38      STATUS      current
39      DESCRIPTION
40          "The value of this object is the default lifetime, in
41          seconds, the BS assigns to a new authorization key."
42      REFERENCE
43          "Table 343 in IEEE Std 802.16-2004"
44      DEFVAL      { 604800 }
45      ::= { wmanIf2BsPkmBaseEntry 1 }
46
47
48  wmanIf2BsPkmDefaultTekLifetime OBJECT-TYPE
49      SYNTAX      Integer32 (1800..604800)
50      UNITS       "seconds"
51      MAX-ACCESS  read-write
52      STATUS      current
53      DESCRIPTION
54          "The value of this object is the default lifetime, in
55          seconds, the BS assigns to a new Traffic Encryption
56          Key (TEK). "
57      REFERENCE
58          "Table 343 in IEEE Std 802.16-2004"

```

```

1      DEFVAL          { 43200 }
2      ::= { wmanIf2BsPkmBaseEntry 2 }
3
4
5
6      wmanIf2BsPkmDefaultSelfSigManufCertTrust OBJECT-TYPE
7          SYNTAX      INTEGER {trusted (1),
8                      untrusted (2)}
9
10         MAX-ACCESS   read-write
11         STATUS       current
12         DESCRIPTION
13             "This object determines the default trust of all (new)
14             self-signed manufacturer certificates obtained after
15             setting the object."
16         ::= { wmanIf2BsPkmBaseEntry 3 }
17
18
19      wmanIf2BsPkmCheckCertValidityPeriods OBJECT-TYPE
20          SYNTAX      TruthValue
21          MAX-ACCESS   read-write
22          STATUS       current
23          DESCRIPTION
24              "Setting this object to TRUE causes all certificates
25              received thereafter to have their validity periods (and
26              their chain's validity periods) checked against the current
27              time of day. A FALSE setting will cause all certificates
28              received Thereafter to not have their validity periods
29              (nor their chain's validity periods) checked against the
30              current time of day."
31          ::= { wmanIf2BsPkmBaseEntry 4 }
32
33
34
35
36      wmanIf2BsPkmAuthentInfos OBJECT-TYPE
37          SYNTAX      Counter32
38          MAX-ACCESS   read-only
39          STATUS       current
40          DESCRIPTION
41              "The value of this object is the count of times the BS has
42              received an Authentication Information message from any
43              SS."
44          ::= { wmanIf2BsPkmBaseEntry 5 }
45
46
47
48      wmanIf2BsPkmAuthRequests OBJECT-TYPE
49          SYNTAX      Counter32
50          MAX-ACCESS   read-only
51          STATUS       current
52          DESCRIPTION
53              "The value of this object is the count of times the BS has
54              received an Authorization Request message from any SS"
55          ::= { wmanIf2BsPkmBaseEntry 6 }
56
57
58
59      wmanIf2BsPkmAuthReplies OBJECT-TYPE
60          SYNTAX      Counter32
61          MAX-ACCESS   read-only
62          STATUS       current
63          DESCRIPTION
64              "The value of this object is the count of times the BS has
65

```

```

1      transmitted an Authorization Reply message to any SS."
2      ::= { wmanIf2BSPkmBaseEntry 7 }
3
4
5      wmanIf2BSPkmAuthRejects OBJECT-TYPE
6          SYNTAX      Counter32
7          MAX-ACCESS   read-only
8          STATUS      current
9          DESCRIPTION
10             "The value of this object is the count of times the BS has
11             transmitted an Authorization Reject message to any SS."
12             ::= { wmanIf2BSPkmBaseEntry 8 }
13
14
15      wmanIf2BSPkmAuthInvalids OBJECT-TYPE
16          SYNTAX      Counter32
17          MAX-ACCESS   read-only
18          STATUS      current
19          DESCRIPTION
20             "The value of this object is the count of times the BS has
21             transmitted an Authorization Invalid message to any SS."
22             ::= { wmanIf2BSPkmBaseEntry 9 }
23
24
25
26      wmanIf2BSPkmAuthGraceTime OBJECT-TYPE
27          SYNTAX      Integer32 (300..3024000)
28          UNITS        "seconds"
29          MAX-ACCESS   read-write
30          STATUS      current
31          DESCRIPTION
32             "The value of this object is the grace time for an
33             authorization key. A SS is expected to start trying to get
34             a new authorization key beginning AuthGraceTime seconds
35             before the authorization key actually expires."
36             REFERENCE
37                 "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
38             DEFVAL    { 600 }
39             ::= { wmanIf2BSPkmBaseEntry 10 }
40
41
42
43
44      wmanIf2BSPkmTekGraceTime OBJECT-TYPE
45          SYNTAX      Integer32 (300..3024000)
46          UNITS        "seconds"
47          MAX-ACCESS   read-write
48          STATUS      current
49          DESCRIPTION
50             "The value of this object is the grace time for the TEK in
51             seconds. The SS is expected to start trying to acquire a
52             new TEK beginning TEK GraceTime seconds before the
53             expiration of the most recent TEK."
54             REFERENCE
55                 "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
56             DEFVAL    { 3600 }
57             ::= { wmanIf2BSPkmBaseEntry 11 }
58
59
60
61
62      wmanIf2BSPkmAuthWaitTimeout OBJECT-TYPE
63          SYNTAX      Integer32 (2..30)
64          UNITS        "seconds"
65

```

```

1      MAX-ACCESS    read-write
2      STATUS        current
3      DESCRIPTION
4          "The value of this object is the Authorize Wait Timeout."
5      REFERENCE
6          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
7      DEFVAL        { 10 }
8      ::= { wmanIf2BSPkmBaseEntry 12 }
9
10
11
12  wmanIf2BSPkmReauthWaitTimeout OBJECT-TYPE
13      SYNTAX        Integer32 (2..30)
14      UNITS          "seconds"
15      MAX-ACCESS    read-write
16      STATUS        current
17      DESCRIPTION
18          "The value of this object is the Reauthorize Wait Timeout
19           in seconds."
20      REFERENCE
21          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
22      DEFVAL        { 10 }
23      ::= { wmanIf2BSPkmBaseEntry 13 }
24
25
26
27  wmanIf2BSPkmOpWaitTimeout OBJECT-TYPE
28      SYNTAX        Integer32 (1..10)
29      UNITS          "seconds"
30      MAX-ACCESS    read-write
31      STATUS        current
32      DESCRIPTION
33          "The value of this object is the Operational Wait Timeout
34           in seconds."
35      REFERENCE
36          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
37      DEFVAL        { 1 }
38      ::= { wmanIf2BSPkmBaseEntry 14 }
39
40
41
42
43  wmanIf2BSPkmRekeyWaitTimeout OBJECT-TYPE
44      SYNTAX        Integer32 (1..10)
45      UNITS          "seconds"
46      MAX-ACCESS    read-write
47      STATUS        current
48      DESCRIPTION
49          "The value of this object is the Rekey Wait Timeout in
50           seconds."
51      REFERENCE
52          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
53      DEFVAL        { 1 }
54      ::= { wmanIf2BSPkmBaseEntry 15 }
55
56
57
58
59  wmanIf2BSPkmAuthRejectWaitTimeout OBJECT-TYPE
60      SYNTAX        Integer32 (10..600)
61      UNITS          "seconds"
62      MAX-ACCESS    read-write
63      STATUS        current
64      DESCRIPTION
65

```

```

1         "The value of this object is the Authorization Reject Wait
2         Timeout in seconds."
3
4     REFERENCE
5         "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
6     DEFVAL      { 60 }
7     ::= { wmanIf2BsPkmBaseEntry 16 }
8
9
10    --
11    -- Table wmanIf2BsSsPkmAuthTable
12    --
13    wmanIf2BsSsPkmAuthTable OBJECT-TYPE
14        SYNTAX      SEQUENCE OF WmanIf2BsSsPkmAuthEntry
15        MAX-ACCESS  not-accessible
16        STATUS      current
17        DESCRIPTION
18            "This table describes PKM attributes related
19            to the authorization for each SS. The BS maintains one
20            Primary Security Association with each Baseline
21            Privacy-enabled SS on each BS wireless interface."
22            ::= { wmanIf2BsPkmObjects 2 }
23
24
25
26    wmanIf2BsSsPkmAuthEntry OBJECT-TYPE
27        SYNTAX      WmanIf2BsSsPkmAuthEntry
28        MAX-ACCESS  not-accessible
29        STATUS      current
30        DESCRIPTION
31            "The BS MUST create one entry per SS per wireless
32            interface, based on the receipt of an Authorization
33            Request message and MUST not delete the entry before
34            the SS authorization permanently expires."
35
36    INDEX          { ifIndex, wmanIf2BsSsPkmAuthMacAddress }
37    ::= { wmanIf2BsSsPkmAuthTable 1 }
38
39
40
41    WmanIf2BsSsPkmAuthEntry ::= SEQUENCE {
42        wmanIf2BsSsPkmAuthMacAddress      MacAddress,
43        wmanIf2BsSsPkmAuthKeySequenceNumber Integer32,
44        wmanIf2BsSsPkmAuthExpiresOld      DateAndTime,
45        wmanIf2BsSsPkmAuthExpiresNew      DateAndTime,
46        wmanIf2BsSsPkmAuthLifetime        Integer32,
47        wmanIf2BsSsPkmAuthReset            INTEGER,
48        wmanIf2BsSsPkmAuthInfos            Counter64,
49        wmanIf2BsSsPkmAuthRequests         Counter64,
50        wmanIf2BsSsPkmAuthReplies          Counter64,
51        wmanIf2BsSsPkmAuthRejects          Counter64,
52        wmanIf2BsSsPkmAuthInvalids         Counter64,
53        wmanIf2BsSsPkmAuthRejectErrorCode  INTEGER,
54        wmanIf2BsSsPkmAuthRejectErrorString SnmpAdminString,
55        wmanIf2BsSsPkmAuthInvalidErrorCode INTEGER,
56        wmanIf2BsSsPkmAuthInvalidErrorString SnmpAdminString,
57        wmanIf2BsSsPkmAuthPrimarySAId      INTEGER,
58        wmanIf2BsSsPkmAuthValidStatus      INTEGER}
59
60
61
62
63    wmanIf2BsSsPkmAuthMacAddress OBJECT-TYPE
64        SYNTAX      MacAddress
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "The value of this object is the physical address of the SS
5           to which the authorization association applies."
6      ::= { wmanIf2BsSsPkmAuthEntry 1 }
7
8
9
10     wmanIf2BsSsPkmAuthKeySequenceNumber OBJECT-TYPE
11         SYNTAX      Integer32 (0..15)
12         MAX-ACCESS  read-only
13         STATUS      current
14         DESCRIPTION
15             "The value of this object is the most recent authorization
16              key sequence number for this SS."
17         ::= { wmanIf2BsSsPkmAuthEntry 2 }
18
19
20     wmanIf2BsSsPkmAuthExpiresOld OBJECT-TYPE
21         SYNTAX      DateAndTime
22         MAX-ACCESS  read-only
23         STATUS      current
24         DESCRIPTION
25             "The value of this object is the actual clock time for
26              expiration of the immediate predecessor of the most recent
27              authorization key for this FSM. If this FSM has only one
28              authorization key, then the value is the time of activation
29              of this FSM."
30         ::= { wmanIf2BsSsPkmAuthEntry 3 }
31
32
33
34     wmanIf2BsSsPkmAuthExpiresNew OBJECT-TYPE
35         SYNTAX      DateAndTime
36         MAX-ACCESS  read-only
37         STATUS      current
38         DESCRIPTION
39             "The value of this object is the actual clock time for
40              expiration of the most recent authorization key for this
41              FSM"
42         ::= { wmanIf2BsSsPkmAuthEntry 4 }
43
44
45
46     wmanIf2BsSsPkmAuthLifetime OBJECT-TYPE
47         SYNTAX      Integer32 (86400..6048000)
48         UNITS        "seconds"
49         MAX-ACCESS  read-only
50         STATUS      current
51         DESCRIPTION
52             "The vaue of this object is the lifetime, in seconds, the
53              BS assigns to an authorization key for this SS."
54         REFERENCE
55             "Table 343 in IEEE Std 802.16-2004"
56         DEFVAL      { 604800 }
57         ::= { wmanIf2BsSsPkmAuthEntry 5 }
58
59
60
61     wmanIf2BsSsPkmAuthReset OBJECT-TYPE
62         SYNTAX      INTEGER {noResetRequested(1),
63                             invalidateAuth(2),
64
65

```



```

1         sendAuthInvalid(3),
2         invalidateTeks(4)}
3
4     MAX-ACCESS    read-write
5     STATUS        current
6     DESCRIPTION
7         "Setting this object to invalidateAuth(2) causes the BS to
8         invalidate the current SS authorization key(s), but not to
9         transmit an Authorization Invalid message nor to invalidate
10        unicast TEKs. Setting this object to sendAuthInvalid(3)
11        causes the BS to invalidate the current SS authorization
12        key(s), and to transmit an Authorization Invalid message to
13        the SS, but not to invalidate unicast TEKs. Setting this
14        object to invalidateTeks(4) causes the BS to invalidate the
15        current SS authorization key(s), to transmit an
16        Authorization Invalid message to the SS, and to
17        invalidate all unicast TEKs associated with this SS
18        authorization. Reading this object returns the
19        most-recently-set value of this object, or returns
20        noResetRequested(1) if the object has not been set since
21        the last BS reboot."
22
23 ::= { wmanIf2BsSsPkmAuthEntry 6 }
24
25
26
27 wmanIf2BsSsPkmAuthInfos OBJECT-TYPE
28     SYNTAX        Counter64
29     MAX-ACCESS    read-only
30     STATUS        current
31     DESCRIPTION
32         "The value of this object is the count of times the BS has
33         received an Authentication Information message from this
34         SS."
35
36 ::= { wmanIf2BsSsPkmAuthEntry 7 }
37
38
39
40 wmanIf2BsSsPkmAuthRequests OBJECT-TYPE
41     SYNTAX        Counter64
42     MAX-ACCESS    read-only
43     STATUS        current
44     DESCRIPTION
45         "The value of this object is the count of times the BS has
46         received an Authorization Request message from this SS."
47
48 ::= { wmanIf2BsSsPkmAuthEntry 8 }
49
50
51
52 wmanIf2BsSsPkmAuthReplies OBJECT-TYPE
53     SYNTAX        Counter64
54     MAX-ACCESS    read-only
55     STATUS        current
56     DESCRIPTION
57         "The value of this object is the count of times the BS has
58         transmitted an Authorization Reply message to this SS."
59
60 ::= { wmanIf2BsSsPkmAuthEntry 9 }
61
62
63
64 wmanIf2BsSsPkmAuthRejects OBJECT-TYPE
65     SYNTAX        Counter64
66     MAX-ACCESS    read-only
67     STATUS        current

```

DESCRIPTION

"The value of this object is the count of times the BS has transmitted an Authorization Reject message to this SS."
 ::= { wmanIf2BsSsPkmAuthEntry 10 }

wmanIf2BsSsPkmAuthInvalids OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of this object is the count of times the BS has transmitted an Authorization Invalid message to this SS."
 ::= { wmanIf2BsSsPkmAuthEntry 11 }

wmanIf2BsSsPkmAuthRejectErrorCode OBJECT-TYPE

SYNTAX INTEGER {noInformation(0),
 unauthorizedSs(1),
 unauthorizedSaid(2),
 permanentAuthorizationFailure(6)}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of this object is the enumerated description of the Error-Code in most recent Authorization Reject message transmitted to the SS."

REFERENCE

"IEEE Std 802.16-2004; Table 371"
 ::= { wmanIf2BsSsPkmAuthEntry 12 }

wmanIf2BsSsPkmAuthRejectErrorString OBJECT-TYPE

SYNTAX SnmpAdminString (SIZE (0..128))

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of this object is the Display-String in most recent Authorization Reject message transmitted to the SS. This is a zero length string if no Authorization Reject message has been transmitted to the SS."
 ::= { wmanIf2BsSsPkmAuthEntry 13 }

wmanIf2BsSsPkmAuthInvalidErrorCode OBJECT-TYPE

SYNTAX INTEGER {noInformation(0),
 unauthorizedSs(1),
 unsolicited(3),
 invalidKeySequence(4),
 keyRequestAuthenticationFailure(5)}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of this object is the enumerated description of the Error-Code in most recent Authorization Invalid message transmitted to the SS."

REFERENCE

"IEEE Std 802.16-2004; Table 371"

```

1      ::= { wmanIf2BsSsPkmAuthEntry 14 }
2
3
4  wmanIf2BsSsPkmAuthInvalidErrorString OBJECT-TYPE
5      SYNTAX      SnmpAdminString (SIZE (0..128))
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "The value of this object is the Display-String in most
10         recent Authorization Invalid message transmitted to the SS.
11         This is a zero length string if no Authorization Invalid
12         message has been transmitted to the SS."
13
14      ::= { wmanIf2BsSsPkmAuthEntry 15 }
15
16
17  wmanIf2BsSsPkmAuthPrimarySAId OBJECT-TYPE
18      SYNTAX      INTEGER (0..65535)
19      MAX-ACCESS  read-only
20      STATUS      current
21      DESCRIPTION
22          "The value of this object is the Primary Security
23         Association identifier."
24
25      REFERENCE
26          "IEEE Std 802.16-2004; 11.9.7"
27
28      ::= { wmanIf2BsSsPkmAuthEntry 16 }
29
30  wmanIf2BsSsPkmAuthValidStatus OBJECT-TYPE
31      SYNTAX      INTEGER {unknown (0),
32                          validSsChained (1),
33                          validSsTrusted (2),
34                          invalidSsUntrusted (3),
35                          invalidCAUntrusted (4),
36                          invalidSsOther (5),
37                          invalidCAOther (6)}
38
39      MAX-ACCESS  read-only
40      STATUS      current
41      DESCRIPTION
42          "Contains the reason why a SS's certificate is deemed valid
43         or invalid. Return unknown if the SS is running PKM mode.
44         ValidSsChained means the certificate is valid because it
45         chains to a valid certificate. ValidSsTrusted means the
46         certificate is valid because it has been provisioned to be
47         trusted. InvalidSsUntrusted means the certificate is
48         invalid because it has been provisioned to be untrusted.
49         InvalidCAUntrusted means the certificate is invalid
50         because it chains to an untrusted certificate.
51         InvalidSsOther and InvalidCAOther refer to errors in
52         parsing, validity periods, etc, which are attributable to
53         the SS certificate or its chain respectively."
54
55      ::= { wmanIf2BsSsPkmAuthEntry 17 }
56
57
58
59  --
60  -- Table wmanIf2BsPkmTekTable
61  --
62
63  wmanIf2BsPkmTekTable OBJECT-TYPE
64      SYNTAX      SEQUENCE OF WmanIf2BsPkmTekEntry
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table describes the attributes of each Traffic
5          Encryption Key (TEK) association. The BS maintains one TEK
6          association per SAID on each BS wireless interface."
7      ::= { wmanIf2BsPkmObjects 3 }
8
9
10
11  wmanIf2BsPkmTekEntry OBJECT-TYPE
12      SYNTAX          WmanIf2BsPkmTekEntry
13      MAX-ACCESS      not-accessible
14      STATUS          current
15      DESCRIPTION
16          "Each entry contains objects describing attributes of one
17          TEK association on a particular BS wireless interface. The
18          BS MUST create one entry per SAID per wireless interface,
19          based on the receipt of a Key Request message, and MUST not
20          delete the entry before the SS authorization for the SAID
21          permanently expires."
22      INDEX            { ifIndex, wmanIf2BsPkmTekSAId }
23      ::= { wmanIf2BsPkmTekTable 1 }
24
25
26
27  WmanIf2BsPkmTekEntry ::= SEQUENCE {
28      wmanIf2BsPkmTekSAId                INTEGER,
29      wmanIf2BsPkmTekSAType               INTEGER,
30      wmanIf2BsPkmTekDataEncryptAlg      WmanIf2DataEncryptAlgId,
31      wmanIf2BsPkmTekDataAuthAlg         WmanIf2DataAuthAlgId,
32      wmanIf2BsPkmTekEncryptAlg          WmanIf2TekEncryptAlgId,
33      wmanIf2BsPkmTekLifetime             Integer32,
34      wmanIf2BsPkmTekKeySequenceNumber   Integer32,
35      wmanIf2BsPkmTekExpiresOld           DateAndTime,
36      wmanIf2BsPkmTekExpiresNew          DateAndTime,
37      wmanIf2BsPkmTekReset                TruthValue,
38      wmanIf2BsPkmKeyRequests             Counter32,
39      wmanIf2BsPkmKeyReplies              Counter32,
40      wmanIf2BsPkmKeyRejects              Counter32,
41      wmanIf2BsPkmTekInvalids             Counter32,
42      wmanIf2BsPkmKeyRejectErrorCode       INTEGER,
43      wmanIf2BsPkmKeyRejectErrorString    SnmpAdminString,
44      wmanIf2BsPkmTekInvalidErrorCode     INTEGER,
45      wmanIf2BsPkmTekInvalidErrorString   SnmpAdminString}
46
47
48
49
50
51  wmanIf2BsPkmTekSAId OBJECT-TYPE
52      SYNTAX          INTEGER (0..65535)
53      MAX-ACCESS      not-accessible
54      STATUS          current
55      DESCRIPTION
56          "The value of this object is the Security Association
57          ID (SAID)."
58      REFERENCE
59          "IEEE Std 802.16-2004; 11.9.7"
60      ::= { wmanIf2BsPkmTekEntry 1 }
61
62
63
64  wmanIf2BsPkmTekSAType OBJECT-TYPE
65

```

```

1      SYNTAX      INTEGER {primarySA(0),
2                      staticSA(1),
3                      dynamicSA(2)}
4
5      MAX-ACCESS   read-only
6      STATUS      current
7      DESCRIPTION
8          "The value of this object is the type of security
9          association. Dynamic does not apply to SSs running in PKM
10         mode."
11
12     REFERENCE
13         "IEEE Std 802.16-2004; subclause 11.9.18"
14     ::= { wmanIf2BSPkmTekEntry 2 }
15
16 wmanIf2BSPkmTekDataEncryptAlg OBJECT-TYPE
17     SYNTAX      WmanIf2DataEncryptAlgId
18     MAX-ACCESS   read-only
19     STATUS      current
20     DESCRIPTION
21         "The value of this object is the data encryption algorithm
22         being utilized."
23     REFERENCE
24         "Table 375, IEEE Std 802.16-2004"
25     ::= { wmanIf2BSPkmTekEntry 3 }
26
27 wmanIf2BSPkmTekDataAuthentAlg OBJECT-TYPE
28     SYNTAX      WmanIf2DataAuthAlgId
29     MAX-ACCESS   read-only
30     STATUS      current
31     DESCRIPTION
32         "The value of this object is the data authentication
33         algorithm being utilized."
34     REFERENCE
35         "Table 376, IEEE Std 802.16-2004"
36     ::= { wmanIf2BSPkmTekEntry 4 }
37
38 wmanIf2BSPkmTekEncryptAlg OBJECT-TYPE
39     SYNTAX      WmanIf2TekEncryptAlgId
40     MAX-ACCESS   read-only
41     STATUS      current
42     DESCRIPTION
43         "The value of this object is the TEK key encryption
44         algorithm being utilized."
45     REFERENCE
46         "Table 377, IEEE Std 802.16-2004"
47     ::= { wmanIf2BSPkmTekEntry 5 }
48
49 wmanIf2BSPkmTekLifetime OBJECT-TYPE
50     SYNTAX      Integer32 (1800..604800)
51     UNITS        "seconds"
52     MAX-ACCESS   read-only
53     STATUS      current
54     DESCRIPTION
55         "The value of this object is the lifetime, in seconds, the
56         BS assigns to keys for this TEK association."
57
58
59
60
61
62
63
64
65

```

```

1      REFERENCE
2          "Table 343 in IEEE Std 802.16-2004"
3      DEFVAL      { 43200 }
4      ::= { wmanIf2BsPkmTekEntry 6 }
5
6
7      wmanIf2BsPkmTekKeySequenceNumber OBJECT-TYPE
8          SYNTAX      Integer32 (0..3)
9          MAX-ACCESS  read-only
10         STATUS      current
11         DESCRIPTION
12             "The value of this object is the most recent TEK key
13              sequence number for this SAID."
14         REFERENCE
15             "IEEE Std 802.16-2004; subclause 11.9.5"
16         ::= { wmanIf2BsPkmTekEntry 7 }
17
18
19
20     wmanIf2BsPkmTekExpiresOld OBJECT-TYPE
21         SYNTAX      DateAndTime
22         MAX-ACCESS  read-only
23         STATUS      current
24         DESCRIPTION
25             "The value of this object is the actual clock time for
26              expiration of the immediate predecessor of the most recent
27              TEK for this FSM. If this FSM has only one TEK, then the
28              value is the time of activation of this FSM."
29         ::= { wmanIf2BsPkmTekEntry 8 }
30
31
32
33     wmanIf2BsPkmTekExpiresNew OBJECT-TYPE
34         SYNTAX      DateAndTime
35         MAX-ACCESS  read-only
36         STATUS      current
37         DESCRIPTION
38             "The value of this object is the actual clock time for
39              expiration of the most recent TEK for this FSM."
40         ::= { wmanIf2BsPkmTekEntry 9 }
41
42
43
44     wmanIf2BsPkmTekReset OBJECT-TYPE
45         SYNTAX      TruthValue
46         MAX-ACCESS  read-write
47         STATUS      current
48         DESCRIPTION
49             "Setting this object to TRUE causes the BS to invalidate
50              the current active TEK(s) (plural due to key transition
51              periods), and to generate a new TEK for the associated
52              SAID; the BS MAY also generate an unsolicited TEK Invalid
53              message, to optimize the TEK synchronization between the BS
54              and the SS. Reading this object always returns FALSE."
55         ::= { wmanIf2BsPkmTekEntry 10 }
56
57
58
59
60     wmanIf2BsPkmKeyRequests OBJECT-TYPE
61         SYNTAX      Counter32
62         MAX-ACCESS  read-only
63         STATUS      current
64         DESCRIPTION
65

```

```

1         "The value of this object is the count of times the BS has
2         received a Key Request message."
3         ::= { wmanIf2BSPkmTekEntry 11 }
4
5
6 wmanIf2BSPkmKeyReplies OBJECT-TYPE
7     SYNTAX      Counter32
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "The value of this object is the count of times the BS has
12        transmitted a Key Reply message."
13        ::= { wmanIf2BSPkmTekEntry 12 }
14
15
16 wmanIf2BSPkmKeyRejects OBJECT-TYPE
17     SYNTAX      Counter32
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21        "The value of this object is the count of times the BS has
22        transmitted a Key Reject message."
23        ::= { wmanIf2BSPkmTekEntry 13 }
24
25
26 wmanIf2BSPkmTekInvalids OBJECT-TYPE
27     SYNTAX      Counter32
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31        "The value of this object is the count of times the BS has
32        transmitted a TEK Invalid message."
33        ::= { wmanIf2BSPkmTekEntry 14 }
34
35
36 wmanIf2BSPkmKeyRejectErrorCode OBJECT-TYPE
37     SYNTAX      INTEGER {noInformation(0),
38                        unauthorizedSaid(2)}
39     MAX-ACCESS  read-only
40     STATUS      current
41     DESCRIPTION
42        "The value of this object is the enumerated; description of
43        the Error-Code in the most recent Key Reject message sent
44        in response to a Key Request for this SAID."
45     REFERENCE
46        "IEEE Std 802.16-2004; Table 371"
47     ::= { wmanIf2BSPkmTekEntry 15 }
48
49
50 wmanIf2BSPkmKeyRejectErrorString OBJECT-TYPE
51     SYNTAX      SnmpAdminString (SIZE (0..128))
52     MAX-ACCESS  read-only
53     STATUS      current
54     DESCRIPTION
55        "The value of this object is the Display-String in the most
56        recent Key Reject message sent in response to a Key Request
57        for this SAID. This is a zero length string if no Key
58        Reject message has been received since reboot."
59     ::= { wmanIf2BSPkmTekEntry 16 }
60
61
62
63
64
65

```

```

1
2 wmanIf2BsPkmTekInvalidErrorCode OBJECT-TYPE
3     SYNTAX      INTEGER {noInformation(0),
4                     invalidKeySequence(4)}
5
6     MAX-ACCESS  read-only
7     STATUS      current
8     DESCRIPTION
9         "The value of this object is the enumerated description of
10        the Error-Code in the most recent TEK Invalid message sent
11        in association with this SAID."
12
13     REFERENCE
14         "IEEE Std 802.16-2004; Table 371"
15     ::= { wmanIf2BsPkmTekEntry 17 }
16
17 wmanIf2BsPkmTekInvalidErrorString OBJECT-TYPE
18     SYNTAX      SnmpAdminString (SIZE (0..128))
19     MAX-ACCESS  read-only
20     STATUS      current
21     DESCRIPTION
22         "The value of this object is the Display-String in the most
23        recent TEK Invalid message sent in association with this
24        SAID. This is a zero length string if no TEK Invalid
25        message has been received since reboot."
26     ::= { wmanIf2BsPkmTekEntry 18 }
27
28
29 --
30 -- Base station Notification Group
31 -- wmanIf2BsNotificationObjects contains the BS SNMP Trap objects
32 --
33 wmanIf2BsNotification OBJECT IDENTIFIER ::= { wmanIf2BsObjects 4 }
34 wmanIf2BsTrapControl OBJECT IDENTIFIER ::= { wmanIf2BsNotification 1 }
35
36 wmanIf2BsTrapDefinitions OBJECT IDENTIFIER ::= { wmanIf2BsNotification 2 }
37
38 -- This object groups all NOTIFICATION-TYPE objects for BS.
39 -- It is defined following RFC2758 sections 8.5 and 8.6
40 -- for the compatibility with SNMPv1.
41 wmanIf2BsTrapPrefix OBJECT IDENTIFIER ::= { wmanIf2BsTrapDefinitions 0 }
42
43 wmanIf2BsTrapControlRegister OBJECT-TYPE
44     SYNTAX      BITS {wmanIf2BsSsStatusNotification (0),
45                     wmanIf2BsSsDynamicServiceFail (1),
46                     wmanIf2BsSsRssiStatusChange (2),
47                     wmanIf2BsSsRegistrer (3),
48                     wmanIf2BsSsPkmFail (4),
49                     wmanIf2BsSsDynamicServiceFail2 (5),
50                     wmanIf2BsSsRegister2Trap (6)}
51
52     MAX-ACCESS  read-write
53     STATUS      current
54     DESCRIPTION
55         "The object is used to enable or disable Base Station traps.
56        From left to right, the set bit indicates the corresponding
57        Base Station trap is enabled."
58
59
60
61
62
63
64
65

```



```

1      ::= { wmanIf2BsTrapControl 1 }
2
3
4  wmanIf2BsStatusTrapControlRegister OBJECT-TYPE
5      SYNTAX      BITS {unused(0),
6                  ssInitRangingSucc(1),
7                  ssInitRangingFail(2),
8                  ssRegistered(3),
9                  ssRegistrationFail(4),
10                 ssDeregistered(5),
11                 ssBasicCapabilitySucc(6),
12                 ssBasicCapabilityFail(7),
13                 ssAuthorizationSucc(8),
14                 ssAuthorizationFail(9),
15                 tftpSucc(10),
16                 tftpFail(11),
17                 sfCreationSucc(12),
18                 sfCreationFail(13)}
19
20
21  MAX-ACCESS read-write
22  STATUS      current
23  DESCRIPTION
24      "The object is used to enable or disable Base Station status
25      notification traps. The set bit indicates the corresponding
26      Base Station trap is enabled."
27
28  ::= { wmanIf2BsTrapControl 2 }
29
30
31  --
32  -- BS threshold Definitions
33  --
34
35  wmanIf2BsThresholdConfigTable OBJECT-TYPE
36      SYNTAX      SEQUENCE OF WmanIf2BsThresholdConfigEntry
37      MAX-ACCESS  not-accessible
38      STATUS      current
39      DESCRIPTION
40          "This table contains threshold objects that can be set
41          to detect the threshold crossing events."
42
43  ::= { wmanIf2BsTrapControl 3 }
44
45
46  wmanIf2BsThresholdConfigEntry OBJECT-TYPE
47      SYNTAX      WmanIf2BsThresholdConfigEntry
48      MAX-ACCESS  not-accessible
49      STATUS      current
50      DESCRIPTION
51          "This table provides one row for each BS sector, and is
52          indexed by ifIndex."
53
54  INDEX          { ifIndex }
55  ::= { wmanIf2BsThresholdConfigTable 1 }
56
57
58  WmanIf2BsThresholdConfigEntry ::= SEQUENCE {
59      wmanIf2BsRssiLowThreshold      Integer32,
60      wmanIf2BsRssiHighThreshold     Integer32}
61
62
63  wmanIf2BsRssiLowThreshold OBJECT-TYPE
64      SYNTAX      Integer32
65      UNITS       "dBm"

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "Low threshold for generating the RSSI alarm."
5      ::= { wmanIf2BsThresholdConfigEntry 1 }
6
7
8  wmanIf2BsRssiHighThreshold OBJECT-TYPE
9      SYNTAX      Integer32
10     UNITS       "dBm"
11     MAX-ACCESS  read-write
12     STATUS      current
13     DESCRIPTION
14         "High threshold for clearing the RSSI alarm."
15     ::= { wmanIf2BsThresholdConfigEntry 2 }
16
17
18
19  --
20  -- Subscriber station Notification Objects Definitions
21  --
22
23  wmanIf2BsSsNotificationObjectsTable OBJECT-TYPE
24      SYNTAX      SEQUENCE OF WmanIf2BsSsNotificationObjectsEntry
25      MAX-ACCESS  not-accessible
26      STATUS      current
27      DESCRIPTION
28          "This table contains SS notification objects that have been
29           reported by the trap."
30      ::= { wmanIf2BsTrapDefinitions 1 }
31
32
33  wmanIf2BsSsNotificationObjectsEntry OBJECT-TYPE
34      SYNTAX      WmanIf2BsSsNotificationObjectsEntry
35      MAX-ACCESS  not-accessible
36      STATUS      current
37      DESCRIPTION
38          "This table provides one row for each SS that has
39           generated traps, and is double indexed by
40           wmanIf2BsSsNotificationMacAddr and ifIndex for BS sector."
41      INDEX       { ifIndex, wmanIf2BsSsNotificationMacAddr }
42      ::= { wmanIf2BsSsNotificationObjectsTable 1 }
43
44
45
46  WmanIf2BsSsNotificationObjectsEntry ::= SEQUENCE {
47      wmanIf2BsSsNotificationMacAddr      MacAddress,
48      wmanIf2BsSsStatusValue              INTEGER,
49      wmanIf2BsSsStatusInfo               OCTET STRING,
50      wmanIf2BsDynamicServiceType         INTEGER,
51      wmanIf2BsDynamicServiceFailReason   OCTET STRING,
52      wmanIf2BsSsRssiStatus               INTEGER,
53      wmanIf2BsSsRssiStatusInfo           OCTET STRING,
54      wmanIf2BsSsRegisterStatus           INTEGER,
55      wmanIf2BsDynamicServiceFailSfid     Unsigned32}
56
57
58
59  wmanIf2BsSsNotificationMacAddr OBJECT-TYPE
60      SYNTAX      MacAddress
61      MAX-ACCESS  read-only
62      STATUS      current
63      DESCRIPTION
64
65

```

```

1         "The MAC address of the SS, reporing the notofiation."
2         ::= { wmanIf2BsSsNotificationObjectsEntry 1 }
3
4
5 wmanIf2BsSsStatusValue OBJECT-TYPE
6     SYNTAX      INTEGER {ssInitRangingSucc(1),
7                     ssInitRangingFail(2),
8                     ssRegistered(3),
9                     ssRegistrationFail(4),
10                    ssDeregistered(5),
11                    ssBasicCapabilitySucc(6),
12                    ssBasicCapabilityFail(7),
13                    ssAuthorizationSucc(8),
14                    ssAuthorizationFail(9),
15                    tftpSucc(10),
16                    tftpFail(11),
17                    sfCreationSucc(12),
18                    sfCreationFail(13)}
19
20     MAX-ACCESS  read-only
21     STATUS      current
22     DESCRIPTION
23         "This object indicates the status of a SS, as it goes
24         through network entry and initialization procedure."
25     ::= { wmanIf2BsSsNotificationObjectsEntry 2 }
26
27
28
29
30 wmanIf2BsSsStatusInfo OBJECT-TYPE
31     SYNTAX      OCTET STRING (SIZE(0..255))
32     MAX-ACCESS  read-only
33     STATUS      current
34     DESCRIPTION
35         "This object indicates the reason of SS's status change."
36     ::= { wmanIf2BsSsNotificationObjectsEntry 3 }
37
38
39
40 wmanIf2BsDynamicServiceType OBJECT-TYPE
41     SYNTAX      INTEGER {bsSfCreationReq(1),
42                     bsSfCreationRsp(2),
43                     bsSfCreationAck(3)}
44
45     MAX-ACCESS  read-only
46     STATUS      current
47     DESCRIPTION
48         "This object indicates the dynamic service flow
49         creation command type."
50     ::= { wmanIf2BsSsNotificationObjectsEntry 4 }
51
52
53
54 wmanIf2BsDynamicServiceFailReason OBJECT-TYPE
55     SYNTAX      OCTET STRING (SIZE(0..255))
56     MAX-ACCESS  read-only
57     STATUS      current
58     DESCRIPTION
59         "This object indicates the reason why the service flow
60         creation has failed."
61     ::= { wmanIf2BsSsNotificationObjectsEntry 5 }
62
63
64
65 wmanIf2BsSsRssiStatus OBJECT-TYPE
66     SYNTAX      INTEGER {bsRssiAlarm(1),

```

```

1          bsRssiNoAlarm(2) }
2      MAX-ACCESS    read-only
3      STATUS        current
4      DESCRIPTION
5          "A RSSI alarm is generated when RSSI becomes lower than
6          wmanIf2BsLowRssiThreshold and is cleared when RSSI becomes
7          higher than wmanIf2BsLowRssiThreshold."
8      ::= { wmanIf2BsSsNotificationObjectsEntry 6 }
9
10
11
12      wmanIf2BsSsRssiStatusInfo OBJECT-TYPE
13          SYNTAX      OCTET STRING (SIZE(0..255))
14          MAX-ACCESS  read-only
15          STATUS      current
16          DESCRIPTION
17              "This object indicates the reason why RSSI alarm is
18              generated."
19          ::= { wmanIf2BsSsNotificationObjectsEntry 7 }
20
21
22
23      wmanIf2BsSsRegisterStatus OBJECT-TYPE
24          SYNTAX      INTEGER {ssRegister(1),
25                          ssDeregister(2)}
26          MAX-ACCESS  read-only
27          STATUS      current
28          DESCRIPTION
29              "This object indicates the status of SS registration."
30          ::= { wmanIf2BsSsNotificationObjectsEntry 8 }
31
32
33
34      wmanIf2BsDynamicServiceFailSfid OBJECT-TYPE
35          SYNTAX      Unsigned32 (1..4294967295)
36          MAX-ACCESS  read-only
37          STATUS      current
38          DESCRIPTION
39              "This object identifies the dynamic service flow
40              for notification purposes."
41          ::= { wmanIf2BsSsNotificationObjectsEntry 9 }
42
43
44      --
45      -- Subscriber station Notification Trap Definitions
46      --
47
48      wmanIf2BsSsStatusNotificationTrap NOTIFICATION-TYPE
49          OBJECTS      {ifIndex,
50                          wmanIf2BsSsNotificationMacAddr,
51                          wmanIf2BsSsStatusValue,
52                          wmanIf2BsSsStatusInfo}
53          STATUS      current
54          DESCRIPTION
55              "This trap reports the status of a SS. Based on this
56              notification the NMS will issue an alarm with certain
57              severity depending on the status and the reason received."
58          ::= { wmanIf2BsTrapPrefix 1 }
59
60
61
62      wmanIf2BsSsDynamicServiceFailTrap NOTIFICATION-TYPE
63          OBJECTS      {ifIndex,
64                          wmanIf2BsSsNotificationMacAddr,
65

```

```

1           wmanIf2BsDynamicServiceType,
2           wmanIf2BsDynamicServiceFailReason}
3 STATUS      deprecated
4 DESCRIPTION
5     "Trap deprecated due to limited value without object
6     reporting SFID of victim service flow.
7     An event to report the failure of a dynamic service
8     operation happened during the dynamic services process
9     and detected in the Bs side."
10    ::= { wmanIf2BsTrapPrefix 2 }
11
12 wmanIf2BsSsRssiStatusChangeTrap NOTIFICATION-TYPE
13     OBJECTS      {ifIndex,
14                   wmanIf2BsSsNotificationMacAddr,
15                   wmanIf2BsSsRssiStatus,
16                   wmanIf2BsSsRssiStatusInfo}
17 STATUS      current
18 DESCRIPTION
19     "An event to report that the uplink RSSI is below
20     wmanIf2BsLowRssiThreshold, or above
21     wmanIf2BsHighRssiThreshold after restore."
22    ::= { wmanIf2BsTrapPrefix 3 }
23
24 wmanIf2BsSsPkmFailTrap NOTIFICATION-TYPE
25     OBJECTS      {wmanIf2BsSsNotificationMacAddr}
26 STATUS      current
27 DESCRIPTION
28     "An event to report the failure of a Pkm operation."
29    ::= { wmanIf2BsTrapPrefix 4 }
30
31 wmanIf2BsSsRegistrerTrap NOTIFICATION-TYPE
32     OBJECTS      {wmanIf2BsSsNotificationMacAddr,
33                   wmanIf2BsSsRegisterStatus}
34 STATUS      deprecated
35 DESCRIPTION
36     "Trap deprecated due to limited value without object ifIndex
37     reported.
38     An event to report SS registration status."
39    ::= { wmanIf2BsTrapPrefix 5 }
40
41 wmanIf2BsSsDynamicServiceFail2Trap NOTIFICATION-TYPE
42     OBJECTS      {ifIndex,
43                   wmanIf2BsSsNotificationMacAddr,
44                   wmanIf2BsDynamicServiceType,
45                   wmanIf2BsDynamicServiceFailReason,
46                   wmanIf2BsDynamicServiceFailSfid}
47 STATUS      current
48 DESCRIPTION
49     "An event reporting failure of DSx operation for a service
50     flow identified by wmanIf2BsDynamicServiceFailSfid and
51     detected in the Bs side."
52    ::= { wmanIf2BsTrapPrefix 6 }
53
54 wmanIf2BsSsRegister2Trap NOTIFICATION-TYPE

```

```

1      OBJECTS      {ifIndex,
2                    wmanIf2BsSsNotificationMacAddr,
3                    wmanIf2BsSsRegisterStatus}
4
5      STATUS      current
6
7      DESCRIPTION
8          "An event to report SS registration status for a given sector
9            identified by ifIndex."
10     ::= { wmanIf2BsTrapPrefix 7 }
11
12     --
13     -- Base station PHY Group
14     --
15     wmanIf2BsPhy OBJECT IDENTIFIER ::= { wmanIf2BsObjects 6 }
16
17     --
18     -- BS OFDM PHY objects
19     --
20     --
21     wmanIf2BsOfdmPhy OBJECT IDENTIFIER ::= { wmanIf2BsPhy 1 }
22
23     wmanIf2BsOfdmUplinkChannelTable OBJECT-TYPE
24         SYNTAX      SEQUENCE OF WmanIf2BsOfdmUplinkChannelEntry
25         MAX-ACCESS  not-accessible
26         STATUS      current
27         DESCRIPTION
28             "This table contains UCD channel attributes, defining the
29               transmission characteristics of uplink channels"
30         REFERENCE
31             "Table 349 and Table 352, in IEEE Std 802.16-2004"
32         ::= { wmanIf2BsOfdmPhy 1 }
33
34     wmanIf2BsOfdmUplinkChannelEntry OBJECT-TYPE
35         SYNTAX      WmanIf2BsOfdmUplinkChannelEntry
36         MAX-ACCESS  not-accessible
37         STATUS      current
38         DESCRIPTION
39             "This table provides one row for each uplink channel of
40               multi-sector BS, and is indexed by BS ifIndex. An entry
41               in this table exists for each ifEntry of BS with an
42               ifType of propBWAmp2Mp."
43         INDEX { ifIndex }
44         ::= { wmanIf2BsOfdmUplinkChannelTable 1 }
45
46     WmanIf2BsOfdmUplinkChannelEntry ::= SEQUENCE {
47         wmanIf2BsOfdmCtBasedResvTimeout      INTEGER,
48         wmanIf2BsOfdmBwReqOppSize             INTEGER,
49         wmanIf2BsOfdmRangReqOppSize           INTEGER,
50         wmanIf2BsOfdmUplinkCenterFreq        Unsigned32,
51         wmanIf2BsOfdmNumSubChReqRegionFull    INTEGER,
52         wmanIf2BsOfdmNumSymbolsReqRegionFull  INTEGER,
53         wmanIf2BsOfdmSubChFocusCtCode         INTEGER,
54         wmanIf2BsOfdmUpLinkChannelId          INTEGER}
55
56     wmanIf2BsOfdmCtBasedResvTimeout OBJECT-TYPE
57         SYNTAX      INTEGER (1..255)
58
59

```

```

1      MAX-ACCESS    read-write
2      STATUS        current
3      DESCRIPTION
4          "The number of UL-MAPs to receive before contention-based
5          reservation is attempted again for the same connection."
6      REFERENCE
7          "Table 349, in IEEE Std 802.16-2004"
8      ::= { wmanIf2BsOfdmUplinkChannelEntry 1 }
9
10
11
12  wmanIf2BsOfdmBwReqOppSize OBJECT-TYPE
13      SYNTAX          INTEGER (1..65535)
14      UNITS            "PS"
15      MAX-ACCESS      read-write
16      STATUS          current
17      DESCRIPTION
18          "Size (in units of PS) of PHY payload that SS may use to
19          format and transmit a bandwidth request message in a
20          contention request opportunity. The value includes all
21          PHY overhead as well as allowance for the MAC data the
22          message may hold."
23      REFERENCE
24          "Table 349, in IEEE Std 802.16-2004"
25      ::= { wmanIf2BsOfdmUplinkChannelEntry 2 }
26
27
28
29
30  wmanIf2BsOfdmRangReqOppSize OBJECT-TYPE
31      SYNTAX          INTEGER (1..65535)
32      UNITS            "PS"
33      MAX-ACCESS      read-write
34      STATUS          current
35      DESCRIPTION
36          "Size (in units of PS) of PHY payload that SS may use to
37          format and transmit a RNG-REQ message in a contention
38          request opportunity. The value includes all PHY overhead
39          as well as allowance for the MAC data the message may
40          hold and the maximum SS/BS roundtrip propagation delay."
41      REFERENCE
42          "Table 349, in IEEE Std 802.16-2004"
43      ::= { wmanIf2BsOfdmUplinkChannelEntry 3 }
44
45
46
47
48  wmanIf2BsOfdmUplinkCenterFreq OBJECT-TYPE
49      SYNTAX          Unsigned32
50      UNITS            "kHz"
51      MAX-ACCESS      read-write
52      STATUS          current
53      DESCRIPTION
54          " Uplink center frequency (kHz)"
55      REFERENCE
56          "Table 349, in IEEE Std 802.16-2004"
57      ::= { wmanIf2BsOfdmUplinkChannelEntry 4 }
58
59
60
61  wmanIf2BsOfdmNumSubChReqRegionFull OBJECT-TYPE
62      SYNTAX          INTEGER {oneSubchannel(0),
63                        twoSubchannels(1),
64                        fourSubchannels(2),
65

```

```

1         eightSubchannels(3),
2         sixteenSubchannels(4) }
3
4     MAX-ACCESS    read-write
5     STATUS        current
6     DESCRIPTION
7         "Number of subchannels used by each transmit
8         opportunity when REQ Region-Full is allocated in
9         subchannelization region."
10
11     REFERENCE
12         "Table 352, in IEEE Std 802.16-2004"
13     ::= { wmanIf2BsOfdmUplinkChannelEntry 5 }
14
15 wmanIf2BsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
16     SYNTAX        INTEGER (0..31)
17     MAX-ACCESS    read-write
18     STATUS        current
19     DESCRIPTION
20         "Number of OFDM symbols used by each transmit
21         opportunity when REQ Region-Full is allocated in
22         subchannelization region."
23
24     REFERENCE
25         "Table 352, in IEEE Std 802.16-2004"
26     ::= { wmanIf2BsOfdmUplinkChannelEntry 6 }
27
28
29 wmanIf2BsOfdmSubChFocusCtCode OBJECT-TYPE
30     SYNTAX        INTEGER (0..8)
31     MAX-ACCESS    read-write
32     STATUS        current
33     DESCRIPTION
34         "Number of contention codes (CSE) that shall only be used to
35         request a subchannelized allocation. Default value 0.
36         Allowed values 0-8."
37
38     REFERENCE
39         "Table 352, in IEEE Std 802.16-2004"
40     DEFVAL        { 0 }
41     ::= { wmanIf2BsOfdmUplinkChannelEntry 7 }
42
43
44 wmanIf2BsOfdmUpLinkChannelId OBJECT-TYPE
45     SYNTAX        INTEGER (0..255)
46     MAX-ACCESS    read-write
47     STATUS        current
48     DESCRIPTION
49         "The identifier of the uplink channel to which the relevant
50         RNG-RSP or RNG-REQ message refers."
51
52     REFERENCE
53         "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
54     ::= { wmanIf2BsOfdmUplinkChannelEntry 8 }
55
56
57 wmanIf2BsOfdmDownlinkChannelTable OBJECT-TYPE
58     SYNTAX        SEQUENCE OF WmanIf2BsOfdmDownlinkChannelEntry
59     MAX-ACCESS    not-accessible
60     STATUS        current
61     DESCRIPTION
62         "This table contains DCD channel attributes, defining the
63

```



```

1         transmission characteristics of downlink channels"
2     REFERENCE
3         "Table 358, in IEEE Std 802.16-2004"
4     ::= { wmanIf2BsOfdmPhy 2 }
5
6
7     wmanIf2BsOfdmDownlinkChannelEntry OBJECT-TYPE
8         SYNTAX      WmanIf2BsOfdmDownlinkChannelEntry
9         MAX-ACCESS  not-accessible
10        STATUS      current
11        DESCRIPTION
12            "This table provides one row for each downlink channel of
13             multi-sector BS, and is indexed by BS ifIndex. An entry
14             in this table exists for each ifEntry of BS with an
15             ifType of propBWA2Mp."
16        INDEX { ifIndex }
17        ::= { wmanIf2BsOfdmDownlinkChannelTable 1 }
18
19
20
21     WmanIf2BsOfdmDownlinkChannelEntry ::= SEQUENCE {
22         wmanIf2BsOfdmBsEIRP          INTEGER,
23         wmanIf2BsOfdmChannelNumber   WmanIf2ChannelNumber,
24         wmanIf2BsOfdmTTG             INTEGER,
25         wmanIf2BsOfdmRTG             INTEGER,
26         wmanIf2BsOfdmInitRngMaxRSS   INTEGER,
27         wmanIf2BsOfdmDownlinkCenterFreq Unsigned32,
28         wmanIf2BsOfdmBsId            WmanIf2BsIdType,
29         wmanIf2BsOfdmMacVersion       WmanIf2MacVersion,
30         wmanIf2BsOfdmFrameDurationCode INTEGER,
31         wmanIf2BsOfdmDownLinkChannelId INTEGER}
32
33
34
35     wmanIf2BsOfdmBsEIRP OBJECT-TYPE
36         SYNTAX      INTEGER (-32768..32767)
37         UNITS       "dBm"
38         MAX-ACCESS  read-write
39         STATUS      current
40         DESCRIPTION
41             "The EIRP is the equivalent isotropic radiated power of
42              the base station, which is computed for a simple
43              single-antenna transmitter."
44         REFERENCE
45             "Table 358, in IEEE Std 802.16-2004"
46         ::= { wmanIf2BsOfdmDownlinkChannelEntry 1 }
47
48
49
50
51     wmanIf2BsOfdmChannelNumber OBJECT-TYPE
52         SYNTAX      WmanIf2ChannelNumber
53         MAX-ACCESS  read-write
54         STATUS      current
55         DESCRIPTION
56             "Downlink channel number as defined in 8.5.
57              Used for license-exempt operation only."
58         REFERENCE
59             "Table 358, in IEEE Std 802.16-2004"
60         ::= { wmanIf2BsOfdmDownlinkChannelEntry 2 }
61
62
63
64     wmanIf2BsOfdmTTG OBJECT-TYPE
65

```

```

1      SYNTAX      INTEGER (0..255)
2      MAX-ACCESS  read-write
3      STATUS      current
4      DESCRIPTION
5          "Transmit / Receive Transition Gap."
6      REFERENCE
7          "Table 358, in IEEE Std 802.16-2004"
8      ::= { wmanIf2BsOfdmDownlinkChannelEntry 3 }
9
10
11
12  wmanIf2BsOfdmRTG OBJECT-TYPE
13      SYNTAX      INTEGER (0..255)
14      MAX-ACCESS  read-write
15      STATUS      current
16      DESCRIPTION
17          "Receive / Transmit Transition Gap."
18      REFERENCE
19          "Table 358, in IEEE Std 802.16-2004"
20      ::= { wmanIf2BsOfdmDownlinkChannelEntry 4 }
21
22
23
24  wmanIf2BsOfdmInitRngMaxRSS OBJECT-TYPE
25      SYNTAX      INTEGER (-32768..32767)
26      UNITS        "dBm"
27      MAX-ACCESS  read-write
28      STATUS      current
29      DESCRIPTION
30          "Initial Ranging Max. equivalent isotropic received power
31          at BS Signed in units of 1 dBm."
32      REFERENCE
33          "Table 358, in IEEE Std 802.16-2004"
34      ::= { wmanIf2BsOfdmDownlinkChannelEntry 5 }
35
36
37
38  wmanIf2BsOfdmDownlinkCenterFreq OBJECT-TYPE
39      SYNTAX      Unsigned32
40      UNITS        "kHz"
41      MAX-ACCESS  read-write
42      STATUS      current
43      DESCRIPTION
44          "Downlink center frequency (kHz)."
45      REFERENCE
46          "Table 358, in IEEE Std 802.16-2004"
47      ::= { wmanIf2BsOfdmDownlinkChannelEntry 6 }
48
49
50
51  wmanIf2BsOfdmBsId OBJECT-TYPE
52      SYNTAX      WmanIf2BsIdType
53      MAX-ACCESS  read-write
54      STATUS      current
55      DESCRIPTION
56          "Base station ID."
57      REFERENCE
58          "Table 358, in IEEE Std 802.16-2004"
59      ::= { wmanIf2BsOfdmDownlinkChannelEntry 7 }
60
61
62
63  wmanIf2BsOfdmMacVersion OBJECT-TYPE
64      SYNTAX      WmanIf2MacVersion
65

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "This parameter specifies the version of 802.16 to which
5          the message originator conforms."
6      REFERENCE
7          "Table 358, in IEEE Std 802.16-2004"
8      ::= { wmanIf2BsOfdmDownlinkChannelEntry 8 }
9
10
11
12  wmanIf2BsOfdmFrameDurationCode OBJECT-TYPE
13      SYNTAX      INTEGER {duration2dot5ms(0),
14                          duration4ms(1),
15                          duration5ms(2),
16                          duration8ms(3),
17                          duration10ms(4),
18                          duration12dot5ms(5),
19                          duration20ms(6)}
20      MAX-ACCESS  read-write
21      STATUS      current
22      DESCRIPTION
23          "The duration of the frame. The frame duration code
24          values are specified in Table 230."
25      REFERENCE
26          "Subclause 11.4.1, Table 230, in IEEE Std 802.16-2004"
27      ::= { wmanIf2BsOfdmDownlinkChannelEntry 9 }
28
29
30
31
32  wmanIf2BsOfdmDownLinkChannelId OBJECT-TYPE
33      SYNTAX      INTEGER (0..255)
34      MAX-ACCESS  read-write
35      STATUS      current
36      DESCRIPTION
37          "The identifier of the downlink channel to which this
38          message refers."
39      REFERENCE
40          "Subclause 6.3.2.3.1. Table 15, in IEEE Std 802.16-2004"
41      ::= { wmanIf2BsOfdmDownlinkChannelEntry 10 }
42
43
44
45  wmanIf2BsOfdmUcdBurstProfileTable OBJECT-TYPE
46      SYNTAX      SEQUENCE OF WmanIf2BsOfdmUcdBurstProfileEntry
47      MAX-ACCESS  not-accessible
48      STATUS      current
49      DESCRIPTION
50          "This table contains UCD burst profiles for each uplink
51          channel"
52      REFERENCE
53          "Table 356, in IEEE Std 802.16-2004"
54      ::= { wmanIf2BsOfdmPhy 3 }
55
56
57
58  wmanIf2BsOfdmUcdBurstProfileEntry OBJECT-TYPE
59      SYNTAX      WmanIf2BsOfdmUcdBurstProfileEntry
60      MAX-ACCESS  not-accessible
61      STATUS      current
62      DESCRIPTION
63          "This table provides one row for each UCD burst profile."
64
65

```

```

1           This table is double indexed. The primary index is an
2           ifIndex with an ifType of propBWA2Mp. The secondary index
3           is wmanIf2BsOfdmUiucIndex."
4       INDEX { ifIndex, wmanIf2BsOfdmUiucIndex }
5       ::= { wmanIf2BsOfdmUcdBurstProfileTable 1 }
6
7
8       WmanIf2BsOfdmUcdBurstProfileEntry ::= SEQUENCE {
9           wmanIf2BsOfdmUiucIndex          INTEGER,
10          wmanIf2BsOfdmUcdFecCodeType      WmanIf2OfdmFecCodeType,
11          wmanIf2BsOfdmFocusCtPowerBoost    INTEGER,
12          wmanIf2BsOfdmUcdTcsEnable         INTEGER,
13          wmanIf2BsOfdmUcdBurstProfileRowStatus RowStatus}
14
15
16       wmanIf2BsOfdmUiucIndex OBJECT-TYPE
17       SYNTAX          INTEGER (5 .. 12)
18       MAX-ACCESS      not-accessible
19       STATUS          current
20       DESCRIPTION
21       "The Uplink Interval Usage Code indicates the uplink burst
22       profile in the UCD message, and is used along with ifIndex
23       to identify an entry in the
24       wmanIf2BsOfdmUcdBurstProfileTable."
25       REFERENCE
26       "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
27       ::= { wmanIf2BsOfdmUcdBurstProfileEntry 1 }
28
29
30       wmanIf2BsOfdmUcdFecCodeType OBJECT-TYPE
31       SYNTAX          WmanIf2OfdmFecCodeType
32       MAX-ACCESS      read-create
33       STATUS          current
34       DESCRIPTION
35       "Uplink FEC code type and modulation type"
36       REFERENCE
37       "Table 356, in IEEE Std 802.16-2004"
38       ::= { wmanIf2BsOfdmUcdBurstProfileEntry 2 }
39
40
41       wmanIf2BsOfdmFocusCtPowerBoost OBJECT-TYPE
42       SYNTAX          INTEGER (0 .. 255)
43       MAX-ACCESS      read-create
44       STATUS          current
45       DESCRIPTION
46       "The power boost in dB of focused contention carriers, as
47       described in 8.3.6.3.3."
48       REFERENCE
49       "Table 356, in IEEE Std 802.16-2004"
50       ::= { wmanIf2BsOfdmUcdBurstProfileEntry 3 }
51
52
53       wmanIf2BsOfdmUcdTcsEnable OBJECT-TYPE
54       SYNTAX          INTEGER {tcsDisabled(0),
55                          tcsEnabled(1)}
56       MAX-ACCESS      read-create
57       STATUS          current
58       DESCRIPTION
59       "This parameter determines the transmission convergence

```

sublayer, as described in 8.1.4.3, can be enabled on a per-burst basis for both uplink and downlink. through DIUC/UIUC messages."

REFERENCE

"Table 356, in IEEE Std 802.16-2004"
 ::= { wmanIf2BsOfdmUcdBurstProfileEntry 4 }

wmanIf2BsOfdmUcdBurstProfileRowStatus OBJECT-TYPE

SYNTAX RowStatus
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION

"This object is used to create a new row or modify or delete an existing row in this table.

If the implementator of this MIB has choosen not to implement 'dynamic assignment' of profiles, this object is not useful and should return noSuchName upon SNMP request."

::= { wmanIf2BsOfdmUcdBurstProfileEntry 5 }

wmanIf2BsOfdmDcdBurstProfileTable OBJECT-TYPE

SYNTAX SEQUENCE OF WmanIf2BsOfdmDcdBurstProfileEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION

"This table provides one row for each DCD burst profile. This table is double indexed. The primary index is an ifIndex with an ifType of propBWA2Mp. The secondary index is wmanIf2BsOfdmDiucIndex."

REFERENCE

"Table 362, in IEEE Std 802.16-2004"
 ::= { wmanIf2BsOfdmPhy 4 }

wmanIf2BsOfdmDcdBurstProfileEntry OBJECT-TYPE

SYNTAX WmanIf2BsOfdmDcdBurstProfileEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION

"This table provides one row for each DCD burst profile. This table is double indexed. The primary index is an ifIndex with an ifType of propBWA2Mp. The secondary index is wmanIf2BsOfdmDiucIndex."

INDEX { ifIndex, wmanIf2BsOfdmDiucIndex }
 ::= { wmanIf2BsOfdmDcdBurstProfileTable 1 }

WmanIf2BsOfdmDcdBurstProfileEntry ::= SEQUENCE {

wmanIf2BsOfdmDiucIndex	INTEGER,
wmanIf2BsOfdmDownlinkFrequency	Unsigned32,
wmanIf2BsOfdmDcdFecCodeType	WmanIf2OfdmFecCodeType,
wmanIf2BsOfdmDiucMandatoryExitThresh	INTEGER,
wmanIf2BsOfdmDiucMinEntryThresh	INTEGER,
wmanIf2BsOfdmTcsEnable	INTEGER,
wmanIf2BsOfdmDcdBurstProfileRowStatus	RowStatus}

```

1
2 wmanIf2BsOfdmDiucIndex OBJECT-TYPE
3     SYNTAX      INTEGER (1..11)
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "The Downlink Interval Usage Code indicates the downlink
8         burst profile in the DCD message, and is used along with
9         ifIndex to identify an entry in the
10        wmanIf2BsOfdmDcdBurstProfileTable."
11    REFERENCE
12        "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
13    ::= { wmanIf2BsOfdmDcdBurstProfileEntry 1 }
14
15 wmanIf2BsOfdmDownlinkFrequency OBJECT-TYPE
16     SYNTAX      Unsigned32
17     UNITS       "kHz"
18     MAX-ACCESS  read-create
19     STATUS      current
20     DESCRIPTION
21         "Downlink Frequency (kHz)."

```

```

1         robust DIUC is required, in 0.25 dB units."
2     REFERENCE
3         "Table 362, in IEEE Std 802.16-2004"
4     ::= { wmanIf2BsOfdmDcdBurstProfileEntry 5 }
5
6
7     wmanIf2BsOfdmTcsEnable OBJECT-TYPE
8         SYNTAX      INTEGER {tcsDisabled (0),
9                     tcsEnabled (1)}
10
11         MAX-ACCESS   read-create
12         STATUS       current
13         DESCRIPTION
14             "Indicates whether Transmission CONvergence Sublayer
15              is enabled or disabled."
16         REFERENCE
17             "Table 362, in IEEE Std 802.16-2004"
18         ::= { wmanIf2BsOfdmDcdBurstProfileEntry 6 }
19
20
21     wmanIf2BsOfdmDcdBurstProfileRowStatus OBJECT-TYPE
22         SYNTAX      RowStatus
23         MAX-ACCESS   read-create
24         STATUS       current
25         DESCRIPTION
26             "This object is used to create a new row or modify or
27              delete an existing row in this table.
28
29              If the implementator of this MIB has choosen not
30              to implement 'dynamic assignment' of profiles, this
31              object is not useful and should return noSuchName
32              upon SNMP request."
33         ::= { wmanIf2BsOfdmDcdBurstProfileEntry 7 }
34
35
36     wmanIf2BsOfdmConfigurationTable OBJECT-TYPE
37         SYNTAX      SEQUENCE OF WmanIf2BsOfdmConfigurationEntry
38         MAX-ACCESS   not-accessible
39         STATUS       current
40         DESCRIPTION
41             "This table contains BS configuration objects, specific to
42              OFDM PHY."
43         ::= { wmanIf2BsOfdmPhy 5 }
44
45
46     wmanIf2BsOfdmConfigurationEntry OBJECT-TYPE
47         SYNTAX      WmanIf2BsOfdmConfigurationEntry
48         MAX-ACCESS   not-accessible
49         STATUS       current
50         DESCRIPTION
51             "This table is indexed by ifIndex with an ifType of
52              propBWA2Mp."
53         INDEX { ifIndex }
54         ::= { wmanIf2BsOfdmConfigurationTable 1 }
55
56
57     WmanIf2BsOfdmConfigurationEntry ::= SEQUENCE {
58         wmanIf2BsOfdmMinReqRegionFullTxOpp      INTEGER,
59         wmanIf2BsOfdmMinFocusedCtTxOpp           INTEGER,
60         wmanIf2BsOfdmMaxRoundTripDelay           INTEGER,

```

```

1      wmanIf2BsOfdmRangeAbortTimingThold      INTEGER,
2      wmanIf2BsOfdmRangeAbortPowerThold      INTEGER,
3      wmanIf2BsOfdmRangeAbortFreqThold      INTEGER,
4      wmanIf2BsOfdmDnlnkRateId      INTEGER,
5      wmanIf2BsOfdmRatioG      INTEGER}
6
7
8  wmanIf2BsOfdmMinReqRegionFullTxOpp OBJECT-TYPE
9      SYNTAX      INTEGER (1..65535)
10     UNITS      "1/sec"
11     MAX-ACCESS      read-write
12     STATUS      current
13     DESCRIPTION
14         "The minimum number of Full bandwidth Req-Region Full
15         Transmit opportunities scheduled in the UL per second."
16     REFERENCE
17         "Subclause 6.3.7.4.3 in IEEE Std 802.16-2004"
18     ::= { wmanIf2BsOfdmConfigurationEntry 1 }
19
20
21  wmanIf2BsOfdmMinFocusedCtTxOpp OBJECT-TYPE
22     SYNTAX      INTEGER (0..65535)
23     UNITS      "1/sec"
24     MAX-ACCESS      read-write
25     STATUS      current
26     DESCRIPTION
27         "The minimum number of focused contention Transmit
28         opportunities scheduled in the UL per second. The value may
29         be 0 if the focused contention is not implemented."
30     REFERENCE
31         "Subclauses 6.3.6.4 and 8.3.7.3.3 in IEEE Std 802.16-2004"
32     ::= { wmanIf2BsOfdmConfigurationEntry 2 }
33
34
35  wmanIf2BsOfdmMaxRoundTripDelay OBJECT-TYPE
36     SYNTAX      INTEGER (1..65535)
37     UNITS      "us"
38     MAX-ACCESS      read-write
39     STATUS      current
40     DESCRIPTION
41         "Maximum supported round trip delay.
42         It is required to limit the cell size."
43     REFERENCE
44         "Subclause 8.3.5.1 in IEEE Std 802.16-2004"
45     ::= { wmanIf2BsOfdmConfigurationEntry 3 }
46
47
48  wmanIf2BsOfdmRangeAbortTimingThold OBJECT-TYPE
49     SYNTAX      INTEGER (0..255)
50     UNITS      "1/Fs"
51     MAX-ACCESS      read-write
52     STATUS      current
53     DESCRIPTION
54         "This object defines Tolerable Timing Offset. BS performs
55         Initial Ranging until the SS transmissions are within
56         limits that are deemed tolerable by the BS. If the SS does
57         not transmit within these limits after a number of
58         correction attempts then the BS aborts Initial Ranging."
59
60
61
62
63
64
65

```


REFERENCE

"Figure 63 and Table 365 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsOfdmConfigurationEntry 4 }

wmanIf2BsOfdmRangeAbortPowerThold OBJECT-TYPE

SYNTAX INTEGER (0..255)

UNITS "0.25dB"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This object defines Tolerable Power Offset. BS performs Initial Ranging until the SS transmissions are within limits that are deemed tolerable by the BS. If the SS does not transmit within these limits after a number of correction attempts then the BS aborts Initial Ranging."

REFERENCE

"Figure 63 and Table 365 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsOfdmConfigurationEntry 5 }

wmanIf2BsOfdmRangeAbortFreqThold OBJECT-TYPE

SYNTAX INTEGER (0..255)

UNITS "Hz"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This object defines Tolerable Frequency Offset. BS performs Initial Ranging until the SS transmissions are within limits that are deemed tolerable by the BS. If the SS does not transmit within these limits after a number of correction attempts then the BS aborts Initial Ranging."

REFERENCE

"Figure 63 and Table 365 in IEEE Std 802.16-2004"
 ::= { wmanIf2BsOfdmConfigurationEntry 6 }

wmanIf2BsOfdmDnlnkRateId OBJECT-TYPE

SYNTAX INTEGER { dnlnkRateIdBpsk1Over2(0),
 dnlnkRateIdQpsk1Over2(1),
 dnlnkRateIdQpsk3Over4(2),
 dnlnkRateId16Qam1Over2(3),
 dnlnkRateId16Qam3Over4(4),
 dnlnkRateId64Qam2Over3(5),
 dnlnkRateId64Qam3Over4(6) }

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The Rate ID to be used in the first downlink burst immediately following the FCH. The Rate ID encoding is static and cannot be changed during system operation. The change of the Rate ID should be applied on system re-initialisation (e.g. following sector or BS reset)."

REFERENCE

"Subclause 8.3.3.4.3 in IEEE Std 802.16-2004"
 DEFVAL { dnlnkRateIdBpsk1Over2 }
 ::= { wmanIf2BsOfdmConfigurationEntry 7 }

```

1
2 wmanIf2BsOfdmRatioG OBJECT-TYPE
3     SYNTAX INTEGER {ratio1To4(0),
4                   ratio1To8(1),
5                   ratio1To16(2),
6                   ratio1To32(3)}
7
8     MAX-ACCESS read-write
9     STATUS current
10    DESCRIPTION
11        "The ratio of CP time to 'useful' time.Values
12        are 1/4, 1/8, 1/16 or 1/32."
13    REFERENCE
14        "Subclause 8.3.1.1.1 in IEEE Std 802.16-2004"
15    DEFVAL { ratio1To4 }
16    ::= { wmanIf2BsOfdmConfigurationEntry 8 }
17
18 wmanIf2BsSsOfdmReqCapabilitiesTable OBJECT-TYPE
19     SYNTAX SEQUENCE OF WmanIf2BsSsOfdmReqCapabilitiesEntry
20     MAX-ACCESS not-accessible
21     STATUS current
22     DESCRIPTION
23         "This table contains the basic capability information,
24         specific to OFDM Phy, of SSs that have been reported by
25         SSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.
26         Entries in this table should be created when an SS
27         registers with a BS."
28     ::= { wmanIf2BsOfdmPhy 6 }
29
30 wmanIf2BsSsOfdmReqCapabilitiesEntry OBJECT-TYPE
31     SYNTAX WmanIf2BsSsOfdmReqCapabilitiesEntry
32     MAX-ACCESS not-accessible
33     STATUS current
34     DESCRIPTION
35         "This table provides one row for each SS that has been
36         registered in the BS. This table augments the table
37         wmanIf2BsRegisteredSsTable."
38     AUGMENTS { wmanIf2BsRegisteredSsEntry }
39     ::= { wmanIf2BsSsOfdmReqCapabilitiesTable 1 }
40
41 WmanIf2BsSsOfdmReqCapabilitiesEntry ::= SEQUENCE {
42     wmanIf2BsSsOfdmReqCapFftSizes WmanIf2OfdmFftSizes,
43     wmanIf2BsSsOfdmReqCapSsDemodulator WmanIf2OfdmSsDeModType,
44     wmanIf2BsSsOfdmReqCapSsModulator WmanIf2OfdmSsModType,
45     wmanIf2BsSsOfdmReqCapFocusedCtSupport WmanIf2OfdmFocusedCt,
46     wmanIf2BsSsOfdmReqCapTcSublayerSupport WmanIf2OfdmTcSublayer,
47     wmanIf2eBsSsOfdmReqCapPrivteMapSupport WmanIf2eOfdmPrivMap,
48     wmanIf2eBsSsOfdmReqCapUlPowerControl WmanIf2eOfdmUlPower,
49     wmanIf2eBsSsOfdmReqCapLoopPwrControlSw Unsigned32}
50
51 wmanIf2BsSsOfdmReqCapFftSizes OBJECT-TYPE
52     SYNTAX WmanIf2OfdmFftSizes
53     MAX-ACCESS read-only
54     STATUS current
55     DESCRIPTION
56
57
58
59
60
61
62
63
64
65

```

```

1         "This field indicates the FFT sizes supported by SS.
2         The usage is defined by WmanIf2OfdmFftSizes."
3         ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 1 }
4
5
6 wmanIf2BsSsOfdmReqCapSsDemodulator OBJECT-TYPE
7     SYNTAX      WmanIf2OfdmSsDeModType
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "This field indicates the different demodulator options
12        supported by SS for downlink.
13        The usage is defined by WmanIf2OfdmSsDeModType."
14        ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 2 }
15
16
17
18 wmanIf2BsSsOfdmReqCapSsModulator OBJECT-TYPE
19     SYNTAX      WmanIf2OfdmSsModType
20     MAX-ACCESS  read-only
21     STATUS      current
22     DESCRIPTION
23        "This field indicates the different modulator options
24        supported by SS for uplink.
25        The usage is defined by WmanIf2OfdmSsModType."
26        ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 3 }
27
28
29
30 wmanIf2BsSsOfdmReqCapFocusedCtSupport OBJECT-TYPE
31     SYNTAX      WmanIf2OfdmFocusedCt
32     MAX-ACCESS  read-only
33     STATUS      current
34     DESCRIPTION
35        "This field indicates whether the SS supports Focused
36        Contention. The usage is defined by
37        WmanIf2OfdmFocusedCt."
38        ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 4 }
39
40
41
42 wmanIf2BsSsOfdmReqCapTcSublayerSupport OBJECT-TYPE
43     SYNTAX      WmanIf2OfdmTcSublayer
44     MAX-ACCESS  read-only
45     STATUS      current
46     DESCRIPTION
47        "This field indicates whether or not the SS supports
48        the TC sublayer. The usage is defined by
49        WmanIf2OfdmTcSublayer."
50        ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 5 }
51
52
53
54 wmanIf2eBsSsOfdmReqCapPrivteMapSupport OBJECT-TYPE
55     SYNTAX      WmanIf2eOfdmPrivMap
56     MAX-ACCESS  read-only
57     STATUS      current
58     DESCRIPTION
59        "This field indicates if the private map parameters
60        is supported."
61        ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 6 }
62
63
64
65 wmanIf2eBsSsOfdmReqCapUlPowerControl OBJECT-TYPE

```

```

1      SYNTAX      WmanIf2eOfdmUlPower
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "This field indicates the uplink power control options
6          supported by SS."
7      ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 7 }
8
9
10
11  wmanIf2eBsSsOfdmReqCapLoopPwrControlSw OBJECT-TYPE
12      SYNTAX      Unsigned32
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "This field indicates the minimum number of frames that
17          SS takes to switch from the open loop power control
18          scheme to the closed loop power control scheme or
19          vice versa."
20      ::= { wmanIf2BsSsOfdmReqCapabilitiesEntry 8 }
21
22
23
24  wmanIf2BsSsOfdmRspCapabilitiesTable OBJECT-TYPE
25      SYNTAX      SEQUENCE OF WmanIf2BsSsOfdmRspCapabilitiesEntry
26      MAX-ACCESS  not-accessible
27      STATUS      current
28      DESCRIPTION
29          "This table contains the basic capability information,
30          specific to OFDM Phy, of SSs that have been negotiated
31          and agreed between BS and SS via RNG-REQ/RSP,
32          SBC-REQ/RSP and REG-REQ/RSP messages. This table
33          augments the wmanIf2BsRegisteredSsTable."
34      REFERENCE
35          "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
36      ::= { wmanIf2BsOfdmPhy 7 }
37
38
39
40
41  wmanIf2BsSsOfdmRspCapabilitiesEntry OBJECT-TYPE
42      SYNTAX      WmanIf2BsSsOfdmRspCapabilitiesEntry
43      MAX-ACCESS  not-accessible
44      STATUS      current
45      DESCRIPTION
46          "This table provides one row for each SS that has been
47          registered in the BS. This table augments the
48          wmanIf2BsRegisteredSsTable. "
49      AUGMENTS { wmanIf2BsRegisteredSsEntry }
50      ::= { wmanIf2BsSsOfdmRspCapabilitiesTable 1 }
51
52
53
54  WmanIf2BsSsOfdmRspCapabilitiesEntry ::= SEQUENCE {
55      wmanIf2BsSsOfdmRspCapFftSizes      WmanIf2OfdmFftSizes,
56      wmanIf2BsSsOfdmRspCapSsDemodulator  WmanIf2OfdmSsDeModType,
57      wmanIf2BsSsOfdmRspCapSsModulator     WmanIf2OfdmSsModType,
58      wmanIf2BsSsOfdmRspCapFocusedCtSupport WmanIf2OfdmFocusedCt,
59      wmanIf2BsSsOfdmRspCapTcSublayerSupport WmanIf2OfdmTcSublayer,
60      wmanIf2eBsSsOfdmRspCapPrivteMapSupport WmanIf2eOfdmPrivMap,
61      wmanIf2eBsSsOfdmRspCapUlPowerControl WmanIf2eOfdmUlPower,
62      wmanIf2eBsSsOfdmRspCapLoopPwrControlSw Unsigned32}
63
64
65

```

```

1  wmanIf2BsSsOfdmRspCapFftSizes OBJECT-TYPE
2      SYNTAX      WmanIf2OfdmFftSizes
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This field indicates the FFT sizes negotiated with the
7          SS. The usage is defined by WmanIf2OfdmFftSizes."
8      ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 1 }
9
10
11  wmanIf2BsSsOfdmRspCapSsDemodulator OBJECT-TYPE
12      SYNTAX      WmanIf2OfdmSsDeModType
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "This field indicates the different demodulator options
17          negotiated for SS for downlink. The usage is defined by
18          WmanIf2OfdmSsDeModType."
19      ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 2 }
20
21
22  wmanIf2BsSsOfdmRspCapSsModulator OBJECT-TYPE
23      SYNTAX      WmanIf2OfdmSsModType
24      MAX-ACCESS  read-only
25      STATUS      current
26      DESCRIPTION
27          "This field indicates the different modulator options
28          negotiated for SS for uplink. The usage is defined by
29          WmanIf2OfdmSsModType."
30      ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 3 }
31
32
33  wmanIf2BsSsOfdmRspCapFocusedCtSupport OBJECT-TYPE
34      SYNTAX      WmanIf2OfdmFocusedCt
35      MAX-ACCESS  read-only
36      STATUS      current
37      DESCRIPTION
38          "This field indicates whether the SS has negotiated the
39          support for Focused Contention. The usage is defined by
40          WmanIf2OfdmFocusedCt."
41      ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 4 }
42
43
44  wmanIf2BsSsOfdmRspCapTcSublayerSupport OBJECT-TYPE
45      SYNTAX      WmanIf2OfdmTcSublayer
46      MAX-ACCESS  read-only
47      STATUS      current
48      DESCRIPTION
49          "This field indicates whether the SS has negotiated
50          support for the TC sublayer. The usage is defined by
51          WmanIf2OfdmTcSublayer."
52      ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 5 }
53
54
55  wmanIf2eBsSsOfdmRspCapPrivteMapSupport OBJECT-TYPE
56      SYNTAX      WmanIf2eOfdmPrivMap
57      MAX-ACCESS  read-only
58      STATUS      current
59      DESCRIPTION
60
61
62
63
64
65

```

```

1         "This field indicates if the private map parameters
2         is supported."
3         ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 6 }
4
5
6 wmanIf2eBsSsOfdmRspCapUlPowerControl OBJECT-TYPE
7     SYNTAX      WmanIf2eOfdmUlPower
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "This field indicates the uplink power control options
12        supported by SS."
13        ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 7 }
14
15
16 wmanIf2eBsSsOfdmRspCapLoopPwrControlSw OBJECT-TYPE
17     SYNTAX      Unsigned32
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21        "This field indicates the minimum number of frames that
22        SS takes to switch from the open loop power control
23        scheme to the closed loop power control scheme or
24        vice versa."
25        ::= { wmanIf2BsSsOfdmRspCapabilitiesEntry 8 }
26
27
28
29
30 wmanIf2BsOfdmCapabilitiesTable OBJECT-TYPE
31     SYNTAX      SEQUENCE OF WmanIf2BsOfdmCapabilitiesEntry
32     MAX-ACCESS  not-accessible
33     STATUS      current
34     DESCRIPTION
35        "This table contains the basic capabilities, specific to
36        OFDM Phy, of the BS as implemented in BS hardware and
37        software. These capabilities along with the configuration
38        for them (wmanIf2BsOfdmCapabilitiesConfigTable) are used
39        for negotiation of basic capabilities with SS using
40        RNG-RSP, SBC-RSP and REG-RSP messages. The negotiated
41        capabilities are obtained by interSubclause of SS raw
42        reported capabilities, BS raw capabilities and BS
43        configured capabilities. The objects in the table have
44        read-only access. The table is maintained by BS."
45        ::= { wmanIf2BsOfdmPhy 8 }
46
47
48
49
50 wmanIf2BsOfdmCapabilitiesEntry OBJECT-TYPE
51     SYNTAX      WmanIf2BsOfdmCapabilitiesEntry
52     MAX-ACCESS  not-accessible
53     STATUS      current
54     DESCRIPTION
55        "This table provides one row for each BS sector and is
56        indexed by ifIndex."
57     INDEX { ifIndex }
58     ::= { wmanIf2BsOfdmCapabilitiesTable 1 }
59
60
61
62 WmanIf2BsOfdmCapabilitiesEntry ::= SEQUENCE {
63     wmanIf2BsOfdmCapFftSizes          WmanIf2OfdmFftSizes,
64     wmanIf2BsOfdmCapSsDemodulator     WmanIf2OfdmSsDeModType,
65

```

```

1      wmanIf2BsOfdmCapSsModulator      WmanIf2OfdmSsModType,
2      wmanIf2BsOfdmCapFocusedCtSupport  WmanIf2OfdmFocusedCt,
3      wmanIf2BsOfdmCapTcSublayerSupport WmanIf2OfdmTcSublayer,
4      wmanIf2eBsOfdmCapPrivteMapSupport WmanIf2eOfdmPrivMap,
5      wmanIf2eBsSsOfdmCapUlPowerControl WmanIf2eOfdmUlPower,
6      wmanIf2eBsSsOfdmCapLoopPwrControlSw Unsigned32}
7
8
9
10     wmanIf2BsOfdmCapFftSizes OBJECT-TYPE
11         SYNTAX      WmanIf2OfdmFftSizes
12         MAX-ACCESS  read-only
13         STATUS      current
14         DESCRIPTION
15             "This field indicates the FFT sizes supported by the BS.
16             The usage is defined by WmanIf2OfdmFftSizes."
17         ::= { wmanIf2BsOfdmCapabilitiesEntry 1 }
18
19
20     wmanIf2BsOfdmCapSsDemodulator OBJECT-TYPE
21         SYNTAX      WmanIf2OfdmSsDeModType
22         MAX-ACCESS  read-only
23         STATUS      current
24         DESCRIPTION
25             "This field indicates the different BS demodulator options
26             for uplink supported by the BS. The usage is defined by
27             WmanIf2OfdmSsDeModType."
28         ::= { wmanIf2BsOfdmCapabilitiesEntry 2 }
29
30
31
32     wmanIf2BsOfdmCapSsModulator OBJECT-TYPE
33         SYNTAX      WmanIf2OfdmSsModType
34         MAX-ACCESS  read-only
35         STATUS      current
36         DESCRIPTION
37             "This field indicates the different BS modulator options
38             for downlink supported by the BS. The usage is defined by
39             WmanIf2OfdmSsModType."
40         ::= { wmanIf2BsOfdmCapabilitiesEntry 3 }
41
42
43
44     wmanIf2BsOfdmCapFocusedCtSupport OBJECT-TYPE
45         SYNTAX      WmanIf2OfdmFocusedCt
46         MAX-ACCESS  read-only
47         STATUS      current
48         DESCRIPTION
49             "This field indicates the BS support for Focused
50             Contention. The usage is defined by
51             WmanIf2OfdmFocusedCt."
52         ::= { wmanIf2BsOfdmCapabilitiesEntry 4 }
53
54
55
56     wmanIf2BsOfdmCapTcSublayerSupport OBJECT-TYPE
57         SYNTAX      WmanIf2OfdmTcSublayer
58         MAX-ACCESS  read-only
59         STATUS      current
60         DESCRIPTION
61             "This field indicates the BS supports for TC sublayer. The
62             usage is defined by WmanIf2OfdmTcSublayer."
63         ::= { wmanIf2BsOfdmCapabilitiesEntry 5 }
64
65

```

```

1
2 wmanIf2eBsOfdmCapPrivteMapSupport OBJECT-TYPE
3     SYNTAX      WmanIf2eOfdmPrivMap
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "This field iindicates if the private map parameters
8         is supported."
9     ::= { wmanIf2BsOfdmCapabilitiesEntry 6 }
10
11
12
13 wmanIf2eBsSsOfdmCapUlPowerControl OBJECT-TYPE
14     SYNTAX      WmanIf2eOfdmUlPower
15     MAX-ACCESS  read-only
16     STATUS      current
17     DESCRIPTION
18         "This field indicates the uplink power control options
19         supported by SS."
20     ::= { wmanIf2BsOfdmCapabilitiesEntry 7 }
21
22
23
24 wmanIf2eBsSsOfdmCapLoopPwrControlSw OBJECT-TYPE
25     SYNTAX      Unsigned32
26     MAX-ACCESS  read-only
27     STATUS      current
28     DESCRIPTION
29         "This field indicates he minimum number of frames that
30         SS takes to switch from the open loop power control
31         scheme to the closed loop power control scheme or
32         vice versa."
33     ::= { wmanIf2BsOfdmCapabilitiesEntry 8 }
34
35
36
37 wmanIf2BsOfdmCapabilitiesConfigTable OBJECT-TYPE
38     SYNTAX      SEQUENCE OF WmanIf2BsOfdmCapabilitiesConfigEntry
39     MAX-ACCESS  not-accessible
40     STATUS      current
41     DESCRIPTION
42         "This table contains the configuration for basic
43         capabilities of BS, specific to OFDM Phy. The table is
44         intended to be used to restrict the Capabilities
45         implemented by BS, for example in order to comply with
46         local regulatory requirements. The BS should use the
47         configuration along with the implemented Capabilities
48         (wmanIf2BsOfdmPhyTable) for negotiation of basic
49         capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP
50         messages. The negotiated capabilities are obtained by
51         interSubclause of SS reported capabilities, BS raw
52         capabilities and BS configured capabilities. The objects
53         in the table have read-write access. The rows are created
54         by BS as a copy of wmanIf2BsBasicCapabilitiesTable
55         and can be modified by NMS."
56     ::= { wmanIf2BsOfdmPhy 9 }
57
58
59
60
61
62 wmanIf2BsOfdmCapabilitiesConfigEntry OBJECT-TYPE
63     SYNTAX      WmanIf2BsOfdmCapabilitiesConfigEntry
64     MAX-ACCESS  not-accessible
65

```



```

1      STATUS      current
2      DESCRIPTION
3          "This table provides one row for each BS sector and is
4          indexed by ifIndex."
5      INDEX { ifIndex }
6      ::= { wmanIf2BsOfdmCapabilitiesConfigTable 1 }
7
8
9
10     WmanIf2BsOfdmCapabilitiesConfigEntry ::= SEQUENCE {
11         wmanIf2BsOfdmCapCfgFftSizes          WmanIf2OfdmFftSizes,
12         wmanIf2BsOfdmCapCfgSsDemodulator      WmanIf2OfdmSsDeModType,
13         wmanIf2BsOfdmCapCfgSsModulator        WmanIf2OfdmSsModType,
14         wmanIf2BsOfdmCapCfgFocusedCtSupport    WmanIf2OfdmFocusedCt,
15         wmanIf2BsOfdmCapCfgTcSublayerSupport   WmanIf2OfdmTcSublayer,
16         wmanIf2eBsOfdmCapCfgPrivteMapSupport   WmanIf2eOfdmPrivMap,
17         wmanIf2eBsSsOfdmCapCfgUlPowerControl   WmanIf2eOfdmUlPower,
18         wmanIf2eBsSsOfdmCapCfgLoopPwrControlSw Unsigned32}
19
20
21     wmanIf2BsOfdmCapCfgFftSizes OBJECT-TYPE
22         SYNTAX      WmanIf2OfdmFftSizes
23         MAX-ACCESS   read-write
24         STATUS      current
25         DESCRIPTION
26             "This field indicates the FFT sizes support configured for
27             the BS. The usage is defined by WmanIf2OfdmFftSizes."
28         ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 1 }
29
30
31     wmanIf2BsOfdmCapCfgSsDemodulator OBJECT-TYPE
32         SYNTAX      WmanIf2OfdmSsDeModType
33         MAX-ACCESS   read-write
34         STATUS      current
35         DESCRIPTION
36             "This field indicates the different BS demodulator options
37             configured for uplink. The usage is defined by
38             WmanIf2OfdmSsDeModType."
39         ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 2 }
40
41
42     wmanIf2BsOfdmCapCfgSsModulator OBJECT-TYPE
43         SYNTAX      WmanIf2OfdmSsModType
44         MAX-ACCESS   read-write
45         STATUS      current
46         DESCRIPTION
47             "This field indicates the different BS modulator options
48             configured for downlink. The usage is defined by
49             WmanIf2OfdmSsModType."
50         ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 3 }
51
52
53     wmanIf2BsOfdmCapCfgFocusedCtSupport OBJECT-TYPE
54         SYNTAX      WmanIf2OfdmFocusedCt
55         MAX-ACCESS   read-write
56         STATUS      current
57         DESCRIPTION
58             "This field indicates the BS support configured for
59             Focused Contention. The usage is defined by
60             WmanIf2OfdmFocusedCt."
61
62
63
64
65

```

```

1      ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 4 }
2
3  wmanIf2BsOfdmCapCfgTcSublayerSupport OBJECT-TYPE
4      SYNTAX      WmanIf2OfdmTcSublayer
5      MAX-ACCESS  read-write
6      STATUS      current
7      DESCRIPTION
8          "This field indicates the BS support configured for TC
9          sublayer. The usage is defined by
10         WmanIf2OfdmTcSublayer."
11
12     ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 5 }
13
14
15  wmanIf2eBsOfdmCapCfgPrivteMapSupport OBJECT-TYPE
16      SYNTAX      WmanIf2eOfdmPrivMap
17      MAX-ACCESS  read-write
18      STATUS      current
19      DESCRIPTION
20          "This field iindicates if the private map parameters
21          is supported."
22
23     ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 6 }
24
25
26  wmanIf2eBsSsOfdmCapCfgUlPowerControl OBJECT-TYPE
27      SYNTAX      WmanIf2eOfdmUlPower
28      MAX-ACCESS  read-write
29      STATUS      current
30      DESCRIPTION
31          "This field indicates the uplink power control options
32          supported by SS."
33
34     ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 7 }
35
36
37  wmanIf2eBsSsOfdmCapCfgLoopPwrControlSw OBJECT-TYPE
38      SYNTAX      Unsigned32
39      MAX-ACCESS  read-write
40      STATUS      current
41      DESCRIPTION
42          "This field indicates he minimum number of frames that
43          SS takes to switch from the open loop power control
44          scheme to the closed loop power control scheme or
45          vice versa."
46
47     ::= { wmanIf2BsOfdmCapabilitiesConfigEntry 8 }
48
49
50  --
51  -- BS OFDMA PHY objects
52  --
53
54  wmanIf2BsOfdmaPhy OBJECT IDENTIFIER ::= { wmanIf2BsPhy 2 }
55
56
57  wmanIf2BsOfdmaUplinkChannelTable OBJECT-TYPE
58      SYNTAX      SEQUENCE OF WmanIf2BsOfdmaUplinkChannelEntry
59      MAX-ACCESS  not-accessible
60      STATUS      current
61      DESCRIPTION
62          "This table contains UCD channel attributes, defining the
63          transmission characteristics of uplink channels"
64
65  REFERENCE

```

"Table 349 and Table 353, in IEEE Std 802.16-2004"
 ::= { wmanIf2BsOfdmaPhy 1 }

wmanIf2BsOfdmaUplinkChannelEntry OBJECT-TYPE

SYNTAX WmanIf2BsOfdmaUplinkChannelEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table provides one row for each uplink channel of multi-sector BS, and is indexed by BS ifIndex. An entry in this table exists for each ifEntry of BS with an ifType of propBWAp2Mp."

INDEX { ifIndex }

::= { wmanIf2BsOfdmaUplinkChannelTable 1 }

WmanIf2BsOfdmaUplinkChannelEntry ::= SEQUENCE {

wmanIf2BsOfdmaCtBasedResvTimeout	INTEGER,
wmanIf2BsOfdmaBwReqOppSize	INTEGER,
wmanIf2BsOfdmaRangReqOppSize	INTEGER,
wmanIf2BsOfdmaUplinkCenterFreq	Unsigned32,
wmanIf2BsOfdmaInitRngCodes	INTEGER,
wmanIf2BsOfdmaPeriodicRngCodes	INTEGER,
wmanIf2BsOfdmaBWRngCodes	INTEGER,
wmanIf2BsOfdmaPerRngBackoffStart	INTEGER,
wmanIf2BsOfdmaPerRngBackoffEnd	INTEGER,
wmanIf2BsOfdmaStartOfRngCodes	INTEGER,
wmanIf2BsOfdmaPermutationBase	INTEGER,
wmanIf2BsOfdmaULAllocSubchBitmap	OCTET STRING,
wmanIf2BsOfdmaOptPermULAllocSubchBitmap	OCTET STRING,
wmanIf2BsOfdmaBandAMCAllocThreshold	INTEGER,
wmanIf2BsOfdmaBandAMCReleaseThreshold	INTEGER,
wmanIf2BsOfdmaBandAMCAllocTimer	INTEGER,
wmanIf2BsOfdmaBandAMCReleaseTimer	INTEGER,
wmanIf2BsOfdmaBandStatRepMAXPeriod	INTEGER,
wmanIf2BsOfdmaBandAMCRetryTimer	INTEGER,
wmanIf2BsOfdmaSafetyChAllocThreshold	INTEGER,
wmanIf2BsOfdmaSafetyChReleaseThreshold	INTEGER,
wmanIf2BsOfdmaSafetyChAllocTimer	INTEGER,
wmanIf2BsOfdmaSafetyChReleaseTimer	INTEGER,
wmanIf2BsOfdmaBinStatRepMAXPeriod	INTEGER,
wmanIf2BsOfdmaSafetyChARetryTimer	INTEGER,
wmanIf2BsOfdmaHARQAackDelayULBurst	INTEGER,
wmanIf2BsOfdmaCQICHBandAMCTranaDelay	INTEGER}

wmanIf2BsOfdmaCtBasedResvTimeout OBJECT-TYPE

SYNTAX INTEGER (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The number of UL-MAPs to receive before contention-based reservation is attempted again for the same connection."

REFERENCE

"Table 349, in IEEE Std 802.16-2004"

::= { wmanIf2BsOfdmaUplinkChannelEntry 1 }

```

1
2 wmanIf2BsOfdmaBwReqOppSize OBJECT-TYPE
3     SYNTAX      INTEGER (1..65535)
4     UNITS       "PS"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "Size (in units of PS) of PHY payload that SS may use to
9         format and transmit a bandwidth request message in a
10        contention request opportunity. The value includes all
11        PHY overhead as well as allowance for the MAC data the
12        message may hold."
13
14    REFERENCE
15        "Table 349, in IEEE Std 802.16-2004"
16    ::= { wmanIf2BsOfdmaUplinkChannelEntry 2 }
17
18 wmanIf2BsOfdmaRangReqOppSize OBJECT-TYPE
19     SYNTAX      INTEGER (1..65535)
20     UNITS       "PS"
21     MAX-ACCESS  read-write
22     STATUS      current
23     DESCRIPTION
24         "Size (in units of PS) of PHY payload that SS may use to
25         format and transmit a RNG-REQ message in a contention
26         request opportunity. The value includes all PHY overhead
27         as well as allowance for the MAC data the message may
28         hold and the maximum SS/BS roundtrip propagation delay."
29
30    REFERENCE
31        "Table 349, in IEEE Std 802.16-2004"
32    ::= { wmanIf2BsOfdmaUplinkChannelEntry 3 }
33
34 wmanIf2BsOfdmaUplinkCenterFreq OBJECT-TYPE
35     SYNTAX      Unsigned32
36     UNITS       "kHz"
37     MAX-ACCESS  read-write
38     STATUS      current
39     DESCRIPTION
40         " Uplink center frequency (kHz)"
41
42    REFERENCE
43        "Table 349, in IEEE Std 802.16-2004"
44    ::= { wmanIf2BsOfdmaUplinkChannelEntry 4 }
45
46 wmanIf2BsOfdmaInitRngCodes OBJECT-TYPE
47     SYNTAX      INTEGER (0..255)
48     MAX-ACCESS  read-write
49     STATUS      deprecated
50     DESCRIPTION
51         "Number of initial ranging CDMA codes. Possible values are
52         0..255. The total number of wmanIf2BsOfdmaInitRngCodes,
53         wmanIf2BsOfdmaPeriodicRngCodes and wmanIf2BsOfdmaBwReqCodes
54         shall be equal or less than 256."
55
56    REFERENCE
57        "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
58    DEFVAL      { 30 }
59
60
61
62
63
64
65

```

```

1      ::= { wmanIf2BsOfdmaUplinkChannelEntry 5 }
2
3
4  wmanIf2BsOfdmaPeriodicRngCodes OBJECT-TYPE
5      SYNTAX      INTEGER (0..255)
6      MAX-ACCESS  read-write
7      STATUS      deprecated
8      DESCRIPTION
9          "Number of periodic ranging CDMA codes. Possible values are
10         0..255. The total number of wmanIf2BsOfdmaInitRngCodes,
11         wmanIf2BsOfdmaPeriodicRngCodes and wmanIf2BsOfdmaBWReqCodes
12         shall be equal or less than 256."
13
14      REFERENCE
15          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
16
17      DEFVAL      { 30 }
18      ::= { wmanIf2BsOfdmaUplinkChannelEntry 6 }
19
20  wmanIf2BsOfdmaBWReqCodes OBJECT-TYPE
21      SYNTAX      INTEGER (0..255)
22      MAX-ACCESS  read-write
23      STATUS      deprecated
24      DESCRIPTION
25          "Number of bandwidth request codes. Possible values are
26         0..255. The total number of wmanIf2BsOfdmaInitRngCodes,
27         wmanIf2BsOfdmaPeriodicRngCodes and wmanIf2BsOfdmaBWReqCodes
28         shall be equal or less than 256."
29
30      REFERENCE
31          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
32
33      DEFVAL      { 30 }
34      ::= { wmanIf2BsOfdmaUplinkChannelEntry 7 }
35
36
37  wmanIf2BsOfdmaPerRngBackoffStart OBJECT-TYPE
38      SYNTAX      INTEGER (0..15)
39      MAX-ACCESS  read-write
40      STATUS      deprecated
41      DESCRIPTION
42          "Initial backoff window size for periodic ranging contention,
43         , expressed as a power of 2. Range: 0..15 (the highest order
44         bits shall be unused and set to 0)."
45
46      REFERENCE
47          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
48
49      DEFVAL      { 0 }
50      ::= { wmanIf2BsOfdmaUplinkChannelEntry 8 }
51
52
53  wmanIf2BsOfdmaPerRngBackoffEnd OBJECT-TYPE
54      SYNTAX      INTEGER (0 .. 15)
55      MAX-ACCESS  read-write
56      STATUS      deprecated
57      DESCRIPTION
58          "Final backoff window size for periodic ranging contention,
59         expressed as a power of 2. Range: 0..15 (the highest order
60         bits shall be unused and set to 0)."
61
62      REFERENCE
63          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
64
65      DEFVAL      { 15 }

```

```

1      ::= { wmanIf2BsOfdmaUplinkChannelEntry 9 }
2
3
4  wmanIf2eBsOfdmaStartOfRngCodes OBJECT-TYPE
5      SYNTAX      INTEGER (0..255)
6      MAX-ACCESS  read-write
7      STATUS      current
8      DESCRIPTION
9          "Indicates the starting number, S, of the group of codes
10         used for this uplink. All the ranging codes used on this
11         uplink will be between S and ((S+N+M+L) mod 256). Where,
12             N: the number of initial-ranging codes
13             M: the number of periodic-ranging codes
14             L: the number of bandwidth-request codes
15             O: the number of handover-ranging codes"
16
17      REFERENCE
18          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
19
20      DEFVAL      { 0 }
21      ::= { wmanIf2BsOfdmaUplinkChannelEntry 10 }
22
23
24  wmanIf2BsOfdmaPermutationBase OBJECT-TYPE
25      SYNTAX      INTEGER (0..255)
26      MAX-ACCESS  read-write
27      STATUS      current
28      DESCRIPTION
29          "Determines the UL_PermBase parameter for the subcarrier
30         permutation to be used on this uplink channel.
31             UL_PermBase = 7 LSBs of Permutation base."
32
33      REFERENCE
34          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
35
36      DEFVAL      { 0 }
37      ::= { wmanIf2BsOfdmaUplinkChannelEntry 11 }
38
39
40  wmanIf2BsOfdmaULAllocSubchBitmap OBJECT-TYPE
41      SYNTAX      OCTET STRING (SIZE (9))
42      MAX-ACCESS  read-write
43      STATUS      current
44      DESCRIPTION
45          "This is a bitmap describing the physical sub-channels
46         allocated to the segment in the UL, when using the uplink
47         PUSC permutation. The LSB of the first byte shall correspond
48         to subchannel 0. For any bit that is not set, the
49         corresponding subchannel shall not be used by the SS on
50         that segment"
51
52      REFERENCE
53          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
54
55      ::= { wmanIf2BsOfdmaUplinkChannelEntry 12 }
56
57
58  wmanIf2BsOfdmaOptPermULAllocSubchBitmap OBJECT-TYPE
59      SYNTAX      OCTET STRING (SIZE (13))
60      MAX-ACCESS  read-write
61      STATUS      current
62      DESCRIPTION
63          "This is a bitmap describing the sub-channels allocated to
64         the segment in the UL, when using the uplink optional PUSC
65

```

permutation (see 8.4.6.2.5 in IEEE Std 802.16-2004). The LSB of the first byte shall correspond to subchannel 0. For any bit that is not set, the corresponding subchannel shall not be used by the SS on that segment. When this TLV is not present, BS may allocate any subchannels to an SS."

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"

::= { wmanIf2BsOfdmaUplinkChannelEntry 13 }

wmanIf2BsOfdmaBandAMCAallocThreshold OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "dB"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Threshold of the maximum of the standard deviations of the individual bands CINR measurements over time to trigger mode transition from normal subchannel to Band AMC"

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"

::= { wmanIf2BsOfdmaUplinkChannelEntry 14 }

wmanIf2BsOfdmaBandAMCReleaseThreshold OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "dB"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Threshold of the maximum of the standard deviations of the individual bands CINR measurements over time to trigger mode transition from Band AMC to normal subchannel"

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"

::= { wmanIf2BsOfdmaUplinkChannelEntry 15 }

wmanIf2BsOfdmaBandAMCAallocTimer OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "Frame"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Minimum required number of frames to measure the average and standard deviation for the event of Band AMC triggering"

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"

::= { wmanIf2BsOfdmaUplinkChannelEntry 16 }

wmanIf2BsOfdmaBandAMCReleaseTimer OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "Frame"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Minimum required number of frames to measure the average

and standard deviation for the event triggering from Band
AMC to normal subchannel"

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
::= { wmanIf2BsOfdmaUplinkChannelEntry 17 }

wmanIf2BsOfdmaBandStatRepMAXPeriod OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "Frame"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Maximum period between refreshing the Band CINR
measurement by the unsolicited REP-RSP"

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
::= { wmanIf2BsOfdmaUplinkChannelEntry 18 }

wmanIf2BsOfdmaBandAMCRetryTimer OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "Frame"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Backoff timer between consecutive mode transitions from
normal subchannel to Band AMC when the previous request
is failed"

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
::= { wmanIf2BsOfdmaUplinkChannelEntry 19 }

wmanIf2BsOfdmaSafetyChAllocThreshold OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "dB"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"This object defines the OFDMA safety channel allocation
threshold."

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
::= { wmanIf2BsOfdmaUplinkChannelEntry 20 }

wmanIf2BsOfdmaSafetyChReleaseThreshold OBJECT-TYPE

SYNTAX INTEGER (0 .. 255)

UNITS "dB"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"This object defines the OFDMA safety channel release
threshold."

REFERENCE

"Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
::= { wmanIf2BsOfdmaUplinkChannelEntry 21 }


```

1
2 wmanIf2BsOfdmaSafetyChAllocTimer OBJECT-TYPE
3     SYNTAX      INTEGER (0 .. 255)
4     UNITS       "Frame"
5     MAX-ACCESS  read-write
6     STATUS      deprecated
7     DESCRIPTION
8         "This object defines the OFDMA safety channel allocation
9         timer."
10
11     REFERENCE
12         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
13     ::= { wmanIf2BsOfdmaUplinkChannelEntry 22 }
14
15
16 wmanIf2BsOfdmaSafetyChReleaseTimer OBJECT-TYPE
17     SYNTAX      INTEGER (0 .. 255)
18     UNITS       "Frame"
19     MAX-ACCESS  read-write
20     STATUS      deprecated
21     DESCRIPTION
22         "This object defines the OFDMA safety channel release
23         timer."
24
25     REFERENCE
26         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
27     ::= { wmanIf2BsOfdmaUplinkChannelEntry 23 }
28
29
30
31 wmanIf2BsOfdmaBinStatRepMAXPeriod OBJECT-TYPE
32     SYNTAX      INTEGER (0 .. 255)
33     UNITS       "Frame"
34     MAX-ACCESS  read-write
35     STATUS      deprecated
36     DESCRIPTION
37         "This object defines the OFDMA bin status reporting
38         maximum period."
39
40     REFERENCE
41         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
42     ::= { wmanIf2BsOfdmaUplinkChannelEntry 24 }
43
44
45
46 wmanIf2BsOfdmaSafetyChaRetryTimer OBJECT-TYPE
47     SYNTAX      INTEGER (0 .. 255)
48     UNITS       "Frame"
49     MAX-ACCESS  read-write
50     STATUS      deprecated
51     DESCRIPTION
52         "This object defines the OFDMA safety channel retry
53         timer."
54
55     REFERENCE
56         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
57     ::= { wmanIf2BsOfdmaUplinkChannelEntry 25 }
58
59
60 wmanIf2BsOfdmaHARQAackDelayULBurst OBJECT-TYPE
61     SYNTAX      INTEGER { oneframeoffset(1),
62                        twoframesoffset(2),
63                        threeframesoffset(3) }
64     MAX-ACCESS  read-write
65

```

```

1      STATUS      deprecated
2      DESCRIPTION
3          "This object defines the OFDMA H-ARQ ACK delay for UL burst.
4              1 = one frame offset
5              2 = two frames offset
6              3 = three frames offset"
7
8      REFERENCE
9          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
10         ::= { wmanIf2BsOfdmaUplinkChannelEntry 26 }
11
12
13 wmanIf2BsOfdmaCQICHBandAMCTranaDelay OBJECT-TYPE
14     SYNTAX      INTEGER (0 .. 255)
15     UNITS       "Frame"
16     MAX-ACCESS  read-write
17     STATUS      deprecated
18     DESCRIPTION
19         "This object defines the OFDMA CQICH band AMC transition
20         delay."
21     REFERENCE
22         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
23         ::= { wmanIf2BsOfdmaUplinkChannelEntry 27 }
24
25
26
27 wmanIf2BsOfdmaDownlinkChannelTable OBJECT-TYPE
28     SYNTAX      SEQUENCE OF WmanIf2BsOfdmaDownlinkChannelEntry
29     MAX-ACCESS  not-accessible
30     STATUS      current
31     DESCRIPTION
32         "This table contains DCD channel attributes, defining the
33         transmission characteristics of downlink channels"
34     REFERENCE
35         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
36         ::= { wmanIf2BsOfdmaPhy 2 }
37
38
39
40
41 wmanIf2BsOfdmaDownlinkChannelEntry OBJECT-TYPE
42     SYNTAX      WmanIf2BsOfdmaDownlinkChannelEntry
43     MAX-ACCESS  not-accessible
44     STATUS      current
45     DESCRIPTION
46         "This table provides one row for each downlink channel of
47         multi-sector BS, and is indexed by BS ifIndex. An entry in
48         this table exists for each ifEntry of BS with an ifType of
49         propBWA2Mp."
50     INDEX       { ifIndex }
51     ::= { wmanIf2BsOfdmaDownlinkChannelTable 1 }
52
53
54
55 WmanIf2BsOfdmaDownlinkChannelEntry ::= SEQUENCE {
56     wmanIf2BsOfdmaBsEIRP          INTEGER,
57     wmanIf2BsOfdmaChannelNumber   WmanIf2ChannelNumber,
58     wmanIf2BsOfdmaATTG            INTEGER,
59     wmanIf2BsOfdmaARTG            INTEGER,
60     wmanIf2BsOfdmaInitRngMaxRSS   INTEGER,
61     wmanIf2BsOfdmaDownlinkCenterFreq Unsigned32,
62     wmanIf2BsOfdmaBsId            WmanIf2BsIdType,
63     wmanIf2BsOfdmaMacVersion      WmanIf2MacVersion,
64

```

```

1      wmanIf2BsOfdmaFrameDurationCode      INTEGER,
2      wmanIf2BsOfdmaSizeCqichIdField      INTEGER,
3      wmanIf2BsOfdmaHARQAackDelayBurst    INTEGER}
4
5
6  wmanIf2BsOfdmaBsEIRP OBJECT-TYPE
7      SYNTAX      INTEGER (-32768..32767)
8      UNITS      "dBm"
9      MAX-ACCESS  read-write
10     STATUS      current
11     DESCRIPTION
12         "The EIRP is the equivalent isotropic radiated power of
13         the base station, which is computed for a simple
14         single-antenna transmitter."
15     REFERENCE
16         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
17     ::= { wmanIf2BsOfdmaDownlinkChannelEntry 1 }
18
19
20
21  wmanIf2BsOfdmaChannelNumber OBJECT-TYPE
22     SYNTAX      WmanIf2ChannelNumber
23     MAX-ACCESS  read-write
24     STATUS      current
25     DESCRIPTION
26         "Downlink channel number as defined in 8.5. Used for
27         license-exempt operation only."
28     REFERENCE
29         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
30     ::= { wmanIf2BsOfdmaDownlinkChannelEntry 2 }
31
32
33
34  wmanIf2BsOfdmaTTG OBJECT-TYPE
35     SYNTAX      INTEGER (0..255)
36     MAX-ACCESS  read-write
37     STATUS      current
38     DESCRIPTION
39         "Transmit / Receive Transition Gap."
40     REFERENCE
41         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
42     ::= { wmanIf2BsOfdmaDownlinkChannelEntry 3 }
43
44
45
46  wmanIf2BsOfdmaRTG OBJECT-TYPE
47     SYNTAX      INTEGER (0..255)
48     MAX-ACCESS  read-write
49     STATUS      current
50     DESCRIPTION
51         "Receive / Transmit Transition Gap."
52     REFERENCE
53         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
54     ::= { wmanIf2BsOfdmaDownlinkChannelEntry 4 }
55
56
57
58  wmanIf2BsOfdmaInitRngMaxRSS OBJECT-TYPE
59     SYNTAX      INTEGER (-32768..32767)
60     UNITS      "dBm"
61     MAX-ACCESS  read-write
62     STATUS      current
63     DESCRIPTION
64
65

```

"Initial Ranging Max. equivalent isotropic received power
at BS Signed in units of 1 dBm."

REFERENCE

"Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
::= { wmanIf2BsOfdmaDownlinkChannelEntry 5 }

wmanIf2BsOfdmaDownlinkCenterFreq OBJECT-TYPE

SYNTAX Unsigned32

UNITS "kHz"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Downlink center frequency (kHz)."

REFERENCE

"Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
::= { wmanIf2BsOfdmaDownlinkChannelEntry 6 }

wmanIf2BsOfdmaBsId OBJECT-TYPE

SYNTAX WmanIf2BsIdType

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Base station ID."

REFERENCE

"Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
::= { wmanIf2BsOfdmaDownlinkChannelEntry 7 }

wmanIf2BsOfdmaMacVersion OBJECT-TYPE

SYNTAX WmanIf2MacVersion

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This parameter specifies the version of 802.16 to which
the message originator conforms."

REFERENCE

"Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
::= { wmanIf2BsOfdmaDownlinkChannelEntry 8 }

wmanIf2BsOfdmaFrameDurationCode OBJECT-TYPE

SYNTAX INTEGER { aASGap(0),
duration2ms(1),
duration2dot5ms(2),
duration4ms(3),
duration5ms(4),
duration8ms(5),
duration10ms(6),
duration12dot5ms(7),
duration20ms(8) }

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The duration of the frame. The frame duration code values
are specified in Table 274."

REFERENCE

```

1         "Table 273, in IEEE Std 802.16-2004"
2         ::= { wmanIf2BsOfdmaDownlinkChannelEntry 9 }
3
4
5 wmanIf2BsOfdmaSizeCqichIdField OBJECT-TYPE
6     SYNTAX      INTEGER {threebits(1),
7                     fourbits(2),
8                     fivebits(3),
9                     sixbits(4),
10                    sevenbits(5),
11                    eightbits(6),
12                    ninebits(7)}
13
14     MAX-ACCESS   read-write
15     STATUS       current
16     DESCRIPTION
17         "This object defines the size of CQICH ID field.
18         0 = Reserved
19         1 = 3 bits
20         2 = 4 bits
21         3 = 5 bits
22         4 = 6 bits
23         5 = 7 bits
24         6 = 8 bits
25         7 = 9 bits
26         8...255 = Reserved"
27
28     REFERENCE
29         "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
30     ::= { wmanIf2BsOfdmaDownlinkChannelEntry 10 }
31
32
33
34
35 wmanIf2BsOfdmaHARQAackDelayBurst OBJECT-TYPE
36     SYNTAX      INTEGER {oneframeoffset(1),
37                     twoframesoffset(2),
38                     threeframesoffset(3)}
39
40     MAX-ACCESS   read-write
41     STATUS       current
42     DESCRIPTION
43         "This object defines the OFDMA H-ARQ ACK delay for DL burst.
44         1 = one frame offset
45         2 = two frames offset
46         3 = three frames offset"
47
48     REFERENCE
49         "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
50     ::= { wmanIf2BsOfdmaDownlinkChannelEntry 11 }
51
52
53
54 wmanIf2BsOfdmaUcdBurstProfileTable OBJECT-TYPE
55     SYNTAX      SEQUENCE OF WmanIf2BsOfdmaUcdBurstProfileEntry
56     MAX-ACCESS   not-accessible
57     STATUS       current
58     DESCRIPTION
59         "This table contains UCD burst profiles for each uplink
60         channel"
61
62     REFERENCE
63         "Table 356, in IEEE Std 802.16-2004"
64     ::= { wmanIf2BsOfdmaPhy 3 }
65

```

```

1  wmanIf2BsOfdmaUcdBurstProfileEntry OBJECT-TYPE
2      SYNTAX      WmanIf2BsOfdmaUcdBurstProfileEntry
3      MAX-ACCESS  not-accessible
4      STATUS      current
5      DESCRIPTION
6          "This table provides one row for each UCD burst profile.
7          This table is double indexed. The primary index is an
8          ifIndex with an ifType of propBWA2Mp. The secondary index
9          is wmanIf2BsOfdmaUiucIndex."
10     INDEX      { ifIndex, wmanIf2BsOfdmaUiucIndex }
11     ::= { wmanIf2BsOfdmaUcdBurstProfileTable 1 }
12
13  WmanIf2BsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
14      wmanIf2BsOfdmaUiucIndex      INTEGER,
15      wmanIf2BsOfdmaUcdFecCodeType WmanIf2OfdmaFecCodeType,
16      wmanIf2BsOfdmaRangingDataRatio INTEGER,
17      wmanIf2BsOfdmaNorCOverNOVERRIDE OCTET STRING,
18      wmanIf2BsOfdmaUcdBurstProfileRowStatus RowStatus}
19
20  wmanIf2BsOfdmaUiucIndex OBJECT-TYPE
21      SYNTAX      INTEGER (1 .. 10)
22      MAX-ACCESS  not-accessible
23      STATUS      current
24      DESCRIPTION
25          "The Uplink Interval Usage Code indicates the uplink burst
26          profile in the UCD message, and is used along with ifIndex
27          to identify an entry in the
28          wmanIf2BsOfdmaUcdBurstProfileTable."
29      REFERENCE
30          "Subclause 8.4.5.4.1, in IEEE Std 802.16-2004"
31      ::= { wmanIf2BsOfdmaUcdBurstProfileEntry 1 }
32
33  wmanIf2BsOfdmaUcdFecCodeType OBJECT-TYPE
34      SYNTAX      WmanIf2OfdmaFecCodeType
35      MAX-ACCESS  read-create
36      STATUS      current
37      DESCRIPTION
38          "Uplink FEC code type and modulation type"
39      REFERENCE
40          "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
41      ::= { wmanIf2BsOfdmaUcdBurstProfileEntry 2 }
42
43  wmanIf2BsOfdmaRangingDataRatio OBJECT-TYPE
44      SYNTAX      INTEGER (0 .. 255)
45      MAX-ACCESS  read-create
46      STATUS      current
47      DESCRIPTION
48          "Reducing factor in units of 1 dB, between the power used
49          for this burst and power should be used for CDMA Ranging."
50      REFERENCE
51          "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
52      ::= { wmanIf2BsOfdmaUcdBurstProfileEntry 3 }
53
54  wmanIf2BsOfdmaNorCOverNOVERRIDE OBJECT-TYPE

```

```

1      SYNTAX OCTET STRING (SIZE (5))
2      MAX-ACCESS read-create
3      STATUS current
4      DESCRIPTION
5          "This is a list of numbers, where each number is encoded by
6           one nibble, and interpreted as a signed integer. The nibbles
7           correspond in order to the list define by Table 334 in IEEE
8           Std 802.16-2004 starting from the second line, such that
9           the LS nibble of the first byte corresponds to the second
10          line in the table. The number encoded by each nibble
11          represents the difference in normalized C/N relative to the
12          previous line in the table"
13      REFERENCE
14          "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
15      ::= { wmanIf2BsOfdmaUcdBurstProfileEntry 4 }
16
17  wmanIf2BsOfdmaUcdBurstProfileRowStatus OBJECT-TYPE
18      SYNTAX      RowStatus
19      MAX-ACCESS  read-create
20      STATUS      current
21      DESCRIPTION
22          "This object is used to create a new row or modify or delete
23          an existing row in this table. If the implementator of this
24          MIB has choosen not to implement 'dynamic assignment' of
25          profiles, this object is not useful and should return
26          noSuchName upon SNMP request."
27      ::= { wmanIf2BsOfdmaUcdBurstProfileEntry 5 }
28
29  wmanIf2BsOfdmaDcdBurstProfileTable OBJECT-TYPE
30      SYNTAX      SEQUENCE OF WmanIf2BsOfdmaDcdBurstProfileEntry
31      MAX-ACCESS  not-accessible
32      STATUS      current
33      DESCRIPTION
34          "This table provides one row for each DCD burst profile.
35          This table is double indexed. The primary index is an
36          ifIndex with an ifType of propBWA2Mp. The secondary index
37          is wmanIf2BsOfdmaDiucIndex."
38      ::= { wmanIf2BsOfdmaPhy 4 }
39
40  wmanIf2BsOfdmaDcdBurstProfileEntry OBJECT-TYPE
41      SYNTAX      WmanIf2BsOfdmaDcdBurstProfileEntry
42      MAX-ACCESS  not-accessible
43      STATUS      current
44      DESCRIPTION
45          "This table provides one row for each DCD burst profile.
46          This table is double indexed. The primary index is an
47          ifIndex with an ifType of propBWA2Mp. The secondary index
48          is wmanIf2BsOfdmaDiucIndex."
49      INDEX      { ifIndex, wmanIf2BsOfdmaDiucIndex }
50      ::= { wmanIf2BsOfdmaDcdBurstProfileTable 1 }
51
52  WmanIf2BsOfdmaDcdBurstProfileEntry ::= SEQUENCE {
53      wmanIf2BsOfdmaDiucIndex          INTEGER,
54      wmanIf2BsOfdmaDownlinkFrequency Unsigned32,

```

```

1      wmanIf2BsOfdmaDcdFecCodeType          WmanIf2OfdmaFecCodeType,
2      wmanIf2BsOfdmaDiucMandatoryExitThresh  INTEGER,
3      wmanIf2BsOfdmaDiucMinEntryThresh       INTEGER,
4      wmanIf2BsOfdmaDcdBurstProfileRowStatus RowStatus}
5
6
7  wmanIf2BsOfdmaDiucIndex OBJECT-TYPE
8      SYNTAX      INTEGER (0 .. 12)
9      MAX-ACCESS  not-accessible
10     STATUS      current
11     DESCRIPTION
12         "The Downlink Interval Usage Code indicates the downlink
13         burst profile in the DCD message, and is used along with
14         ifIndex to identify an entry in the
15         wmanIf2BsOfdmaDcdBurstProfileTable."
16     REFERENCE
17         "Subclause 8.4.5.3.1, in IEEE Std 802.16-2004"
18     ::= { wmanIf2BsOfdmaDcdBurstProfileEntry 1 }
19
20
21  wmanIf2BsOfdmaDownlinkFrequency OBJECT-TYPE
22     SYNTAX      Unsigned32
23     UNITS       "kHz"
24     MAX-ACCESS  read-create
25     STATUS      current
26     DESCRIPTION
27         "Downlink Frequency (kHz)."

```



```

1      MAX-ACCESS    read-create
2      STATUS        current
3      DESCRIPTION
4          "DIUC minimum entry threshold: 0 - 63.75 dB The minimum
5          CINR required to start using this DIUC when changing from
6          a more robust DIUC is required, in 0.25 dB units."
7      REFERENCE
8          "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
9      ::= { wmanIf2BsOfdmaDcdBurstProfileEntry 5 }
10
11
12
13  wmanIf2BsOfdmaDcdBurstProfileRowStatus OBJECT-TYPE
14      SYNTAX          RowStatus
15      MAX-ACCESS      read-create
16      STATUS          current
17      DESCRIPTION
18          "This object is used to create a new row or modify or delete
19          an existing row in this table. If the implementator of this
20          MIB has choosen not to implement 'dynamic assignment' of
21          profiles, this object is not useful and should return
22          noSuchName upon SNMP request."
23      ::= { wmanIf2BsOfdmaDcdBurstProfileEntry 6 }
24
25
26
27  wmanIf2BsSsOfdmaReqCapabilitiesTable OBJECT-TYPE
28      SYNTAX          SEQUENCE OF WmanIf2BsSsOfdmaReqCapabilitiesEntry
29      MAX-ACCESS      not-accessible
30      STATUS          current
31      DESCRIPTION
32          "This table contains the basic capability information,
33          specific to OFDMA Phy, of MSs that have been reported by
34          MSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.
35          Entries in this table should be created when an MS
36          registers with a BS."
37      ::= { wmanIf2BsOfdmaPhy 5 }
38
39
40
41
42  wmanIf2BsSsOfdmaReqCapabilitiesEntry OBJECT-TYPE
43      SYNTAX          WmanIf2BsSsOfdmaReqCapabilitiesEntry
44      MAX-ACCESS      not-accessible
45      STATUS          current
46      DESCRIPTION
47          "This table provides one row for each MS that has been
48          registered in the BS. This table augments the table
49          wmanIf2BsRegisteredSsTable."
50      AUGMENTS { wmanIf2BsRegisteredSsEntry }
51      ::= { wmanIf2BsSsOfdmaReqCapabilitiesTable 1 }
52
53
54
55  WmanIf2BsSsOfdmaReqCapabilitiesEntry ::= SEQUENCE {
56      wmanIf2BsSsOfdmaReqCapFftSizes          WmanIf2OfdmaFftSizes,
57      wmanIf2BsSsOfdmaReqCapDemodulator        WmanIf2OfdmaMsDeModType,
58      wmanIf2BsSsOfdmaReqCapModulator          WmanIf2OfdmaMsModType,
59      wmanIf2BsSsOfdmaReqCapNoHarqChannel      Unsigned32,
60      wmanIf2BsSsOfdmaReqCapPermutation        WmanIf2OfdmaPermutation,
61      wmanIf2BsSsOfdmaReqCapMobilityFeature    WmanIf2eOfdmaMobility,
62      wmanIf2BsSsOfdmaReqCapMaxMacLevelDlFm    WmanIf2eMaxMacLevel,
63      wmanIf2BsSsOfdmaReqCapMaxMacLevelUlFm    WmanIf2eMaxMacLevel,
64
65

```

```

1      wmanIf2eBsSsOfdmaReqCapDemMimo      WmanIf2eOfdmaDemMimo,
2      wmanIf2eBsSsOfdmaReqCapMimoCapability WmanIf2eOfdmaMimoCap,
3      wmanIf2eBsSsOfdmaReqCapUlMimo        WmanIf2eOfdmaUlMimo,
4      wmanIf2eBsSsOfdmaReqCapPrivateMap    WmanIf2eOfdmaPrivMap,
5      wmanIf2eBsSsOfdmaReqCapAasCapability WmanIf2eOfdmaAasCap,
6      wmanIf2eBsSsOfdmaReqCapCinrMeasurement WmanIf2eOfdmaCinrCap,
7      wmanIf2eBsSsOfdmaReqCapUlPowerControl WmanIf2eOfdmaUlPower,
8      wmanIf2eBsSsOfdmaReqCapMapCapability WmanIf2eOfdmaMapCap,
9      wmanIf2eBsSsOfdmaReqCapUlControlChannel WmanIf2eOfdmaUlCntlCh,
10     wmanIf2eBsSsOfdmaReqCapCistCapability WmanIf2eOfdmaMsCistCap,
11     wmanIf2eBsSsOfdmaReqCapMaxHarqBurst   WmanIf2eOfdmaMaxHarq,
12     wmanIf2eBsSsOfdmaReqCapModMimo        WmanIf2eOfdmaModMimo,
13     wmanIf2eBsSsOfdmaReqCapSdmaPilot      WmanIf2eSdmaPilotCap,
14     wmanIf2eBsSsOfdmaReqCapMultipleBurst  WmanIf2eMultiBurst,
15     wmanIf2eBsSsOfdmaReqCapIncrHarqBuffer WmanIf2eIncrHarqBuf,
16     wmanIf2eBsSsOfdmaReqCapChaseHarqBuffer WmanIf2eChaseHarqBuf}
17
18
19
20
21
22     wmanIf2BsSsOfdmaReqCapFftSizes OBJECT-TYPE
23         SYNTAX      WmanIf2OfdmaFftSizes
24         MAX-ACCESS  read-only
25         STATUS      current
26         DESCRIPTION
27             "This field indicates the FFT sizes supported by MS."
28             ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 1 }
29
30
31
32     wmanIf2BsSsOfdmaReqCapDemodulator OBJECT-TYPE
33         SYNTAX      WmanIf2OfdmaMsDeModType
34         MAX-ACCESS  read-only
35         STATUS      current
36         DESCRIPTION
37             "This field indicates the different demodulator options
38             supported by MS for downlink."
39             ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 2 }
40
41
42
43     wmanIf2BsSsOfdmaReqCapModulator OBJECT-TYPE
44         SYNTAX      WmanIf2OfdmaMsModType
45         MAX-ACCESS  read-only
46         STATUS      current
47         DESCRIPTION
48             "This field indicates the different modulator options
49             supported by MS for uplink."
50             ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 3 }
51
52
53
54     wmanIf2eBsSsOfdmaReqCapNoHarqChannel OBJECT-TYPE
55         SYNTAX      Unsigned32
56         MAX-ACCESS  read-only
57         STATUS      current
58         DESCRIPTION
59             "This field specifies the number of uplink H-ARQ
60             channels (n) the SS supports, where n = 1..16.
61             The value of this object should be 0..15."
62             ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 4 }
63
64
65
66     wmanIf2BsSsOfdmaReqCapPermutation OBJECT-TYPE

```

```

1      SYNTAX      WmanIf2OfdmaPermutation
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "This field indicates the OFDMA MS Permutation support."
6      ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 5 }
7
8
9
10     wmanIf2eBsSsOfdmaReqCapMobilityFeature OBJECT-TYPE
11         SYNTAX      WmanIf2eOfdmaMobility
12         MAX-ACCESS  read-only
13         STATUS      current
14         DESCRIPTION
15             "The field indicates whether or not the MS supports
16             mobility hand-over, Sleepmode, and Idle-mode."
17         ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 6 }
18
19
20     wmanIf2eBsSsOfdmaReqCapMaxMacLevelDlFm OBJECT-TYPE
21         SYNTAX      WmanIf2eMaxMacLevel
22         MAX-ACCESS  read-only
23         STATUS      current
24         DESCRIPTION
25             "Maximum amount of MAC level data the MS is capable of
26             processing per DL frame."
27         REFERENCE
28             "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
29         DEFVAL      { 0 }
30         ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 7 }
31
32
33
34     wmanIf2eBsSsOfdmaReqCapMaxMacLevelUlFm OBJECT-TYPE
35         SYNTAX      WmanIf2eMaxMacLevel
36         MAX-ACCESS  read-only
37         STATUS      current
38         DESCRIPTION
39             "Maximum amount of MAC level data the MS is capable of
40             processing per UL frame."
41         REFERENCE
42             "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
43         DEFVAL      { 0 }
44         ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 8 }
45
46
47
48     wmanIf2eBsSsOfdmaReqCapDemMimo OBJECT-TYPE
49         SYNTAX      WmanIf2eOfdmaDemMimo
50         MAX-ACCESS  read-only
51         STATUS      current
52         DESCRIPTION
53             "This field indicates the different MIMO options supported
54             by a WirelessMAN-OFDMA PHY SS in the downlink."
55         ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 9 }
56
57
58
59     wmanIf2eBsSsOfdmaReqCapMimoCapability OBJECT-TYPE
60         SYNTAX      WmanIf2eOfdmaMimoCap
61         MAX-ACCESS  read-only
62         STATUS      current
63         DESCRIPTION
64
65

```

```

1         "This field indicates the MIMO capability of OFDMA MS
2         demodulator."
3         ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 10 }
4
5
6 wmanIf2eBsSsOfdmaReqCapUlMimo OBJECT-TYPE
7     SYNTAX      WmanIf2eOfdmaUlMimo
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "This field indicates different MIMO options supported
12        by a OFDMA PHY SS in the uplink"
13    REFERENCE
14        "Subclause 11.8.3.7.6 in IEEE 802.16e"
15    ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 11 }
16
17
18
19 wmanIf2eBsSsOfdmaReqCapPrivateMap OBJECT-TYPE
20     SYNTAX      WmanIf2eOfdmaPrivMap
21     MAX-ACCESS  read-only
22     STATUS      current
23     DESCRIPTION
24        "This field indicates AAS private map parameters
25        supported by a OFDMA SS"
26    REFERENCE
27        "Subclause 11.8.3.7.7 in IEEE 802.16e"
28    ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 12 }
29
30
31
32 wmanIf2eBsSsOfdmaReqCapAasCapability OBJECT-TYPE
33     SYNTAX      WmanIf2eOfdmaAasCap
34     MAX-ACCESS  read-only
35     STATUS      current
36     DESCRIPTION
37        "This field indicates different AAS options
38        supported by a OFDMA PHY SS in the downlink"
39    REFERENCE
40        "Subclause 11.8.3.7.8 in IEEE 802.16e"
41    ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 13 }
42
43
44
45 wmanIf2eBsSsOfdmaReqCapCinrMesurement OBJECT-TYPE
46     SYNTAX      WmanIf2eOfdmaCinrCap
47     MAX-ACCESS  read-only
48     STATUS      current
49     DESCRIPTION
50        "This field indicates the CINR measurement capability
51        supported by a OFDMA PHY SS in the downlink."
52    REFERENCE
53        "Subclause 11.8.3.7.9 in IEEE 802.16e"
54    ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 14 }
55
56
57
58
59 wmanIf2eBsSsOfdmaReqCapUlPowerControl OBJECT-TYPE
60     SYNTAX      WmanIf2eOfdmaUlPower
61     MAX-ACCESS  read-only
62     STATUS      current
63     DESCRIPTION
64        "This field indicates the power control options
65

```

```

1         supported by a OFDMA PHY SS for uplink transmission."
2     REFERENCE
3         "Subclause 11.8.3.7.11 in IEEE 802.16e"
4         ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 15 }
5
6
7     wmanIf2eBsSsOfdmaReqCapMapCapability OBJECT-TYPE
8         SYNTAX      WmanIf2eOfdmaMapCap
9         MAX-ACCESS   read-only
10        STATUS      current
11        DESCRIPTION
12            "This field indicates the different MAP options supported
13             by a OFDMA PHY SS"
14        REFERENCE
15            "Subclause 11.8.3.7.11 in IEEE 802.16e"
16            ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 16 }
17
18
19
20    wmanIf2eBsSsOfdmaReqCapUlControlChannel OBJECT-TYPE
21        SYNTAX      WmanIf2eOfdmaUlCntlCh
22        MAX-ACCESS   read-only
23        STATUS      current
24        DESCRIPTION
25            "This field indicates the different uplink control channels
26             supported by a OFDMA PHY SS."
27        REFERENCE
28            "Subclause 11.8.3.7.13 in IEEE 802.16e"
29            ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 17 }
30
31
32
33    wmanIf2eBsSsOfdmaReqCapCistCapability OBJECT-TYPE
34        SYNTAX      WmanIf2eOfdmaMsCistCap
35        MAX-ACCESS   read-only
36        STATUS      current
37        DESCRIPTION
38            "This field indicates the MS capability of supporting CSIT
39             (uplink sounding)."
40        REFERENCE
41            "Subclause 11.8.3.7.14 in IEEE 802.16e"
42            ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 18 }
43
44
45
46    wmanIf2eBsSsOfdmaReqCapMaxHarqBurst OBJECT-TYPE
47        SYNTAX      WmanIf2eOfdmaMaxHarq
48        MAX-ACCESS   read-only
49        STATUS      current
50        DESCRIPTION
51            "This field indicates the maximum number of UL/DL HARQ
52             burst allocations for the SS in a single UL/DL subframe."
53        REFERENCE
54            "Subclause 11.8.3.7.15 in IEEE 802.16e"
55            ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 19 }
56
57
58
59    wmanIf2eBsSsOfdmaReqCapModMimo OBJECT-TYPE
60        SYNTAX      WmanIf2eOfdmaModMimo
61        MAX-ACCESS   read-only
62        STATUS      current
63        DESCRIPTION
64
65

```

"This field indicates the MIMO capability of OFDMA SS modulator."

REFERENCE

"Subclause 11.8.3.7.16 in IEEE 802.16e"
 ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 20 }

wmanIf2eBsSsOfdmaReqCapSdmaPilot OBJECT-TYPE

SYNTAX WmanIf2eSdmaPilotCap

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This field indicates the SDMA pilot pattern support for AMC zone."

REFERENCE

"Subclause 11.8.3.7.17 in IEEE 802.16e"
 ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 21 }

wmanIf2eBsSsOfdmaReqCapMultipleBurst OBJECT-TYPE

SYNTAX WmanIf2eMultiBurst

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This field indicates whether multiple FEC types are supported in DL/UL burst profiles."

REFERENCE

"Subclause 11.8.3.7.18 in IEEE 802.16e"
 ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 22 }

wmanIf2eBsSsOfdmaReqCapIncrHarqBuffer OBJECT-TYPE

SYNTAX WmanIf2eIncrHarqBuf

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This field indicates the maximal number of data bits the SS is able to use for buffering for NEP/NSCH based incremental redundancy CTC in downlink and uplink transmissions."

REFERENCE

"Subclause 11.8.3.7.19 in IEEE 802.16e"
 ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 23 }

wmanIf2eBsSsOfdmaReqCapChaseHarqBuffer OBJECT-TYPE

SYNTAX WmanIf2eChaseHarqBuf

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This field indicates the maximal number of data bits the SS is able to use for buffering for DIUC/duration based HARQ methods (Chase combining and CC-IR) in downlink and uplink transmissions."

REFERENCE

"Subclause 11.8.3.7.19 in IEEE 802.16e"
 ::= { wmanIf2BsSsOfdmaReqCapabilitiesEntry 24 }

```

1  wmanIf2BsSsOfdmaRspCapabilitiesTable OBJECT-TYPE
2      SYNTAX      SEQUENCE OF WmanIf2BsSsOfdmaRspCapabilitiesEntry
3      MAX-ACCESS  not-accessible
4      STATUS      current
5      DESCRIPTION
6          "This table contains the basic capability information,
7           specific to OFDMA Phy, of MSs that have been reported by
8           MSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.
9           Entries in this table should be created when an MS
10          registers with a BS."
11      ::= { wmanIf2BsOfdmaPhy 6 }
12
13  wmanIf2BsSsOfdmaRspCapabilitiesEntry OBJECT-TYPE
14      SYNTAX      WmanIf2BsSsOfdmaRspCapabilitiesEntry
15      MAX-ACCESS  not-accessible
16      STATUS      current
17      DESCRIPTION
18          "This table provides one row for each MS that has been
19          registered in the BS. This table augments the table
20          wmanIf2BsRegisteredSsTable."
21      AUGMENTS { wmanIf2BsRegisteredSsEntry }
22      ::= { wmanIf2BsSsOfdmaRspCapabilitiesTable 1 }
23
24  WmanIf2BsSsOfdmaRspCapabilitiesEntry ::= SEQUENCE {
25      wmanIf2BsSsOfdmaRspCapFftSizes      WmanIf2OfdmaFftSizes,
26      wmanIf2BsSsOfdmaRspCapDemodulator    WmanIf2OfdmaMsDeModType,
27      wmanIf2BsSsOfdmaRspCapModulator      WmanIf2OfdmaMsModType,
28      wmanIf2BsSsOfdmaRspCapNoHarqChannel  Unsigned32,
29      wmanIf2BsSsOfdmaRspCapPermutation    WmanIf2OfdmaPermutation,
30      wmanIf2BsSsOfdmaRspCapMobilityFeature WmanIf2eOfdmaMobility,
31      wmanIf2BsSsOfdmaRspCapMaxMacLevelDlFm WmanIf2eMaxMacLevel,
32      wmanIf2BsSsOfdmaRspCapMaxMacLevelUlFm WmanIf2eMaxMacLevel,
33      wmanIf2BsSsOfdmaRspCapDemMimo        WmanIf2eOfdmaDemMimo,
34      wmanIf2BsSsOfdmaRspCapMimoCapability WmanIf2eOfdmaMimoCap,
35      wmanIf2BsSsOfdmaRspCapUlMimo         WmanIf2eOfdmaUlMimo,
36      wmanIf2BsSsOfdmaRspCapPrivateMap     WmanIf2eOfdmaPrivMap,
37      wmanIf2BsSsOfdmaRspCapAasCapability  WmanIf2eOfdmaAasCap,
38      wmanIf2BsSsOfdmaRspCapCinrMeasurement WmanIf2eOfdmaCinrCap,
39      wmanIf2BsSsOfdmaRspCapUlPowerControl WmanIf2eOfdmaUlPower,
40      wmanIf2BsSsOfdmaRspCapMapCapability  WmanIf2eOfdmaMapCap,
41      wmanIf2BsSsOfdmaRspCapUlControlChannel WmanIf2eOfdmaUlCntlCh,
42      wmanIf2BsSsOfdmaRspCapCistCapability WmanIf2eOfdmaMsCistCap,
43      wmanIf2BsSsOfdmaRspCapMaxHarqBurst   WmanIf2eOfdmaMaxHarq,
44      wmanIf2BsSsOfdmaRspCapModMimo        WmanIf2eOfdmaModMimo,
45      wmanIf2BsSsOfdmaRspCapSdmaPilot      WmanIf2eSdmaPilotCap,
46      wmanIf2BsSsOfdmaRspCapMultipleBurst  WmanIf2eMultiBurst,
47      wmanIf2BsSsOfdmaRspCapIncrHarqBuffer WmanIf2eIncrHarqBuf,
48      wmanIf2BsSsOfdmaRspCapChaseHarqBuffer WmanIf2eChaseHarqBuf }
49
50  wmanIf2BsSsOfdmaRspCapFftSizes OBJECT-TYPE
51      SYNTAX      WmanIf2OfdmaFftSizes
52      MAX-ACCESS  read-only
53      STATUS      current
54      DESCRIPTION
55

```

```

1         "This field indicates the FFT sizes negotiated with the
2         MS."
3         ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 1 }
4
5
6 wmanIf2BsSsOfdmaRspCapDemodulator OBJECT-TYPE
7     SYNTAX      WmanIf2OfdmaMsDeModType
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "This field indicates the different demodulator options
12        negotiated for MS for downlink."
13        ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 2 }
14
15
16 wmanIf2BsSsOfdmaRspCapModulator OBJECT-TYPE
17     SYNTAX      WmanIf2OfdmaMsModType
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21        "This field indicates the different modulator options
22        negotiated for MS for uplink."
23        ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 3 }
24
25
26 wmanIf2BsSsOfdmaRspCapNoHarqChannel OBJECT-TYPE
27     SYNTAX      Unsigned32
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31        "This field specifies the number of uplink H-ARQ
32        channels (n) the SS supports, where n = 1..16.
33        The value of this object should be 0..15."
34        ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 4 }
35
36
37 wmanIf2BsSsOfdmaRspCapPermutation OBJECT-TYPE
38     SYNTAX      WmanIf2OfdmaPermutation
39     MAX-ACCESS  read-only
40     STATUS      current
41     DESCRIPTION
42        "This field indicates the OFDMA MS Permutation support
43        negotiated for MS."
44        ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 5 }
45
46
47 wmanIf2eBsSsOfdmaRspCapMobilityFeature OBJECT-TYPE
48     SYNTAX      WmanIf2eOfdmaMobility
49     MAX-ACCESS  read-only
50     STATUS      current
51     DESCRIPTION
52        "The field indicates the mobility hand-over, Sleepmode,
53        and Idle-mode negotiated for MS."
54        ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 6 }
55
56
57 wmanIf2eBsSsOfdmaRspCapMaxMacLevelDlFm OBJECT-TYPE
58     SYNTAX      WmanIf2eMaxMacLevel
59     MAX-ACCESS  read-only
60     STATUS      current
61
62
63
64
65

```



```

1      DESCRIPTION
2          "Maximum amount of MAC level data the MS is capable of
3            processing per DL frame. A value of 0 indicates such
4            limitation does not exist, except the limitation of
5            the physical medium"
6      REFERENCE
7          "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
8      DEFVAL      { 0 }
9      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 7 }
10
11  wmanIf2eBsSsOfdmaRspCapMaxMacLevelUlFm OBJECT-TYPE
12      SYNTAX      WmanIf2eMaxMacLevel
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "Maximum amount of MAC level data the MS is capable of
17            processing per UL frame. A value of 0 indicates such
18            limitation does not exist, except the limitation of
19            the physical medium"
20      REFERENCE
21          "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
22      DEFVAL      { 0 }
23      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 8 }
24
25  wmanIf2eBsSsOfdmaRspCapDemMimo OBJECT-TYPE
26      SYNTAX      WmanIf2eOfdmaDemMimo
27      MAX-ACCESS  read-only
28      STATUS      current
29      DESCRIPTION
30          "This field indicates the different MIMO options supported
31            by a WirelessMAN-OFDMA PHY SS in the downlink."
32      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 9 }
33
34  wmanIf2eBsSsOfdmaRspCapMimoCapability OBJECT-TYPE
35      SYNTAX      WmanIf2eOfdmaMimoCap
36      MAX-ACCESS  read-only
37      STATUS      current
38      DESCRIPTION
39          "This field indicates the MIMO capability of OFDMA MS
40            demodulator."
41      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 10 }
42
43  wmanIf2eBsSsOfdmaRspCapUlMimo OBJECT-TYPE
44      SYNTAX      WmanIf2eOfdmaUlMimo
45      MAX-ACCESS  read-only
46      STATUS      current
47      DESCRIPTION
48          "This field indicates different MIMO options supported
49            by a OFDMA PHY SS in the uplink"
50      REFERENCE
51          "Subclause 11.8.3.7.6 in IEEE 802.16e"
52      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 11 }
53
54  wmanIf2eBsSsOfdmaRspCapPrivateMap OBJECT-TYPE

```

```

1      SYNTAX      WmanIf2eOfdmaPrivMap
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "This field indicates AAS private map parameters
6          supported by a OFDMA SS"
7      REFERENCE
8          "Subclause 11.8.3.7.7 in IEEE 802.16e"
9      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 12 }
10
11
12
13  wmanIf2eBsSsOfdmaRspCapAasCapability OBJECT-TYPE
14      SYNTAX      WmanIf2eOfdmaAasCap
15      MAX-ACCESS  read-only
16      STATUS      current
17      DESCRIPTION
18          "This field indicates different AAS options
19          supported by a OFDMA PHY SS in the downlink"
20      REFERENCE
21          "Subclause 11.8.3.7.8 in IEEE 802.16e"
22      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 13 }
23
24
25
26  wmanIf2eBsSsOfdmaRspCapCinrMesurement OBJECT-TYPE
27      SYNTAX      WmanIf2eOfdmaCinrCap
28      MAX-ACCESS  read-only
29      STATUS      current
30      DESCRIPTION
31          "This field indicates the CINR measurement capability
32          supported by a OFDMA PHY SS in the downlink."
33      REFERENCE
34          "Subclause 11.8.3.7.9 in IEEE 802.16e"
35      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 14 }
36
37
38
39  wmanIf2eBsSsOfdmaRspCapUlPowerControl OBJECT-TYPE
40      SYNTAX      WmanIf2eOfdmaUlPower
41      MAX-ACCESS  read-only
42      STATUS      current
43      DESCRIPTION
44          "This field indicates the power control options
45          supported by a OFDMA PHY SS for uplink transmission."
46      REFERENCE
47          "Subclause 11.8.3.7.11 in IEEE 802.16e"
48      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 15 }
49
50
51
52  wmanIf2eBsSsOfdmaRspCapMapCapability OBJECT-TYPE
53      SYNTAX      WmanIf2eOfdmaMapCap
54      MAX-ACCESS  read-only
55      STATUS      current
56      DESCRIPTION
57          "This field indicates the different MAP options supported
58          by a OFDMA PHY SS"
59      REFERENCE
60          "Subclause 11.8.3.7.11 in IEEE 802.16e"
61      ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 16 }
62
63
64
65

```

```

1  wmanIf2eBsSsOfdmaRspCapUlControlChannel OBJECT-TYPE
2      SYNTAX      WmanIf2eOfdmaUlCntlCh
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This field indicates the different uplink control channels
7          supported by a OFDMA PHY SS."
8      REFERENCE
9          "Subclause 11.8.3.7.13 in IEEE 802.16e"
10     ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 17 }
11
12  wmanIf2eBsSsOfdmaRspCapCistCapability OBJECT-TYPE
13      SYNTAX      WmanIf2eOfdmaMsCistCap
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "This field indicates the MS capability of supporting CSIT
18          (uplink sounding)."
19      REFERENCE
20          "Subclause 11.8.3.7.14 in IEEE 802.16e"
21     ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 18 }
22
23  wmanIf2eBsSsOfdmaRspCapMaxHarqBurst OBJECT-TYPE
24      SYNTAX      WmanIf2eOfdmaMaxHarq
25      MAX-ACCESS  read-only
26      STATUS      current
27      DESCRIPTION
28          "This field indicates the maximum number of UL/DL HARQ
29          burst allocations for the SS in a single UL/DL subframe."
30      REFERENCE
31          "Subclause 11.8.3.7.15 in IEEE 802.16e"
32     ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 19 }
33
34  wmanIf2eBsSsOfdmaRspCapModMimo OBJECT-TYPE
35      SYNTAX      WmanIf2eOfdmaModMimo
36      MAX-ACCESS  read-only
37      STATUS      current
38      DESCRIPTION
39          "This field indicates the MIMO capability of OFDMA SS
40          modulator."
41      REFERENCE
42          "Subclause 11.8.3.7.16 in IEEE 802.16e"
43     ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 20 }
44
45  wmanIf2eBsSsOfdmaRspCapSdmaPilot OBJECT-TYPE
46      SYNTAX      WmanIf2eSdmaPilotCap
47      MAX-ACCESS  read-only
48      STATUS      current
49      DESCRIPTION
50          "This field indicates the SDMA pilot pattern support
51          for AMC zone."
52      REFERENCE
53          "Subclause 11.8.3.7.17 in IEEE 802.16e"
54     ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 21 }
55
56
57
58
59
60
61
62
63
64
65

```

```

1
2 wmanIf2eBsSsOfdmaRspCapMultipleBurst OBJECT-TYPE
3     SYNTAX      WmanIf2eMultiBurst
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "This field indicates whether multiple FEC types are
8         supported in DL/UL burst profiles."
9     REFERENCE
10        "Subclause 11.8.3.7.18 in IEEE 802.16e"
11    ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 22 }
12
13 wmanIf2eBsSsOfdmaRspCapIncrHarqBuffer OBJECT-TYPE
14     SYNTAX      WmanIf2eIncrHarqBuf
15     MAX-ACCESS  read-only
16     STATUS      current
17     DESCRIPTION
18         "This field indicates the maximal number of data
19         bits the SS is able to use for buffering for NEP/NSCH
20         based incremental redundancy CTC in downlink and uplink
21         transmissions."
22     REFERENCE
23        "Subclause 11.8.3.7.19 in IEEE 802.16e"
24    ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 23 }
25
26 wmanIf2eBsSsOfdmaRspCapChaseHarqBuffer OBJECT-TYPE
27     SYNTAX      WmanIf2eChaseHarqBuf
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31         "This field indicates the maximal number of data
32         bits the SS is able to use for buffering for
33         DIUC/duration based HARQ methods (Chase combining and
34         CC-IR) in downlink and uplink transmissions."
35     REFERENCE
36        "Subclause 11.8.3.7.19 in IEEE 802.16e"
37    ::= { wmanIf2BsSsOfdmaRspCapabilitiesEntry 24 }
38
39 wmanIf2BsOfdmaCapabilitiesTable OBJECT-TYPE
40     SYNTAX      SEQUENCE OF WmanIf2BsOfdmaCapabilitiesEntry
41     MAX-ACCESS  not-accessible
42     STATUS      current
43     DESCRIPTION
44         "This table contains the basic capabilities, specific to
45         OFDMA Phy, of the BS as implemented in BS hardware and
46         software. These capabilities along with the configuration
47         for them (wmanIf2BsOfdmaCapabilitiesConfigTable) are used
48         for negotiation of basic capabilities with SS using
49         RNG-RSP, SBC-RSP and REG-RSP messages. The negotiated
50         capabilities are obtained by interSubclause of MS raw
51         reported capabilities, BS raw capabilities and BS
52         configured capabilities. The objects in the table have
53         read-only access. The table is maintained by BS."
54    ::= { wmanIf2BsOfdmaPhy 7 }
55
56
57
58
59
60
61
62
63
64
65

```

```

1
2 wmanIf2BsOfdmaCapabilitiesEntry OBJECT-TYPE
3     SYNTAX      WmanIf2BsOfdmaCapabilitiesEntry
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "This table provides one row for each BS sector and is
8         indexed by ifIndex."
9     INDEX { ifIndex }
10    ::= { wmanIf2BsOfdmaCapabilitiesTable 1 }
11
12
13
14 WmanIf2BsOfdmaCapabilitiesEntry ::= SEQUENCE {
15     wmanIf2BsOfdmaCapFftSizes          WmanIf2OfdmaFftSizes,
16     wmanIf2BsOfdmaCapDemodulator        WmanIf2OfdmaMsDeModType,
17     wmanIf2BsOfdmaCapModulator          WmanIf2OfdmaMsModType,
18     wmanIf2BsOfdmaCapNoHarqChannel      Unsigned32,
19     wmanIf2BsOfdmaCapPermutation        WmanIf2OfdmaPermutation,
20     wmanIf2eBsOfdmaCapMobilityFeature   WmanIf2eOfdmaMobility,
21     wmanIf2eBsSsOfdmaCapMaxMacLevelDlFm WmanIf2eMaxMacLevel,
22     wmanIf2eBsSsOfdmaCapMaxMacLevelUlFm WmanIf2eMaxMacLevel,
23     wmanIf2eBsSsOfdmaCapDemMimo        WmanIf2eOfdmaDemMimo,
24     wmanIf2eBsSsOfdmaCapMimoCapability WmanIf2eOfdmaMimoCap,
25     wmanIf2eBsSsOfdmaCapUlMimo        WmanIf2eOfdmaUlMimo,
26     wmanIf2eBsSsOfdmaCapPrivateMap     WmanIf2eOfdmaPrivMap,
27     wmanIf2eBsSsOfdmaCapAasCapability   WmanIf2eOfdmaAasCap,
28     wmanIf2eBsSsOfdmaCapCinrMeasurement WmanIf2eOfdmaCinrCap,
29     wmanIf2eBsSsOfdmaCapUlPowerControl WmanIf2eOfdmaUlPower,
30     wmanIf2eBsSsOfdmaCapMapCapability   WmanIf2eOfdmaMapCap,
31     wmanIf2eBsSsOfdmaCapUlControlChannel WmanIf2eOfdmaUlCntlCh,
32     wmanIf2eBsSsOfdmaCapCistCapability  WmanIf2eOfdmaMsCistCap,
33     wmanIf2eBsSsOfdmaCapMaxHarqBurst    WmanIf2eOfdmaMaxHarq,
34     wmanIf2eBsSsOfdmaCapModMimo        WmanIf2eOfdmaModMimo,
35     wmanIf2eBsSsOfdmaCapSdmaPilot      WmanIf2eSdmaPilotCap,
36     wmanIf2eBsSsOfdmaCapMultipleBurst  WmanIf2eMultiBurst,
37     wmanIf2eBsSsOfdmaCapIncrHarqBuffer WmanIf2eIncrHarqBuf,
38     wmanIf2eBsSsOfdmaCapChaseHarqBuffer WmanIf2eChaseHarqBuf }
39
40
41
42
43
44
45 wmanIf2BsOfdmaCapFftSizes OBJECT-TYPE
46     SYNTAX      WmanIf2OfdmaFftSizes
47     MAX-ACCESS  read-only
48     STATUS      current
49     DESCRIPTION
50         "This field indicates the FFT sizes supported by BS."
51     ::= { wmanIf2BsOfdmaCapabilitiesEntry 1 }
52
53
54
55 wmanIf2BsOfdmaCapDemodulator OBJECT-TYPE
56     SYNTAX      WmanIf2OfdmaMsDeModType
57     MAX-ACCESS  read-only
58     STATUS      current
59     DESCRIPTION
60         "This field indicates the different demodulator options
61         supported by BS."
62     ::= { wmanIf2BsOfdmaCapabilitiesEntry 2 }
63
64
65

```

```

1  wmanIf2BsOfdmaCapModulator OBJECT-TYPE
2      SYNTAX      WmanIf2OfdmaMsModType
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This field indicates the different modulator options
7           supported by BS."
8      ::= { wmanIf2BsOfdmaCapabilitiesEntry 3 }
9
10
11  wmanIf2BsOfdmaCapNoHarqChannel OBJECT-TYPE
12      SYNTAX      Unsigned32
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "This field specifies the number of uplink H-ARQ
17           channels (n) the SS supports, where n = 1..16.
18           The value of this object should be 0..15."
19      ::= { wmanIf2BsOfdmaCapabilitiesEntry 4 }
20
21
22  wmanIf2BsOfdmaCapPermutation OBJECT-TYPE
23      SYNTAX      WmanIf2OfdmaPermutation
24      MAX-ACCESS  read-only
25      STATUS      current
26      DESCRIPTION
27          "This field indicates the OFDMA MS Permutation support
28           supported by BS."
29      ::= { wmanIf2BsOfdmaCapabilitiesEntry 5 }
30
31
32  wmanIf2eBsOfdmaCapMobilityFeature OBJECT-TYPE
33      SYNTAX      WmanIf2eOfdmaMobility
34      MAX-ACCESS  read-only
35      STATUS      current
36      DESCRIPTION
37          "The field indicates the mobility hand-over, Sleepmode,
38           and Idle-mode supported by BS."
39      ::= { wmanIf2BsOfdmaCapabilitiesEntry 6 }
40
41
42  wmanIf2eBsSsOfdmaCapMaxMacLevelDlFm OBJECT-TYPE
43      SYNTAX      WmanIf2eMaxMacLevel
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47          "Maximum amount of MAC level data the MS is capable of
48           processing per DL frame. A value of 0 indicates such
49           limitation does not exist, except the limitation of
50           the physical medium"
51      REFERENCE
52          "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
53      DEFVAL      { 0 }
54      ::= { wmanIf2BsOfdmaCapabilitiesEntry 7 }
55
56
57  wmanIf2eBsSsOfdmaCapMaxMacLevelUlFm OBJECT-TYPE
58      SYNTAX      WmanIf2eMaxMacLevel
59      MAX-ACCESS  read-only
60
61
62
63
64
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "Maximum amount of MAC level data the MS is capable of
4          processing per UL frame. A value of 0 indicates such
5          limitation does not exist, except the limitation of
6          the physical medium"
7
8      REFERENCE
9          "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
10
11     DEFVAL      { 0 }
12     ::= { wmanIf2BsOfdmaCapabilitiesEntry 8 }
13
14 wmanIf2eBsSsOfdmaCapDemMimo OBJECT-TYPE
15     SYNTAX      WmanIf2eOfdmaDemMimo
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "This field indicates the different MIMO options supported
20         by a WirelessMAN-OFDMA PHY SS in the downlink."
21     ::= { wmanIf2BsOfdmaCapabilitiesEntry 9 }
22
23
24
25 wmanIf2eBsSsOfdmaCapMimoCapability OBJECT-TYPE
26     SYNTAX      WmanIf2eOfdmaMimoCap
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30         "This field indicates the MIMO capability of OFDMA MS
31         demodulator."
32     ::= { wmanIf2BsOfdmaCapabilitiesEntry 10 }
33
34
35
36 wmanIf2eBsSsOfdmaCapUlMimo OBJECT-TYPE
37     SYNTAX      WmanIf2eOfdmaUlMimo
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41         "This field indicates different MIMO options supported
42         by a OFDMA PHY SS in the uplink"
43     REFERENCE
44         "Subclause 11.8.3.7.6 in IEEE 802.16e"
45     ::= { wmanIf2BsOfdmaCapabilitiesEntry 11 }
46
47
48
49 wmanIf2eBsSsOfdmaCapPrivateMap OBJECT-TYPE
50     SYNTAX      WmanIf2eOfdmaPrivMap
51     MAX-ACCESS  read-only
52     STATUS      current
53     DESCRIPTION
54         "This field indicates AAS private map parameters
55         supported by a OFDMA SS"
56     REFERENCE
57         "Subclause 11.8.3.7.7 in IEEE 802.16e"
58     ::= { wmanIf2BsOfdmaCapabilitiesEntry 12 }
59
60
61
62 wmanIf2eBsSsOfdmaCapAasCapability OBJECT-TYPE
63     SYNTAX      WmanIf2eOfdmaAasCap
64     MAX-ACCESS  read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This field indicates different AAS options
4          supported by a OFDMA PHY SS in the downlink"
5      REFERENCE
6          "Subclause 11.8.3.7.8 in IEEE 802.16e"
7          ::= { wmanIf2BsOfdmaCapabilitiesEntry 13 }
8
9
10
11      wmanIf2eBsSsOfdmaCapCinrMesurement OBJECT-TYPE
12          SYNTAX      WmanIf2eOfdmaCinrCap
13          MAX-ACCESS  read-only
14          STATUS      current
15          DESCRIPTION
16              "This field indicates the CINR measurement capability
17              supported by a OFDMA PHY SS in the downlink."
18          REFERENCE
19              "Subclause 11.8.3.7.9 in IEEE 802.16e"
20              ::= { wmanIf2BsOfdmaCapabilitiesEntry 14 }
21
22
23
24      wmanIf2eBsSsOfdmaCapUlPowerControl OBJECT-TYPE
25          SYNTAX      WmanIf2eOfdmaUlPower
26          MAX-ACCESS  read-only
27          STATUS      current
28          DESCRIPTION
29              "This field indicates the power control options
30              supported by a OFDMA PHY SS for uplink transmission."
31          REFERENCE
32              "Subclause 11.8.3.7.11 in IEEE 802.16e"
33              ::= { wmanIf2BsOfdmaCapabilitiesEntry 15 }
34
35
36
37      wmanIf2eBsSsOfdmaCapMapCapability OBJECT-TYPE
38          SYNTAX      WmanIf2eOfdmaMapCap
39          MAX-ACCESS  read-only
40          STATUS      current
41          DESCRIPTION
42              "This field indicates the different MAP options supported
43              by a OFDMA PHY SS"
44          REFERENCE
45              "Subclause 11.8.3.7.11 in IEEE 802.16e"
46              ::= { wmanIf2BsOfdmaCapabilitiesEntry 16 }
47
48
49
50      wmanIf2eBsSsOfdmaCapUlControlChannel OBJECT-TYPE
51          SYNTAX      WmanIf2eOfdmaUlCntlCh
52          MAX-ACCESS  read-only
53          STATUS      current
54          DESCRIPTION
55              "This field indicates the different uplink control channels
56              supported by a OFDMA PHY SS."
57          REFERENCE
58              "Subclause 11.8.3.7.13 in IEEE 802.16e"
59              ::= { wmanIf2BsOfdmaCapabilitiesEntry 17 }
60
61
62
63      wmanIf2eBsSsOfdmaCapCistCapability OBJECT-TYPE
64          SYNTAX      WmanIf2eOfdmaMsCistCap
65

```



```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "This field indicates the MS capability of supporting CSIT
5          (uplink sounding)."
```

REFERENCE

```

8          "Subclause 11.8.3.7.14 in IEEE 802.16e"
9          ::= { wmanIf2BsOfdmaCapabilitiesEntry 18 }
```

wmanIf2eBsSsOfdmaCapMaxHarqBurst OBJECT-TYPE

```

12     SYNTAX          WmanIf2eOfdmaMaxHarq
13     MAX-ACCESS      read-only
14     STATUS          current
15     DESCRIPTION
16         "This field indicates the maximum number of UL/DL HARQ
17         burst allocations for the SS in a single UL/DL subframe."
```

REFERENCE

```

21     "Subclause 11.8.3.7.15 in IEEE 802.16e"
22     ::= { wmanIf2BsOfdmaCapabilitiesEntry 19 }
```

wmanIf2eBsSsOfdmaCapModMimo OBJECT-TYPE

```

25     SYNTAX          WmanIf2eOfdmaModMimo
26     MAX-ACCESS      read-only
27     STATUS          current
28     DESCRIPTION
29         "This field indicates the MIMO capability of OFDMA SS
30         modulator."
```

REFERENCE

```

34     "Subclause 11.8.3.7.16 in IEEE 802.16e"
35     ::= { wmanIf2BsOfdmaCapabilitiesEntry 20 }
```

wmanIf2eBsSsOfdmaCapSdmaPilot OBJECT-TYPE

```

38     SYNTAX          WmanIf2eSdmaPilotCap
39     MAX-ACCESS      read-only
40     STATUS          current
41     DESCRIPTION
42         "This field indicates the SDMA pilot pattern support
43         for AMC zone."
```

REFERENCE

```

48     "Subclause 11.8.3.7.17 in IEEE 802.16e"
49     ::= { wmanIf2BsOfdmaCapabilitiesEntry 21 }
```

wmanIf2eBsSsOfdmaCapMultipleBurst OBJECT-TYPE

```

52     SYNTAX          WmanIf2eMultiBurst
53     MAX-ACCESS      read-only
54     STATUS          current
55     DESCRIPTION
56         "This field indicates whether multiple FEC types are
57         supported in DL/UL burst profiles."
```

REFERENCE

```

61     "Subclause 11.8.3.7.18 in IEEE 802.16e"
62     ::= { wmanIf2BsOfdmaCapabilitiesEntry 22 }
```

wmanIf2eBsSsOfdmaCapIncrHarqBuffer OBJECT-TYPE

```

1      SYNTAX      WmanIf2eIncrHarqBuf
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "This field indicates the maximal number of data
6          bits the SS is able to use for buffering for NEP/NSCH
7          based incremental redundancy CTC in downlink and uplink
8          transmissions."
9
10     REFERENCE
11         "Subclause 11.8.3.7.19 in IEEE 802.16e"
12     ::= { wmanIf2BsOfdmaCapabilitiesEntry 23 }
13
14
15 wmanIf2eBsSsOfdmaCapChaseHarqBuffer OBJECT-TYPE
16     SYNTAX      WmanIf2eChaseHarqBuf
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20         "This field indicates the maximal number of data
21         bits the SS is able to use for buffering for
22         DIUC/duration based HARQ methods (Chase combining and
23         CC-IR) in downlink and uplink transmissions."
24
25     REFERENCE
26         "Subclause 11.8.3.7.19 in IEEE 802.16e"
27     ::= { wmanIf2BsOfdmaCapabilitiesEntry 24 }
28
29
30
31 wmanIf2BsOfdmaCapabilitiesConfigTable OBJECT-TYPE
32     SYNTAX      SEQUENCE OF WmanIf2BsOfdmaCapabilitiesConfigEntry
33     MAX-ACCESS  not-accessible
34     STATUS      current
35     DESCRIPTION
36         "This table contains the configuration for basic
37         capabilities of BS, specific to OFDMA Phy. The table is
38         intended to be used to restrict the Capabilities
39         implemented by BS, for example in order to comply with
40         local regulatory requirements. The BS should use the
41         configuration along with the implemented Capabilities
42         (wmanIf2BsOfdmaPhyTable) for negotiation of basic
43         capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP
44         messages. The negotiated capabilities are obtained by
45         interSubclause of MS reported capabilities, BS raw
46         capabilities and BS configured capabilities. The objects
47         in the table have read-write access. The rows are created
48         by BS as a copy of wmanIf2BsBasicCapabilitiesTable
49         and can be modified by NMS."
50     ::= { wmanIf2BsOfdmaPhy 8 }
51
52
53
54
55
56 wmanIf2BsOfdmaCapabilitiesConfigEntry OBJECT-TYPE
57     SYNTAX      WmanIf2BsOfdmaCapabilitiesConfigEntry
58     MAX-ACCESS  not-accessible
59     STATUS      current
60     DESCRIPTION
61         "This table provides one row for each BS sector and is
62         indexed by ifIndex."
63     INDEX { ifIndex }
64
65

```

```

1      ::= { wmanIf2BsOfdmaCapabilitiesConfigTable 1 }
2
3
4  WmanIf2BsOfdmaCapabilitiesConfigEntry ::= SEQUENCE {
5      wmanIf2BsOfdmaCapCfgFftSizes          WmanIf2OfdmaFftSizes,
6      wmanIf2BsOfdmaCapCfgDemodulator       WmanIf2OfdmaMsDeModType,
7      wmanIf2BsOfdmaCapCfgModulator         WmanIf2OfdmaMsModType,
8      wmanIf2BsOfdmaCapCfgNoHarqChannel     Unsigned32,
9      wmanIf2BsOfdmaCapCfgPermutation       WmanIf2OfdmaPermutation,
10     wmanIf2eBsOfdmaCapCfgMobilityFeature   WmanIf2eOfdmaMobility,
11     wmanIf2eBsSsOfdmaCapCfgMaxMacLevelDlFm WmanIf2eMaxMacLevel,
12     wmanIf2eBsSsOfdmaCapCfgMaxMacLevelUlFm WmanIf2eMaxMacLevel,
13     wmanIf2eBsSsOfdmaCapCfgDemMimo        WmanIf2eOfdmaDemMimo,
14     wmanIf2eBsSsOfdmaCapCfgMimoCapability WmanIf2eOfdmaMimoCap,
15     wmanIf2eBsSsOfdmaCapCfgUlMimo         WmanIf2eOfdmaUlMimo,
16     wmanIf2eBsSsOfdmaCapCfgPrivateMap     WmanIf2eOfdmaPrivMap,
17     wmanIf2eBsSsOfdmaCapCfgAasCapability  WmanIf2eOfdmaAasCap,
18     wmanIf2eBsSsOfdmaCapCfgCinrMeasurement WmanIf2eOfdmaCinrCap,
19     wmanIf2eBsSsOfdmaCapCfgUlPowerControl WmanIf2eOfdmaUlPower,
20     wmanIf2eBsSsOfdmaCapCfgMapCapability  WmanIf2eOfdmaMapCap,
21     wmanIf2eBsSsOfdmaCapCfgUlControlChannel WmanIf2eOfdmaUlCntlCh,
22     wmanIf2eBsSsOfdmaCapCfgCistCapability WmanIf2eOfdmaMsCistCap,
23     wmanIf2eBsSsOfdmaCapCfgMaxHarqBurst   WmanIf2eOfdmaMaxHarq,
24     wmanIf2eBsSsOfdmaCapCfgModMimo        WmanIf2eOfdmaModMimo,
25     wmanIf2eBsSsOfdmaCapCfgSdmaPilot      WmanIf2eSdmaPilotCap,
26     wmanIf2eBsSsOfdmaCapCfgMultipleBurst  WmanIf2eMultiBurst,
27     wmanIf2eBsSsOfdmaCapCfgIncrHarqBuffer WmanIf2eIncrHarqBuf,
28     wmanIf2eBsSsOfdmaCapCfgChaseHarqBuffer WmanIf2eChaseHarqBuf }
29
30
31
32
33
34
35  wmanIf2BsOfdmaCapCfgFftSizes OBJECT-TYPE
36      SYNTAX          WmanIf2OfdmaFftSizes
37      MAX-ACCESS      read-write
38      STATUS          current
39      DESCRIPTION
40          "This field indicates the FFT sizes configured for the BS."
41      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 1 }
42
43
44  wmanIf2BsOfdmaCapCfgDemodulator OBJECT-TYPE
45      SYNTAX          WmanIf2OfdmaMsDeModType
46      MAX-ACCESS      read-write
47      STATUS          current
48      DESCRIPTION
49          "This field indicates the different demodulator options
50          configured for the BS."
51      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 2 }
52
53
54
55  wmanIf2BsOfdmaCapCfgModulator OBJECT-TYPE
56      SYNTAX          WmanIf2OfdmaMsModType
57      MAX-ACCESS      read-write
58      STATUS          current
59      DESCRIPTION
60          "This field indicates the different modulator options
61          configured for the BS."
62      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 3 }
63
64
65

```

```

1  wmanIf2BsOfdmaCapCfgNoHarqChannel OBJECT-TYPE
2      SYNTAX      Unsigned32
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This field specifies the number of uplink H-ARQ
7          channels (n) the SS supports, where n = 1..16.
8          The value of this object should be 0..15."
9      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 4 }
10
11
12
13  wmanIf2BsOfdmaCapCfgPermutation OBJECT-TYPE
14      SYNTAX      WmanIf2OfdmaPermutation
15      MAX-ACCESS  read-write
16      STATUS      current
17      DESCRIPTION
18          "This field indicates the OFDMA MS Permutation support
19          configured for the BS."
20      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 5 }
21
22
23
24  wmanIf2eBsOfdmaCapCfgMobilityFeature OBJECT-TYPE
25      SYNTAX      WmanIf2eOfdmaMobility
26      MAX-ACCESS  read-write
27      STATUS      current
28      DESCRIPTION
29          "The field indicates the mobility hand-over, Sleepmode,
30          and Idle-mode configured for the BS."
31      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 6 }
32
33
34
35  wmanIf2eBsSsOfdmaCapCfgMaxMacLevelDlFm OBJECT-TYPE
36      SYNTAX      WmanIf2eMaxMacLevel
37      MAX-ACCESS  read-write
38      STATUS      current
39      DESCRIPTION
40          "Maximum amount of MAC level data the MS is capable of
41          processing per DL frame. A value of 0 indicates such
42          limitation does not exist, except the limitation of
43          the physical medium"
44      REFERENCE
45          "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
46      DEFVAL      { 0 }
47      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 7 }
48
49
50
51  wmanIf2eBsSsOfdmaCapCfgMaxMacLevelUlFm OBJECT-TYPE
52      SYNTAX      WmanIf2eMaxMacLevel
53      MAX-ACCESS  read-write
54      STATUS      current
55      DESCRIPTION
56          "Maximum amount of MAC level data the MS is capable of
57          processing per UL frame. A value of 0 indicates such
58          limitation does not exist, except the limitation of
59          the physical medium"
60      REFERENCE
61          "Subclause 11.7.8.10.1 in IEEE Std 802.16e-2005"
62      DEFVAL      { 0 }
63
64
65

```

```

1      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 8 }
2
3  wmanIf2eBsSsOfdmaCapCfgDemMimo OBJECT-TYPE
4      SYNTAX      WmanIf2eOfdmaDemMimo
5      MAX-ACCESS  read-write
6      STATUS      current
7      DESCRIPTION
8          "This field indicates the different MIMO options supported
9          by a WirelessMAN-OFDMA PHY SS in the downlink."
10     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 9 }
11
12
13
14  wmanIf2eBsSsOfdmaCapCfgMimoCapability OBJECT-TYPE
15      SYNTAX      WmanIf2eOfdmaMimoCap
16      MAX-ACCESS  read-write
17      STATUS      current
18      DESCRIPTION
19          "This field indicates the MIMO capability of OFDMA MS
20          demodulator."
21     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 10 }
22
23
24
25  wmanIf2eBsSsOfdmaCapCfgUlMimo OBJECT-TYPE
26      SYNTAX      WmanIf2eOfdmaUlMimo
27      MAX-ACCESS  read-write
28      STATUS      current
29      DESCRIPTION
30          "This field indicates different MIMO options supported
31          by a OFDMA PHY SS in the uplink"
32     REFERENCE
33         "Subclause 11.8.3.7.6 in IEEE 802.16e"
34     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 11 }
35
36
37
38  wmanIf2eBsSsOfdmaCapCfgPrivateMap OBJECT-TYPE
39      SYNTAX      WmanIf2eOfdmaPrivMap
40      MAX-ACCESS  read-write
41      STATUS      current
42      DESCRIPTION
43          "This field indicates AAS private map parameters
44          supported by a OFDMA SS"
45     REFERENCE
46         "Subclause 11.8.3.7.7 in IEEE 802.16e"
47     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 12 }
48
49
50
51  wmanIf2eBsSsOfdmaCapCfgAasCapability OBJECT-TYPE
52      SYNTAX      WmanIf2eOfdmaAasCap
53      MAX-ACCESS  read-write
54      STATUS      current
55      DESCRIPTION
56          "This field indicates different AAS options
57          supported by a OFDMA PHY SS in the downlink"
58     REFERENCE
59         "Subclause 11.8.3.7.8 in IEEE 802.16e"
60     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 13 }
61
62
63
64  wmanIf2eBsSsOfdmaCapCfgCinrMeasurement OBJECT-TYPE
65

```

```

1      SYNTAX      WmanIf2eOfdmaCinrCap
2      MAX-ACCESS  read-write
3      STATUS      current
4      DESCRIPTION
5          "This field indicates the CINR measurement capability
6           supported by a OFDMA PHY SS in the downlink."
7      REFERENCE
8          "Subclause 11.8.3.7.9 in IEEE 802.16e"
9      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 14 }
10
11  wmanIf2eBsSsOfdmaCapCfgUlPowerControl OBJECT-TYPE
12      SYNTAX      WmanIf2eOfdmaUlPower
13      MAX-ACCESS  read-write
14      STATUS      current
15      DESCRIPTION
16          "This field indicates the power control options
17           supported by a OFDMA PHY SS for uplink transmission."
18      REFERENCE
19          "Subclause 11.8.3.7.11 in IEEE 802.16e"
20      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 15 }
21
22  wmanIf2eBsSsOfdmaCapCfgMapCapability OBJECT-TYPE
23      SYNTAX      WmanIf2eOfdmaMapCap
24      MAX-ACCESS  read-write
25      STATUS      current
26      DESCRIPTION
27          "This field indicates the different MAP options supported
28           by a OFDMA PHY SS"
29      REFERENCE
30          "Subclause 11.8.3.7.11 in IEEE 802.16e"
31      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 16 }
32
33  wmanIf2eBsSsOfdmaCapCfgUlControlChannel OBJECT-TYPE
34      SYNTAX      WmanIf2eOfdmaUlCntlCh
35      MAX-ACCESS  read-write
36      STATUS      current
37      DESCRIPTION
38          "This field indicates the different uplink control channels
39           supported by a OFDMA PHY SS."
40      REFERENCE
41          "Subclause 11.8.3.7.13 in IEEE 802.16e"
42      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 17 }
43
44  wmanIf2eBsSsOfdmaCapCfgCistCapability OBJECT-TYPE
45      SYNTAX      WmanIf2eOfdmaMsCistCap
46      MAX-ACCESS  read-write
47      STATUS      current
48      DESCRIPTION
49          "This field indicates the MS capability of supporting CSIT
50           (uplink sounding)."
51      REFERENCE
52          "Subclause 11.8.3.7.14 in IEEE 802.16e"
53      ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 18 }
54
55
56
57
58
59
60
61
62
63
64
65

```

```

1  wmanIf2eBsSsOfdmaCapCfgMaxHarqBurst OBJECT-TYPE
2      SYNTAX      WmanIf2eOfdmaMaxHarq
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "This field indicates the maximum number of UL/DL HARQ
7          burst allocations for the SS in a single UL/DL subframe."
8      REFERENCE
9          "Subclause 11.8.3.7.15 in IEEE 802.16e"
10     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 19 }
11
12  wmanIf2eBsSsOfdmaCapCfgModMimo OBJECT-TYPE
13      SYNTAX      WmanIf2eOfdmaModMimo
14      MAX-ACCESS  read-write
15      STATUS      current
16      DESCRIPTION
17          "This field indicates the MIMO capability of OFDMA SS
18          modulator."
19      REFERENCE
20          "Subclause 11.8.3.7.16 in IEEE 802.16e"
21     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 20 }
22
23  wmanIf2eBsSsOfdmaCapCfgSdmaPilot OBJECT-TYPE
24      SYNTAX      WmanIf2eSdmaPilotCap
25      MAX-ACCESS  read-write
26      STATUS      current
27      DESCRIPTION
28          "This field indicates the SDMA pilot pattern support
29          for AMC zone."
30      REFERENCE
31          "Subclause 11.8.3.7.17 in IEEE 802.16e"
32     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 21 }
33
34  wmanIf2eBsSsOfdmaCapCfgMultipleBurst OBJECT-TYPE
35      SYNTAX      WmanIf2eMultiBurst
36      MAX-ACCESS  read-write
37      STATUS      current
38      DESCRIPTION
39          "This field indicates whether multiple FEC types are
40          supported in DL/UL burst profiles."
41      REFERENCE
42          "Subclause 11.8.3.7.18 in IEEE 802.16e"
43     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 22 }
44
45  wmanIf2eBsSsOfdmaCapCfgIncrHarqBuffer OBJECT-TYPE
46      SYNTAX      WmanIf2eIncrHarqBuf
47      MAX-ACCESS  read-write
48      STATUS      current
49      DESCRIPTION
50          "This field indicates the maximal number of data
51          bits the SS is able to use for buffering for NEP/NSCH
52          based incremental redundancy CTC in downlink and uplink
53          transmissions."
54      REFERENCE
55

```

```

1         "Subclause 11.8.3.7.19 in IEEE 802.16e"
2         ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 23 }
3
4
5 wmanIf2eBsSsOfdmaCapCfgChaseHarqBuffer OBJECT-TYPE
6     SYNTAX      WmanIf2eChaseHarqBuf
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "This field indicates the maximal number of data
11        bits the SS is able to use for buffering for
12        DIUC/duration based HARQ methods (Chase combining and
13        CC-IR) in downlink and uplink transmissions."
14
15     REFERENCE
16        "Subclause 11.8.3.7.19 in IEEE 802.16e"
17     ::= { wmanIf2BsOfdmaCapabilitiesConfigEntry 24 }
18
19
20 wmanIf2eBsOfdmaUplinkChannelTable OBJECT-TYPE
21     SYNTAX      SEQUENCE OF WmanIf2eBsOfdmaUplinkChannelEntry
22     MAX-ACCESS  not-accessible
23     STATUS      current
24     DESCRIPTION
25        "This table contains UCD channel attributes, defining the
26        transmission characteristics of uplink channels"
27
28     REFERENCE
29        "Table 349 and Table 353, in IEEE Std 802.16-2004"
30     ::= { wmanIf2BsOfdmaPhy 9 }
31
32
33 wmanIf2eBsOfdmaUplinkChannelEntry OBJECT-TYPE
34     SYNTAX      WmanIf2eBsOfdmaUplinkChannelEntry
35     MAX-ACCESS  not-accessible
36     STATUS      current
37     DESCRIPTION
38        "This table provides one row for each uplink channel of
39        multi-sector BS, and is indexed by BS ifIndex. An entry
40        in this table exists for each ifEntry of BS with an
41        ifType of propBWAp2Mp."
42
43     AUGMENTS { wmanIf2BsOfdmaUplinkChannelEntry }
44     ::= { wmanIf2eBsOfdmaUplinkChannelTable 1 }
45
46
47
48 WmanIf2eBsOfdmaUplinkChannelEntry ::= SEQUENCE {
49     wmanIf2eBsOfdmaHandoverRangingStart    INTEGER,
50     wmanIf2eBsOfdmaHandoverRangingEnd      INTEGER,
51     wmanIf2eBsOfdmaHARQAckDelayDLBurst     WmanIf2eHarqAckDelay,
52     wmanIf2eBsOfdmaUlAmcAlloPhyBandsBitmap OCTET STRING,
53     wmanIf2eBsOfdmaMaxRetransmission        INTEGER,
54     wmanIf2eBsOfdmaNormalizedCnOverride     OCTET STRING,
55     wmanIf2eBsOfdmaSizeOfCqichId           INTEGER,
56     wmanIf2eBsOfdmaNormalizedCnValue       INTEGER,
57     wmanIf2eBsOfdmaNormalizedCnOverride2   OCTET STRING,
58     wmanIf2eBsOfdmaBandAmcEntryAvgCinr     INTEGER,
59     wmanIf2eBsOfdmaAasPreambleUpperBond    INTEGER,
60     wmanIf2eBsOfdmaAasPreambleLowerBond    INTEGER,
61     wmanIf2eBsOfdmaAasBeamSelectAllowed    WmanIf2eAasBeamSel,
62     wmanIf2eBsOfdmaCqichIndicationFlag     OCTET STRING,
63
64
65

```



```

1      wmanIf2eBsOfdmaUpPowerAdjStep      Unsigned32,
2      wmanIf2eBsOfdmaDownPowerAdjStep    Unsigned32,
3      wmanIf2eBsOfdmaMinPowerOffsetAdj    INTEGER,
4      wmanIf2eBsOfdmaMaxPowerOffsetAdj    INTEGER,
5      wmanIf2eBsOfdmaHandoverRngCodes     INTEGER,
6      wmanIf2eBsOfdmaTxPwrRepThreshold     INTEGER,
7      wmanIf2eBsOfdmaTprPower             INTEGER,
8      wmanIf2eBsOfdmaAlphaPavg            INTEGER,
9      wmanIf2eBsOfdmaCqichTxPwrRepThreshold INTEGER,
10     wmanIf2eBsOfdmaCqichTprPower         INTEGER,
11     wmanIf2eBsOfdmaCqichAlphaPavg        INTEGER,
12     wmanIf2eBsOfdmaNormalizedCnChSounding INTEGER,
13     wmanIf2eBsOfdmaInitialRngInterval    INTEGER,
14     wmanIf2eBsOfdmaInitialRngBackoffStart INTEGER,
15     wmanIf2eBsOfdmaInitialRngBackoffEnd  INTEGER,
16     wmanIf2eBsOfdmaBwRequestBackoffStart INTEGER,
17     wmanIf2eBsOfdmaBwRequestBackoffEnd   INTEGER}
18
19
20
21
22
23     wmanIf2eBsOfdmaHandoverRangingStart OBJECT-TYPE
24         SYNTAX      INTEGER (0..15)
25         MAX-ACCESS  read-write
26         STATUS      current
27         DESCRIPTION
28             "Initial backoff window size for MS performing initial
29             ranging during handover process, expressed as a power
30             of 2."
31         REFERENCE
32             "Table 349, in IEEE Std 802.16e-2005"
33             ::= { wmanIf2eBsOfdmaUplinkChannelEntry 1 }
34
35
36
37     wmanIf2eBsOfdmaHandoverRangingEnd OBJECT-TYPE
38         SYNTAX      INTEGER (0..15)
39         MAX-ACCESS  read-write
40         STATUS      current
41         DESCRIPTION
42             "Final backoff window size for MS performing initial
43             ranging during handover process, expressed as a power
44             of 2."
45         REFERENCE
46             "Table 349, in IEEE Std 802.16e-2005"
47             ::= { wmanIf2eBsOfdmaUplinkChannelEntry 2 }
48
49
50
51     wmanIf2eBsOfdmaHARQAckDelayDLBurst OBJECT-TYPE
52         SYNTAX      WmanIf2eHarqAckDelay
53         MAX-ACCESS  read-write
54         STATUS      current
55         DESCRIPTION
56             "This object defines the OFDMA H-ARQ ACK delay for DL
57             burst."
58         REFERENCE
59             "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
60             ::= { wmanIf2eBsOfdmaUplinkChannelEntry 3 }
61
62
63
64     wmanIf2eBsOfdmaUlAmcAlloPhyBandsBitmap OBJECT-TYPE
65

```

```

1      SYNTAX      OCTET STRING (SIZE (6))
2      MAX-ACCESS  read-write
3      STATUS      current
4      DESCRIPTION
5          "A bitmap describing the physical bands allocated to the
6           segment in the UL, when using the optional AMC permutation
7           with regular MAPs (see 8.4.6.3). The LSB of the first byte
8           shall correspond to the physical band 0. For any bit that
9           is not set, the corresponding physical bands shall not be
10          used by the SS on that segment. When this TLV is not
11          present, BS may allocate any physical bands to an SS."
12      REFERENCE
13          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
14      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 4 }
15
16  wmanIf2eBsOfdmaMaxRetransmission OBJECT-TYPE
17      SYNTAX      INTEGER (1..255)
18      MAX-ACCESS  read-write
19      STATUS      current
20      DESCRIPTION
21          "Maximum number of retransmission in UL HARQ."
22      REFERENCE
23          "Table 353, in IEEE Std 802.16e-2005"
24      DEFVAL      { 4 }
25      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 5 }
26
27  wmanIf2eBsOfdmaNormalizedCnOverride OBJECT-TYPE
28      SYNTAX      OCTET STRING (SIZE (8))
29      MAX-ACCESS  read-write
30      STATUS      current
31      DESCRIPTION
32          "This is a list of numbers, where each number is encoded by
33           one nibble, and interpreted as a signed integer. The
34           nibbles correspond in order to the list define by Table
35           334, starting from the second line, such that the LS
36           nibble of the first byte corresponds to the second line in
37           the table. The number encoded by each nibble represents
38           the difference in normalized C/N relative to the previous
39           line in the table."
40      REFERENCE
41          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
42      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 6 }
43
44  wmanIf2eBsOfdmaSizeOfCqichId OBJECT-TYPE
45      SYNTAX      INTEGER (0..7)
46      MAX-ACCESS  read-write
47      STATUS      current
48      DESCRIPTION
49          "Size of CQICH ID field.
50           0 = 0 bits
51           1 = 3 bits
52           2 = 4 bits
53           3 = 5 bits
54           4 = 6 bits
55
56
57
58
59
60
61
62
63
64
65

```

```

1           5 = 7 bits
2           6 = 8 bits
3           7 = 9 bits"
4
5     REFERENCE
6         "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
7     DEFVAL      { 0 }
8         ::= { wmanIf2eBsOfdmaUplinkChannelEntry 7 }
9
10
11 wmanIf2eBsOfdmaNormalizedCnValue OBJECT-TYPE
12     SYNTAX      INTEGER (-128..128)
13     UNITS       "dB"
14     MAX-ACCESS  read-write
15     STATUS      current
16     DESCRIPTION
17         "It shall be interpreted as signed integer in dB. It
18         corresponds to the normalized C/N value in the first line
19         (counting except for header cell of table)"
20     REFERENCE
21         "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
22     ::= { wmanIf2eBsOfdmaUplinkChannelEntry 8 }
23
24
25
26 wmanIf2eBsOfdmaNormalizedCnOverride2 OBJECT-TYPE
27     SYNTAX      OCTET STRING (SIZE (7))
28     MAX-ACCESS  read-write
29     STATUS      current
30     DESCRIPTION
31         "This is a list of numbers, where each number is encoded
32         by one nibble, and interpreted as a signed integer. The
33         nibbles correspond in order to the list define by Table
34         334, starting from the second line (counting except for
35         the header cell of table), such that the LS nibble of
36         the first byte corresponds to the second line in the
37         table. The number encoded by each nibble represents the
38         difference in normalized C/N relative to the previous
39         line in the table."
40     REFERENCE
41         "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
42     ::= { wmanIf2eBsOfdmaUplinkChannelEntry 9 }
43
44
45
46
47
48 wmanIf2eBsOfdmaBandAmcEntryAvgCinr OBJECT-TYPE
49     SYNTAX      INTEGER (-128..128)
50     UNITS       "dB"
51     MAX-ACCESS  read-write
52     STATUS      current
53     DESCRIPTION
54         "Threshold of the average CINR of the whole bandwidth to
55         trigger mode transition from normal subchannel to AMC"
56     REFERENCE
57         "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
58     ::= { wmanIf2eBsOfdmaUplinkChannelEntry 10 }
59
60
61
62 wmanIf2eBsOfdmaAasPreambleUpperBond OBJECT-TYPE
63     SYNTAX      INTEGER (-128..128)
64     UNITS       "0.25 dB"
65

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "Upper bound of AAS preamble."
5      REFERENCE
6          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
7      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 11 }
8
9
10
11  wmanIf2eBsOfdmaAasPreambleLowerBond OBJECT-TYPE
12      SYNTAX      INTEGER (-128..128)
13      UNITS        "0.25 dB"
14      MAX-ACCESS  read-write
15      STATUS      current
16      DESCRIPTION
17          "Lower bound of AAS preamble."
18      REFERENCE
19          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
20      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 12 }
21
22
23
24  wmanIf2eBsOfdmaAasBeamSelectAllowed OBJECT-TYPE
25      SYNTAX      WmanIf2eAasBeamSel
26      UNITS        "0.25 dB"
27      MAX-ACCESS  read-write
28      STATUS      current
29      DESCRIPTION
30          "Indicate whether unsolicited AAS Beam Select messages
31          (see 6.3.2.3.41 in IEEE 802.16e-2005) should be sent by
32          the MS."
33      REFERENCE
34          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
35      DEFVAL      { allowed }
36      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 13 }
37
38
39
40
41  wmanIf2eBsOfdmaCqichIndicationFlag OBJECT-TYPE
42      SYNTAX      OCTET STRING (SIZE (1))
43      MAX-ACCESS  read-write
44      STATUS      current
45      DESCRIPTION
46          "The N MSB values of this field represents the N-bit
47          payload value on the Fast-Feedback channel reserved as
48          indication flag for MS to initiate feedback on the
49          Feedback header, where N is the number of payload bits
50          used for S/N measurement feedback on the Fast-Feedback
51          channel. The value shall not be set to all zeros."
52      REFERENCE
53          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
54      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 14 }
55
56
57
58
59  wmanIf2eBsOfdmaUpPowerAdjStep OBJECT-TYPE
60      SYNTAX      Unsigned32
61      UNITS        "0.01 dB"
62      MAX-ACCESS  read-write
63      STATUS      current
64      DESCRIPTION
65

```

```

1         "MS-specific up power offset adjustment step"
2     REFERENCE
3         "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
4     ::= { wmanIf2eBsOfdmaUplinkChannelEntry 15 }
5
6
7     wmanIf2eBsOfdmaDownPowerAdjStep OBJECT-TYPE
8         SYNTAX      Unsigned32
9         UNITS        "0.01 dB"
10        MAX-ACCESS   read-write
11        STATUS        current
12        DESCRIPTION
13            "MS-specific down power offset adjustment step"
14        REFERENCE
15            "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
16        ::= { wmanIf2eBsOfdmaUplinkChannelEntry 16 }
17
18
19
20    wmanIf2eBsOfdmaMinPowerOffsetAdj OBJECT-TYPE
21        SYNTAX      INTEGER
22        UNITS        "0.1 dB"
23        MAX-ACCESS   read-write
24        STATUS        current
25        DESCRIPTION
26            "Minimum level of power offset adjustment"
27        REFERENCE
28            "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
29        ::= { wmanIf2eBsOfdmaUplinkChannelEntry 17 }
30
31
32
33    wmanIf2eBsOfdmaMaxPowerOffsetAdj OBJECT-TYPE
34        SYNTAX      INTEGER
35        UNITS        "0.1 dB"
36        MAX-ACCESS   read-write
37        STATUS        current
38        DESCRIPTION
39            "Minimum level of power offset adjustment"
40        REFERENCE
41            "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
42        ::= { wmanIf2eBsOfdmaUplinkChannelEntry 18 }
43
44
45
46    wmanIf2eBsOfdmaHandoverRngCodes OBJECT-TYPE
47        SYNTAX      INTEGER (0..255)
48        MAX-ACCESS   read-write
49        STATUS        current
50        DESCRIPTION
51            "Number of handover ranging CDMA codes"
52        REFERENCE
53            "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
54        ::= { wmanIf2eBsOfdmaUplinkChannelEntry 19 }
55
56
57
58    wmanIf2eBsOfdmaInitialRngInterval OBJECT-TYPE
59        SYNTAX      INTEGER
60        MAX-ACCESS   read-write
61        STATUS        current
62        DESCRIPTION
63            "Number of frames between initial ranging interval
64
65

```

```

1         allocation."
2     REFERENCE
3         "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
4     ::= { wmanIf2eBsOfdmaUplinkChannelEntry 20 }
5
6
7     wmanIf2eBsOfdmaTxPwrRepThreshold OBJECT-TYPE
8         SYNTAX      INTEGER (0..15)
9         UNITS        "dB"
10        MAX-ACCESS   read-write
11        STATUS        current
12        DESCRIPTION
13            "Tx power report threshold.
14             wmanIf2eBsOfdmaTxPwrRepThreshold = 0b1111 means infinite."
15        REFERENCE
16            "Subclause 11.3.1, Table 353, and 8.4.10.3.2.1 in IEEE
17             Std 802.16e-2005"
18        ::= { wmanIf2eBsOfdmaUplinkChannelEntry 21 }
19
20
21
22
23     wmanIf2eBsOfdmaTprPower OBJECT-TYPE
24         SYNTAX      INTEGER (0..15)
25         UNITS        "dB"
26         MAX-ACCESS   read-write
27         STATUS        current
28         DESCRIPTION
29             "Tx power report interval = 2 ^ wmanIf2eBsOfdmaTprPower.
30              The unit of Tx power report interval is frame.
31              wmanIf2eBsOfdmaTprPower = 0b1111 means infinite."
32         REFERENCE
33             "Subclause 11.3.1, Table 353, and 8.4.10.3.2.1 in IEEE
34              Std 802.16e-2005"
35         ::= { wmanIf2eBsOfdmaUplinkChannelEntry 22 }
36
37
38
39
40     wmanIf2eBsOfdmaAlphaPavg OBJECT-TYPE
41         SYNTAX      INTEGER (0..15)
42         UNITS        "dB"
43         MAX-ACCESS   read-write
44         STATUS        current
45         DESCRIPTION
46             "Alpha p_avg parameter as shown in equation 138d in
47              IEEE 802.16e-2005 indicates the multiple of 1/16. For
48              example '0' means 1/16, 15 means 16/16. "
49         REFERENCE
50             "Subclause 11.3.1, Table 353, and 8.4.10.3.2.1 in IEEE
51              Std 802.16e-2005"
52         ::= { wmanIf2eBsOfdmaUplinkChannelEntry 23 }
53
54
55
56
57     wmanIf2eBsOfdmaCqichTxPwrRepThreshold OBJECT-TYPE
58         SYNTAX      INTEGER (0..15)
59         UNITS        "dB"
60         MAX-ACCESS   read-write
61         STATUS        current
62         DESCRIPTION
63             "Tx power report threshold.
64              wmanIf2eBsOfdmaTxPwrRepThreshold = 0b1111 means infinite."
65

```

It shall be used when CQICH is allocated to the SS."
 REFERENCE
 "Subclause 11.3.1, Table 353, and 8.4.10.3.2.1 in IEEE
 Std 802.16e-2005"
 ::= { wmanIf2eBsOfdmaUplinkChannelEntry 24 }

wmanIf2eBsOfdmaCqichTprPower OBJECT-TYPE

SYNTAX INTEGER (0..15)
 UNITS "dB"
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "Tx power report interval = $2^{\text{wmanIf2eBsOfdmaTprPower}}$.
 The unit of Tx power report interval is frame.
 wmanIf2eBsOfdmaTprPower = 0b1111 means infinite.
 It shall be used when CQICH is allocated to the SS."
 REFERENCE
 "Subclause 11.3.1, Table 353, and 8.4.10.3.2.1 in IEEE
 Std 802.16e-2005"
 ::= { wmanIf2eBsOfdmaUplinkChannelEntry 25 }

wmanIf2eBsOfdmaCqichAlphaPavg OBJECT-TYPE

SYNTAX INTEGER (0..15)
 UNITS "dB"
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "Alpha p_avg parameter as shown in equation 138d in
 IEEE 802.16e-2005 indicates the multiple of 1/16. For
 example '0' means 1/16, 15 means 16/16. It shall be
 used when CQICH is allocated to the SS."
 REFERENCE
 "Subclause 11.3.1, Table 353, and 8.4.10.3.2.1 in IEEE
 Std 802.16e-2005"
 ::= { wmanIf2eBsOfdmaUplinkChannelEntry 26 }

wmanIf2eBsOfdmaNormalizedCnChSounding OBJECT-TYPE

SYNTAX INTEGER
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "Signed integer for the required C/N (dB) for Channel
 Sounding."
 REFERENCE
 "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
 ::= { wmanIf2eBsOfdmaUplinkChannelEntry 27 }

wmanIf2eBsOfdmaInitialRngBackoffStart OBJECT-TYPE

SYNTAX INTEGER (0..15)
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "Initial backoff window size for initial ranging
 contention, expressed as a power of 2."

```

1      REFERENCE
2          "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
3      ::= { wmanIf2eBsOfdmaUplinkChannelEntry 28 }
4
5
6      wmanIf2eBsOfdmaInitialRngBackoffEnd OBJECT-TYPE
7          SYNTAX      INTEGER (0..15)
8          MAX-ACCESS   read-write
9          STATUS       current
10         DESCRIPTION
11             "Final backoff window size for initial ranging
12              contention, expressed as a power of 2."
13         REFERENCE
14             "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
15         ::= { wmanIf2eBsOfdmaUplinkChannelEntry 29 }
16
17
18
19         wmanIf2eBsOfdmaBwRequestBackoffStart OBJECT-TYPE
20             SYNTAX      INTEGER (0..15)
21             MAX-ACCESS   read-write
22             STATUS       current
23             DESCRIPTION
24                 "Initial backoff window size for contention BW requests,
25                  expressed as a power of 2."
26             REFERENCE
27                 "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
28             ::= { wmanIf2eBsOfdmaUplinkChannelEntry 30 }
29
30
31
32         wmanIf2eBsOfdmaBwRequestBackoffEnd OBJECT-TYPE
33             SYNTAX      INTEGER (0..15)
34             MAX-ACCESS   read-write
35             STATUS       current
36             DESCRIPTION
37                 "Final backoff window size for contention BW requests,
38                  expressed as a power of 2."
39             REFERENCE
40                 "Subclause 11.3.1, Table 353, in IEEE Std 802.16e-2005"
41             ::= { wmanIf2eBsOfdmaUplinkChannelEntry 31 }
42
43
44
45
46         --
47         -- SS object group - containing tables and objects to be implemented in
48         -- the Subscriber station
49
50         --
51         -- wmanIf2SsCps contain the SS Common Part Sublayer objects
52         --
53         wmanIf2SsCps OBJECT IDENTIFIER ::= { wmanIf2SsObjects 1 }
54
55
56         --
57         -- wmanIf2SsConfigurationTable contains global parameters for SS
58         --
59
60         wmanIf2SsConfigurationTable OBJECT-TYPE
61             SYNTAX      SEQUENCE OF WmanIf2SsConfigurationEntry
62             MAX-ACCESS   not-accessible
63             STATUS       current
64             DESCRIPTION
65

```



```

1         "This table contains one row for the SS system
2         parameters."
3     REFERENCE
4         "Subclause 10.1 in IEEE Std 802.16-2004"
5     ::= { wmanIf2SsCps 1 }
6
7
8     wmanIf2SsConfigurationEntry OBJECT-TYPE
9         SYNTAX      WmanIf2SsConfigurationEntry
10        MAX-ACCESS  not-accessible
11        STATUS      current
12        DESCRIPTION
13            "This table is indexed by ifIndex."
14        INDEX { ifIndex }
15        ::= { wmanIf2SsConfigurationTable 1 }
16
17
18
19    WmanIf2SsConfigurationEntry ::= SEQUENCE {
20        wmanIf2SsLostDLMapInterval      INTEGER,
21        wmanIf2SsLostULMapInterval      INTEGER,
22        wmanIf2SsContentionRangRetries  INTEGER,
23        wmanIf2SsRequestRetries        INTEGER,
24        wmanIf2SsRegRequestRetries      INTEGER,
25        wmanIf2SsTftpBackoffStart      INTEGER,
26        wmanIf2SsTftpBackoffEnd        INTEGER,
27        wmanIf2SsTftpRequestRetries    INTEGER,
28        wmanIf2SsTftpDownloadRetries   INTEGER,
29        wmanIf2SsTftpWait               INTEGER,
30        wmanIf2SsToDRetries             INTEGER,
31        wmanIf2SsToDRetryPeriod         INTEGER,
32        wmanIf2SsT1Timeout              INTEGER,
33        wmanIf2SsT2Timeout              INTEGER,
34        wmanIf2SsT3Timeout              INTEGER,
35        wmanIf2SsT4Timeout              INTEGER,
36        wmanIf2SsT6Timeout              INTEGER,
37        wmanIf2SsT12Timeout             INTEGER,
38        wmanIf2SsT14Timeout             INTEGER,
39        wmanIf2SsT16Timeout             INTEGER,
40        wmanIf2SsT18Timeout             INTEGER,
41        wmanIf2SsT19Timeout             INTEGER,
42        wmanIf2SsT20Timeout             INTEGER,
43        wmanIf2SsT21Timeout             INTEGER,
44        wmanIf2SsSBCRequestRetries     INTEGER,
45        wmanIf2SsTftpCpltRetries        INTEGER,
46        wmanIf2SsT26Timeout             INTEGER,
47        wmanIf2SsDLManagProcTime        INTEGER}
48
49
50
51    wmanIf2SsLostDLMapInterval OBJECT-TYPE
52        SYNTAX      INTEGER (0..600)
53        UNITS       "milliseconds"
54        MAX-ACCESS  read-write
55        STATUS      current
56        DESCRIPTION
57            "Time since last received DL-MAP message before downlink
58            synchronization is considered lost in ms."
59        ::= { wmanIf2SsConfigurationEntry 1 }
60
61
62
63
64
65

```

```

1
2 wmanIf2SsLostULMapInterval OBJECT-TYPE
3     SYNTAX      INTEGER (0..600)
4     UNITS       "milliseconds"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "Time since last received UL-MAP message before uplink
9         synchronization is considered lost in ms."
10    ::= { wmanIf2SsConfigurationEntry 2 }
11
12
13
14 wmanIf2SsContentionRangRetries OBJECT-TYPE
15     SYNTAX      INTEGER (16..65535)
16     MAX-ACCESS  read-write
17     STATUS      current
18     DESCRIPTION
19         "Number of retries on contention Ranging Requests."
20    ::= { wmanIf2SsConfigurationEntry 3 }
21
22
23
24 wmanIf2SsRequestRetries OBJECT-TYPE
25     SYNTAX      INTEGER (16..65535)
26     MAX-ACCESS  read-write
27     STATUS      current
28     DESCRIPTION
29         "Number of retries on bandwidth allocation requests."
30    ::= { wmanIf2SsConfigurationEntry 4 }
31
32
33
34 wmanIf2SsRegRequestRetries OBJECT-TYPE
35     SYNTAX      INTEGER (3..65535)
36     MAX-ACCESS  read-write
37     STATUS      current
38     DESCRIPTION
39         "Number of retries on registration requests."
40    ::= { wmanIf2SsConfigurationEntry 5 }
41
42
43
44 wmanIf2SsTftpBackoffStart OBJECT-TYPE
45     SYNTAX      INTEGER (1..65535)
46     UNITS       "seconds"
47     MAX-ACCESS  read-write
48     STATUS      current
49     DESCRIPTION
50         "Initial value for TFTP backoff in second."
51    ::= { wmanIf2SsConfigurationEntry 6 }
52
53
54
55 wmanIf2SsTftpBackoffEnd OBJECT-TYPE
56     SYNTAX      INTEGER (16..65535)
57     UNITS       "seconds"
58     MAX-ACCESS  read-write
59     STATUS      current
60     DESCRIPTION
61         "Last value for TFTP backoff in second."
62    ::= { wmanIf2SsConfigurationEntry 7 }
63
64
65 wmanIf2SsTftpRequestRetries OBJECT-TYPE

```

```

1      SYNTAX      INTEGER (16..65535)
2      MAX-ACCESS  read-write
3      STATUS      current
4      DESCRIPTION
5          "Number of retries on TFTP request."
6      ::= { wmanIf2SsConfigurationEntry 8 }
7
8
9
10     wmanIf2SsTftpDownloadRetries OBJECT-TYPE
11         SYNTAX      INTEGER (3..65535)
12         MAX-ACCESS  read-write
13         STATUS      current
14         DESCRIPTION
15             "Number of retries on entire TFTP downloads."
16         ::= { wmanIf2SsConfigurationEntry 9 }
17
18
19
20     wmanIf2SsTftpWait OBJECT-TYPE
21         SYNTAX      INTEGER (2..65535)
22         UNITS        "minutes"
23         MAX-ACCESS  read-write
24         STATUS      current
25         DESCRIPTION
26             "The duration between two consecutive Transfer
27              operational parameters (TFTP) retries in min."
28         ::= { wmanIf2SsConfigurationEntry 10 }
29
30
31
32     wmanIf2SsToDRetries OBJECT-TYPE
33         SYNTAX      INTEGER (3..65535)
34         MAX-ACCESS  read-write
35         STATUS      current
36         DESCRIPTION
37             "Number of Retries to establish the Time of Day."
38         ::= { wmanIf2SsConfigurationEntry 11 }
39
40
41
42     wmanIf2SsToDRetryPeriod OBJECT-TYPE
43         SYNTAX      INTEGER (5..65535)
44         UNITS        "minutes"
45         MAX-ACCESS  read-write
46         STATUS      current
47         DESCRIPTION
48             "The retry period to re-establish the Time of Day, as
49              describe in the network entry procedure."
50         ::= { wmanIf2SsConfigurationEntry 12 }
51
52
53
54     wmanIf2SsT1Timeout OBJECT-TYPE
55         SYNTAX      INTEGER (0..50000)
56         UNITS        "milliseconds"
57         MAX-ACCESS  read-write
58         STATUS      current
59         DESCRIPTION
60             "Wait for DCD timeout in ms."
61         ::= { wmanIf2SsConfigurationEntry 13 }
62
63
64
65     wmanIf2SsT2Timeout OBJECT-TYPE
66         SYNTAX      INTEGER (0..10000)

```

```

1      UNITS      "milliseconds"
2      MAX-ACCESS read-write
3      STATUS     current
4      DESCRIPTION
5          "Wait for broadcast ranging timeout in ms."
6      ::= { wmanIf2SsConfigurationEntry 14 }
7
8
9
10     wmanIf2SsT3Timeout OBJECT-TYPE
11         SYNTAX      INTEGER (0..200)
12         UNITS       "milliseconds"
13         MAX-ACCESS  read-write
14         STATUS      current
15         DESCRIPTION
16             "Ranging Response reception timeout following the
17              transmission of a Ranging Request in ms."
18         ::= { wmanIf2SsConfigurationEntry 15 }
19
20
21
22     wmanIf2SsT4Timeout OBJECT-TYPE
23         SYNTAX      INTEGER (30..35)
24         UNITS       "seconds"
25         MAX-ACCESS  read-write
26         STATUS      current
27         DESCRIPTION
28             "Wait for unicast ranging opportunity. If the pending until
29              complete field was used earlier by this SS, then the value
30              of that field shall be added to this interval in second."
31         ::= { wmanIf2SsConfigurationEntry 16 }
32
33
34
35     wmanIf2SsT6Timeout OBJECT-TYPE
36         SYNTAX      INTEGER (0..3000)
37         UNITS       "milliseconds"
38         MAX-ACCESS  read-write
39         STATUS      current
40         DESCRIPTION
41             "Wait for registration response in ms."
42         ::= { wmanIf2SsConfigurationEntry 17 }
43
44
45
46     wmanIf2SsT12Timeout OBJECT-TYPE
47         SYNTAX      INTEGER (0..50000)
48         UNITS       "milliseconds"
49         MAX-ACCESS  read-write
50         STATUS      current
51         DESCRIPTION
52             "Wait for UCD descriptor in ms."
53         ::= { wmanIf2SsConfigurationEntry 18 }
54
55
56
57     wmanIf2SsT14Timeout OBJECT-TYPE
58         SYNTAX      INTEGER (0..200)
59         UNITS       "milliseconds"
60         MAX-ACCESS  read-write
61         STATUS      current
62         DESCRIPTION
63             "Wait for DSX-RVD Timeout in ms."
64         ::= { wmanIf2SsConfigurationEntry 19 }
65

```

```

1
2 wmanIf2SsT16Timeout OBJECT-TYPE
3     SYNTAX      INTEGER (10..65535)
4     UNITS       "milliseconds"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "wait for bandwidth request grant in ms."
9         ::= { wmanIf2SsConfigurationEntry 20 }
10
11
12
13 wmanIf2SsT18Timeout OBJECT-TYPE
14     SYNTAX      INTEGER (0..65535)
15     UNITS       "milliseconds"
16     MAX-ACCESS  read-write
17     STATUS      current
18     DESCRIPTION
19         "wait for SBC-RSP timeout in ms."
20         ::= { wmanIf2SsConfigurationEntry 21 }
21
22
23
24 wmanIf2SsT19Timeout OBJECT-TYPE
25     SYNTAX      INTEGER (0..1048575)
26     UNITS       "milliseconds"
27     MAX-ACCESS  read-write
28     STATUS      current
29     DESCRIPTION
30         "Time DL-channel remains unusable in ms."
31         ::= { wmanIf2SsConfigurationEntry 22 }
32
33
34
35 wmanIf2SsT20Timeout OBJECT-TYPE
36     SYNTAX      INTEGER (0..65535)
37     UNITS       "milliseconds"
38     MAX-ACCESS  read-write
39     STATUS      current
40     DESCRIPTION
41         "Time SS searches for preambles on a given channel in ms."
42         ::= { wmanIf2SsConfigurationEntry 23 }
43
44
45
46 wmanIf2SsT21Timeout OBJECT-TYPE
47     SYNTAX      INTEGER (0..10000)
48     UNITS       "milliseconds"
49     MAX-ACCESS  read-write
50     STATUS      current
51     DESCRIPTION
52         "Time SS searches for DL-MAP on a given channel in ms."
53         ::= { wmanIf2SsConfigurationEntry 24 }
54
55
56
57 wmanIf2SsSBCRequestRetries OBJECT-TYPE
58     SYNTAX      INTEGER (3..16)
59     MAX-ACCESS  read-write
60     STATUS      current
61     DESCRIPTION
62         "Number of retries on SBC Request."
63         ::= { wmanIf2SsConfigurationEntry 25 }
64
65

```

```

1  wmanIf2SsTftpCpltRetries OBJECT-TYPE
2      SYNTAX      INTEGER (3..16)
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "Number of retries on TFTP-CPLT."
7      ::= { wmanIf2SsConfigurationEntry 26 }
8
9
10 wmanIf2SsT26Timeout OBJECT-TYPE
11     SYNTAX      INTEGER (10..200)
12     UNITS        "milliseconds"
13     MAX-ACCESS  read-write
14     STATUS      current
15     DESCRIPTION
16         "Wait for TFTP-RSP in ms."
17     ::= { wmanIf2SsConfigurationEntry 27 }
18
19
20
21 wmanIf2SsDLManagProcTime OBJECT-TYPE
22     SYNTAX      INTEGER (0..200)
23     UNITS        "micro seconds"
24     MAX-ACCESS  read-write
25     STATUS      current
26     DESCRIPTION
27         "Max. time between reception of Fast Power Control
28         management message and compliance to its instructions
29         by SS in us."
30     ::= { wmanIf2SsConfigurationEntry 28 }
31
32
33
34 --
35 -- Subscriber Channel Measurement Table
36 --
37
38 wmanIf2SsChannelMeasurementTable OBJECT-TYPE
39     SYNTAX      SEQUENCE OF WmanIf2SsChannelMeasurementEntry
40     MAX-ACCESS  not-accessible
41     STATUS      current
42     DESCRIPTION
43         "This table contains downlink channel measurement
44         information for each SS."
45     REFERENCE
46         "6.3.2.3.33 in IEEE Std 802.16-2004"
47     ::= { wmanIf2SsCps 2 }
48
49
50
51 wmanIf2SsChannelMeasurementEntry OBJECT-TYPE
52     SYNTAX      WmanIf2SsChannelMeasurementEntry
53     MAX-ACCESS  not-accessible
54     STATUS      current
55     DESCRIPTION
56         "Each entry in the table contains RSSI and CINR
57         signal quality measurement taken from the SS. The primary
58         index is the ifIndex pointing to SS.
59         wmanIf2CmnHistogramIndex is the index to histogram
60         samples. Since there is no time stamp in the table,
61         wmanIf2SsHistogramIndex should be increased monotonically,
62         and wraps around when it reaches the limit."
63
64
65

```

```

1           When the measurement entry for a SS reaches the limit,
2           the oldest entry shall be deleted as the new entry is
3           added to the table."
4
5       INDEX          { ifIndex, wmanIf2SsHistogramIndex }
6       ::= { wmanIf2SsChannelMeasurementTable 1 }
7
8
9       WmanIf2SsChannelMeasurementEntry ::= SEQUENCE {
10          wmanIf2SsHistogramIndex          Unsigned32,
11          wmanIf2SsChannelNumber           WmanIf2ChannelNumber,
12          wmanIf2SsStartFrame              INTEGER,
13          wmanIf2SsDuration                 INTEGER,
14          wmanIf2SsBasicReport              BITS,
15          wmanIf2SsMeanCinrReport           INTEGER,
16          wmanIf2SsStdDeviationCinrReport   INTEGER,
17          wmanIf2SsMeanRssiReport           INTEGER,
18          wmanIf2SsStdDeviationRssiReport   INTEGER}
19
20
21       wmanIf2SsHistogramIndex OBJECT-TYPE
22       SYNTAX          Unsigned32 (1 .. 4294967295)
23       MAX-ACCESS      not-accessible
24       STATUS          current
25       DESCRIPTION
26       "wmanIf2SsHistogramIndex identifies the histogram samples
27       in the table for each subscriber station."
28       ::= { wmanIf2SsChannelMeasurementEntry 1 }
29
30
31       wmanIf2SsChannelNumber OBJECT-TYPE
32       SYNTAX          WmanIf2ChannelNumber
33       MAX-ACCESS      read-only
34       STATUS          current
35       DESCRIPTION
36       "Physical channel number to be reported on."
37       REFERENCE
38       "Subclause 8.5.1 in IEEE Std 802.16-2004"
39       ::= { wmanIf2SsChannelMeasurementEntry 2 }
40
41
42       wmanIf2SsStartFrame OBJECT-TYPE
43       SYNTAX          INTEGER (0 .. 65535)
44       MAX-ACCESS      read-only
45       STATUS          current
46       DESCRIPTION
47       "Frame number in which measurement for this channel
48       started."
49       REFERENCE
50       "Subclause 11.12 in IEEE Std 802.16-2004"
51       ::= { wmanIf2SsChannelMeasurementEntry 3 }
52
53
54       wmanIf2SsDuration OBJECT-TYPE
55       SYNTAX          INTEGER (0..16777215)
56       MAX-ACCESS      read-only
57       STATUS          current
58       DESCRIPTION
59       "Cumulative measurement duration on the channel in
60       multiples of Ts. For any value exceeding 0xFFFFF,
61

```

```

1         report 0xFFFFFFFF."
2     REFERENCE
3         "Subclause 11.12 in IEEE Std 802.16-2004"
4     ::= { wmanIf2SsChannelMeasurementEntry 4 }
5
6
7 wmanIf2SsBasicReport OBJECT-TYPE
8     SYNTAX      BITS {wirelessHuman(0),
9                  unknownTransmission(1),
10                 primaryUser(2),
11                 channelNotMeasured(3)}
12
13     MAX-ACCESS  read-only
14     STATUS      current
15     DESCRIPTION
16         "Bit #0: WirelessHUMAN detected on the channel
17         Bit #1: Unknown transmissions detected on the channel
18         Bit #2: Primary User detected on the channel
19         Bit #3: Unmeasured. Channel not measured"
20
21     REFERENCE
22         "Subclause 11.12 in IEEE Std 802.16-2004"
23     ::= { wmanIf2SsChannelMeasurementEntry 5 }
24
25
26 wmanIf2SsMeanCinrReport OBJECT-TYPE
27     SYNTAX      INTEGER (0 .. 41)
28     UNITS       "dB"
29     MAX-ACCESS  read-only
30     STATUS      current
31     DESCRIPTION
32         "Mean CINR report."
33
34     REFERENCE
35         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
36     ::= { wmanIf2SsChannelMeasurementEntry 6 }
37
38
39 wmanIf2SsStdDeviationCinrReport OBJECT-TYPE
40     SYNTAX      INTEGER (0 .. 41)
41     UNITS       "dB"
42     MAX-ACCESS  read-only
43     STATUS      current
44     DESCRIPTION
45         "Standard deviation CINR report."
46
47     REFERENCE
48         "Subclause 8.2.2 and Subclause 8.3.9 in IEEE Std 802.16-2004"
49     ::= { wmanIf2SsChannelMeasurementEntry 7 }
50
51
52 wmanIf2SsMeanRssiReport OBJECT-TYPE
53     SYNTAX      INTEGER (0 .. 83)
54     UNITS       "dBm"
55     MAX-ACCESS  read-only
56     STATUS      current
57     DESCRIPTION
58         "Mean RSSI report."
59
60     REFERENCE
61         "Subclause 8.2.2 and Subclause 8.3.9 in IEEE Std 802.16-2004"
62     ::= { wmanIf2SsChannelMeasurementEntry 8 }
63
64
65

```



```

1  wmanIf2SsStdDeviationRssiReport OBJECT-TYPE
2      SYNTAX      INTEGER (0 .. 83)
3      UNITS       "dB"
4      MAX-ACCESS  read-only
5      STATUS      current
6      DESCRIPTION
7          "Standard deviation RSSI report."
8      REFERENCE
9          "Subclause 8.2.2 and Subclause 8.3.9 in IEEE Std 802.16-2004"
10     ::= { wmanIf2SsChannelMeasurementEntry 9 }
11
12  --
13  -- Subscriber station PKM group
14  -- wmanIf2SsPkmObjects contain the Subscriber Station Privacy Sublayer
15  -- objects
16  --
17  wmanIf2SsPkmObjects OBJECT IDENTIFIER ::= { wmanIf2SsObjects 2 }
18
19  --
20  -- Table wmanIf2SsPkmAuthTable
21  --
22  wmanIf2SsPkmAuthTable OBJECT-TYPE
23      SYNTAX      SEQUENCE OF WmanIf2SsPkmAuthEntry
24      MAX-ACCESS  not-accessible
25      STATUS      current
26      DESCRIPTION
27          "This table describes the PKM attributes related
28           to the authorization for each SS wireless interface."
29      ::= { wmanIf2SsPkmObjects 1 }
30
31  wmanIf2SsPkmAuthEntry OBJECT-TYPE
32      SYNTAX      WmanIf2SsPkmAuthEntry
33      MAX-ACCESS  not-accessible
34      STATUS      current
35      DESCRIPTION
36          "Each entry contains objects describing attributes of one
37           SS wireless interface."
38      INDEX      { ifIndex }
39      ::= { wmanIf2SsPkmAuthTable 1 }
40
41  WmanIf2SsPkmAuthEntry ::= SEQUENCE {
42      wmanIf2SsPkmAuthState          INTEGER,
43      wmanIf2SsPkmAuthKeySequenceNumber Integer32,
44      wmanIf2SsPkmAuthExpiresOld     DateAndTime,
45      wmanIf2SsPkmAuthExpiresNew     DateAndTime,
46      wmanIf2SsPkmAuthReset          TruthValue,
47      wmanIf2SsPkmAuthentInfos        Counter32,
48      wmanIf2SsPkmAuthRequests        Counter32,
49      wmanIf2SsPkmAuthReplies         Counter32,
50      wmanIf2SsPkmAuthRejects         Counter32,
51      wmanIf2SsPkmAuthInvalids        Counter32,
52      wmanIf2SsPkmAuthRejectErrorCode INTEGER,
53      wmanIf2SsPkmAuthRejectErrorString SnmpAdminString,
54      wmanIf2SsPkmAuthInvalidErrorCode INTEGER,

```

```

1      wmanIf2SsPkmAuthInvalidErrorString      SnmpAdminString,
2      wmanIf2SsPkmAuthGraceTime                Integer32,
3      wmanIf2SsPkmTekGraceTime                 Integer32,
4      wmanIf2SsPkmAuthWaitTimeout              Integer32,
5      wmanIf2SsPkmReauthWaitTimeout            Integer32,
6      wmanIf2SsPkmOpWaitTimeout                Integer32,
7      wmanIf2SsPkmRekeyWaitTimeout             Integer32,
8      wmanIf2SsPkmAuthRejectWaitTimeout        Integer32}
9
10
11
12  wmanIf2SsPkmAuthState OBJECT-TYPE
13      SYNTAX          INTEGER {start(1),
14                          authWait(2),
15                          authorized(3),
16                          reauthWait(4),
17                          authRejectWait(5),
18                          silent(6)}
19
20      MAX-ACCESS      read-only
21      STATUS          current
22      DESCRIPTION
23          "The value of this object is the state of the SS
24          authorization FSM. The start state indicates that FSM is
25          in its initial state."
26      ::= { wmanIf2SsPkmAuthEntry 1 }
27
28
29
30  wmanIf2SsPkmAuthKeySequenceNumber OBJECT-TYPE
31      SYNTAX          Integer32 (0..15)
32      MAX-ACCESS      read-only
33      STATUS          current
34      DESCRIPTION
35          "The value of this object is the most recent authorization
36          key sequence number for this FSM."
37      ::= { wmanIf2SsPkmAuthEntry 2 }
38
39
40
41  wmanIf2SsPkmAuthExpiresOld OBJECT-TYPE
42      SYNTAX          DateAndTime
43      MAX-ACCESS      read-only
44      STATUS          current
45      DESCRIPTION
46          "The value of this object is the actual clock time for
47          expiration of the immediate predecessor of the most recent
48          authorization key for this FSM. If this FSM has only one
49          authorization key, then the value is the time of activation
50          of this FSM."
51      ::= { wmanIf2SsPkmAuthEntry 3 }
52
53
54
55  wmanIf2SsPkmAuthExpiresNew OBJECT-TYPE
56      SYNTAX          DateAndTime
57      MAX-ACCESS      read-only
58      STATUS          current
59      DESCRIPTION
60          "The value of this object is the actual clock time for
61          expiration of the most recent authorization key for this
62          FSM."
63      ::= { wmanIf2SsPkmAuthEntry 4 }
64
65

```

```

1
2 wmanIf2SsPkmAuthReset OBJECT-TYPE
3     SYNTAX      TruthValue
4     MAX-ACCESS  read-write
5     STATUS      current
6     DESCRIPTION
7         "Setting this object to TRUE generates a Reauthorize event
8         in the authorization FSM. Reading this object always
9         returns FALSE."
10
11     ::= { wmanIf2SsPkmAuthEntry 5 }
12
13
14 wmanIf2SsPkmAuthentInfos OBJECT-TYPE
15     SYNTAX      Counter32
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "The value of this object is the count of times the SS has
20         transmitted an Authentication Information message."
21
22     ::= { wmanIf2SsPkmAuthEntry 6 }
23
24
25 wmanIf2SsPkmAuthRequests OBJECT-TYPE
26     SYNTAX      Counter32
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30         "The value of this object is the count of times the SS has
31         transmitted an Authorization Request message."
32
33     ::= { wmanIf2SsPkmAuthEntry 7 }
34
35
36 wmanIf2SsPkmAuthReplies OBJECT-TYPE
37     SYNTAX      Counter32
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41         "The value of this object is the count of times the SS has
42         received an Authorization Reply message."
43
44     ::= { wmanIf2SsPkmAuthEntry 8 }
45
46
47 wmanIf2SsPkmAuthRejects OBJECT-TYPE
48     SYNTAX      Counter32
49     MAX-ACCESS  read-only
50     STATUS      current
51     DESCRIPTION
52         "The value of this object is the count of times the SS has
53         received an Authorization Reject message."
54
55     ::= { wmanIf2SsPkmAuthEntry 9 }
56
57
58 wmanIf2SsPkmAuthInvalids OBJECT-TYPE
59     SYNTAX      Counter32
60     MAX-ACCESS  read-only
61     STATUS      current
62     DESCRIPTION
63         "The value of this object is the count of times the SS has
64         received an Authorization Invalid message."
65

```

```

1      ::= { wmanIf2SsPkmAuthEntry 10 }
2
3
4  wmanIf2SsPkmAuthRejectErrorCode OBJECT-TYPE
5      SYNTAX      INTEGER {none(1),
6                      unknown(2),
7                      unauthorizedSs(3),
8                      unauthorizedSaid(4),
9                      permanentAuthorizationFailure(8),
10                     timeOfDayNotAcquired(11)}
11
12      MAX-ACCESS  read-only
13      STATUS      current
14      DESCRIPTION
15          "The value of this object is the enumerated description of
16          the Error-Code in most recent Authorization Reject message
17          received by the SS. This has value unknown(2) if the last
18          Error-Code value was 0, and none(1) if no Authorization
19          Reject message has been received since reboot."
20
21      ::= { wmanIf2SsPkmAuthEntry 11 }
22
23
24  wmanIf2SsPkmAuthRejectErrorString OBJECT-TYPE
25      SYNTAX      SnmpAdminString (SIZE (0..128))
26      MAX-ACCESS  read-only
27      STATUS      current
28      DESCRIPTION
29          "The value of this object is the Display-String in most
30          recent Authorization Reject message received by the SS.
31          This is a zero length string if no Authorization Reject
32          message has been received since reboot."
33
34      ::= { wmanIf2SsPkmAuthEntry 12 }
35
36
37  wmanIf2SsPkmAuthInvalidErrorCode OBJECT-TYPE
38      SYNTAX      INTEGER {none(1),
39                      unknown(2),
40                      unauthorizedSs(3),
41                      unsolicited(5),
42                      invalidKeySequence(6),
43                      keyRequestAuthenticationFailure(7)}
44
45      MAX-ACCESS  read-only
46      STATUS      current
47      DESCRIPTION
48          "The value of this object is the enumerated description of
49          the Error-Code in most recent Authorization Invalid message
50          received by the SS. This has value unknown(2) if the last
51          Error-Code value was 0, and none(1) if no Authorization
52          Invalid message has been received since reboot."
53
54      ::= { wmanIf2SsPkmAuthEntry 13 }
55
56
57  wmanIf2SsPkmAuthInvalidErrorString OBJECT-TYPE
58      SYNTAX      SnmpAdminString (SIZE (0..128))
59      MAX-ACCESS  read-only
60      STATUS      current
61      DESCRIPTION
62          "The value of this object is the Display-String in most
63          recent Authorization Invalid message received by the SS."
64
65

```

```

1         This is a zero length string if no Authorization Invalid
2         message has been received since reboot."
3
4     ::= { wmanIf2SsPkmAuthEntry 14 }

```

wmanIf2SsPkmAuthGraceTime OBJECT-TYPE

```

7     SYNTAX      Integer32 (300..3024000)
8     UNITS       "seconds"
9     MAX-ACCESS  read-only
10    STATUS      current
11    DESCRIPTION
12
13        "The value of this object is the grace time for an
14        authorization key. A SS is expected to start trying to get
15        a new authorization key beginning AuthGraceTime seconds
16        before the authorization key actually expires."
17
18    REFERENCE
19
20        "Table 343 in IEEE Std 802.16-2004"
21
22    DEFVAL      { 600 }
23    ::= { wmanIf2SsPkmAuthEntry 15 }

```

wmanIf2SsPkmTekGraceTime OBJECT-TYPE

```

25    SYNTAX      Integer32 (300..3024000)
26    UNITS       "seconds"
27    MAX-ACCESS  read-only
28    STATUS      current
29    DESCRIPTION
30
31        "The value of this object is the grace time for the TEK in
32        seconds. The SS is expected to start trying to acquire a
33        new TEK beginning TEK GraceTime seconds before the
34        expiration of the most recent TEK."
35
36    REFERENCE
37
38        "Table 343 in IEEE Std 802.16-2004"
39
40    DEFVAL      { 3600 }
41    ::= { wmanIf2SsPkmAuthEntry 16 }

```

wmanIf2SsPkmAuthWaitTimeout OBJECT-TYPE

```

43    SYNTAX      Integer32 (2..30)
44    UNITS       "seconds"
45    MAX-ACCESS  read-only
46    STATUS      current
47    DESCRIPTION
48
49        "The value of this object is the Authorize Wait Timeout."
50
51    REFERENCE
52
53        "Table 343 in IEEE Std 802.16-2004"
54
55    DEFVAL      { 10 }
56    ::= { wmanIf2SsPkmAuthEntry 17 }

```

wmanIf2SsPkmReauthWaitTimeout OBJECT-TYPE

```

57    SYNTAX      Integer32 (2..30)
58    UNITS       "seconds"
59    MAX-ACCESS  read-only
60    STATUS      current
61    DESCRIPTION
62
63        "The value of this object is the Reauthorize Wait Timeout
64        in seconds."
65

```

```

1      REFERENCE
2          "Table 343 in IEEE Std 802.16-2004"
3      DEFVAL      { 10 }
4      ::= { wmanIf2SsPkmAuthEntry 18 }
5
6
7      wmanIf2SsPkmOpWaitTimeout OBJECT-TYPE
8          SYNTAX      Integer32 (1..10)
9          UNITS        "seconds"
10         MAX-ACCESS   read-only
11         STATUS        current
12         DESCRIPTION
13             "The value of this object is the Operational Wait Timeout
14             in seconds."
15         REFERENCE
16             "Table 343 in IEEE Std 802.16-2004"
17         DEFVAL      { 1 }
18         ::= { wmanIf2SsPkmAuthEntry 19 }
19
20
21
22
23         wmanIf2SsPkmRekeyWaitTimeout OBJECT-TYPE
24             SYNTAX      Integer32 (1..10)
25             UNITS        "seconds"
26             MAX-ACCESS   read-only
27             STATUS        current
28             DESCRIPTION
29                 "The value of this object is the Rekey Wait Timeout in
30                 seconds."
31             REFERENCE
32                 "Table 343 in IEEE Std 802.16-2004"
33             DEFVAL      { 1 }
34             ::= { wmanIf2SsPkmAuthEntry 20 }
35
36
37
38
39         wmanIf2SsPkmAuthRejectWaitTimeout OBJECT-TYPE
40             SYNTAX      Integer32 (10..600)
41             UNITS        "seconds"
42             MAX-ACCESS   read-only
43             STATUS        current
44             DESCRIPTION
45                 "The value of this object is the Authorization Reject Wait
46                 Timeout in seconds."
47             REFERENCE
48                 "Table 343 in IEEE Std 802.16-2004"
49             DEFVAL      { 60 }
50             ::= { wmanIf2SsPkmAuthEntry 21 }
51
52
53
54         --
55         -- Table wmanIf2SsPkmTekTable
56         --
57         wmanIf2SsPkmTekTable OBJECT-TYPE
58             SYNTAX      SEQUENCE OF WmanIf2SsPkmTekEntry
59             MAX-ACCESS   not-accessible
60             STATUS        current
61             DESCRIPTION
62                 "This table describes the attributes of each SS Traffic
63                 Encryption Key (TEK) association. The SS maintains (no more
64                 Encryption Key (TEK) association. The SS maintains (no more
65                 Encryption Key (TEK) association. The SS maintains (no more

```

```

1         than) one TEK association per SAID per SS wireless
2         interface."
3         ::= { wmanIf2SsPkmObjects 2 }
4
5
6 wmanIf2SsPkmTekEntry OBJECT-TYPE
7     SYNTAX      WmanIf2SsPkmTekEntry
8     MAX-ACCESS  not-accessible
9     STATUS      current
10    DESCRIPTION
11        "Each entry contains objects describing the TEK association
12         attributes of one SAID. The SS MUST create one entry per
13         SAID, regardless of whether the SAID was obtained from a
14         Registration Response message, from an Authorization Reply
15         message, or from any dynamic SAID establishment
16         mechanisms."
17    INDEX        { ifIndex, wmanIf2SsPkmTekSAID }
18    ::= { wmanIf2SsPkmTekTable 1 }
19
20
21
22
23 WmanIf2SsPkmTekEntry ::= SEQUENCE {
24     wmanIf2SsPkmTekSAID                INTEGER,
25     wmanIf2SsPkmTekSAType              INTEGER,
26     wmanIf2SsPkmTekDataEncryptAlg     WmanIf2DataEncryptAlgId,
27     wmanIf2SsPkmTekDataAuthAlg       WmanIf2DataAuthAlgId,
28     wmanIf2SsPkmTekEncryptAlg        WmanIf2TekEncryptAlgId,
29     wmanIf2SsPkmTekState              INTEGER,
30     wmanIf2SsPkmTekKeySequenceNumber Integer32,
31     wmanIf2SsPkmTekExpiresOld         DateAndTime,
32     wmanIf2SsPkmTekExpiresNew        DateAndTime,
33     wmanIf2SsPkmTekKeyRequests        Counter32,
34     wmanIf2SsPkmTekKeyReplies         Counter32,
35     wmanIf2SsPkmTekKeyRejects         Counter32,
36     wmanIf2SsPkmTekInvalids           Counter32,
37     wmanIf2SsPkmTekAuthPends          Counter32,
38     wmanIf2SsPkmTekKeyRejectErrorCode INTEGER,
39     wmanIf2SsPkmTekKeyRejectErrorString SnmpAdminString,
40     wmanIf2SsPkmTekInvalidErrorCode   INTEGER,
41     wmanIf2SsPkmTekInvalidErrorString SnmpAdminString}
42
43
44
45
46
47 wmanIf2SsPkmTekSAID OBJECT-TYPE
48     SYNTAX      INTEGER (0..65535)
49     MAX-ACCESS  not-accessible
50     STATUS      current
51     DESCRIPTION
52        "The value of this object is the Security Association
53         ID (SAID). "
54     ::= { wmanIf2SsPkmTekEntry 1 }
55
56
57
58 wmanIf2SsPkmTekSAType OBJECT-TYPE
59     SYNTAX      INTEGER {primarySA(0),
60                        staticSA(1),
61                        dynamicSA(2)}
62     MAX-ACCESS  read-only
63     STATUS      current
64     DESCRIPTION
65

```

```

1         "The value of this object is the type of security
2         association."
3     REFERENCE
4         "IEEE Std 802.16-2004; 11.9.18"
5     ::= { wmanIf2SsPkmTekEntry 2 }
6
7
8     wmanIf2SsPkmTekDataEncryptAlg OBJECT-TYPE
9         SYNTAX      WmanIf2DataEncryptAlgId
10        MAX-ACCESS   read-only
11        STATUS       current
12        DESCRIPTION
13            "The value of this object is the data encryption algorithm
14            being utilized."
15        REFERENCE
16            "Table 375, IEEE Std 802.16-2004"
17        ::= { wmanIf2SsPkmTekEntry 3 }
18
19
20
21     wmanIf2SsPkmTekDataAuthentAlg OBJECT-TYPE
22         SYNTAX      WmanIf2DataAuthAlgId
23        MAX-ACCESS   read-only
24        STATUS       current
25        DESCRIPTION
26            "The value of this object is the data authentication
27            algorithm being utilized."
28        REFERENCE
29            "Table 376, IEEE Std 802.16-2004"
30        ::= { wmanIf2SsPkmTekEntry 4 }
31
32
33
34
35     wmanIf2SsPkmTekEncryptAlg OBJECT-TYPE
36         SYNTAX      WmanIf2TekEncryptAlgId
37        MAX-ACCESS   read-only
38        STATUS       current
39        DESCRIPTION
40            "The value of this object is the TEK key encryption
41            algorithm for this cryptographic suite capability."
42        REFERENCE
43            "Table 377, IEEE Std 802.16-2004"
44        ::= { wmanIf2SsPkmTekEntry 5 }
45
46
47
48     wmanIf2SsPkmTekState OBJECT-TYPE
49         SYNTAX      INTEGER {start(1),
50                           opWait(2),
51                           opReauthWait(3),
52                           operational(4),
53                           rekeyWait(5),
54                           rekeyReauthWait(6)}
55        MAX-ACCESS   read-only
56        STATUS       current
57        DESCRIPTION
58            "The value of this object is the state of the indicated TEK
59            FSM. The start(1) state indicates that FSM is in its
60            initial state."
61        ::= { wmanIf2SsPkmTekEntry 6 }
62
63
64
65

```



```

1  wmanIf2SsPkmTekKeySequenceNumber OBJECT-TYPE
2      SYNTAX      Integer32 (0..3)
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "The value of this object is the most recent TEK key
7          sequence number for this TEK FSM."
8      REFERENCE
9          "IEEE Std 802.16-2004; 11.9.5"
10     ::= { wmanIf2SsPkmTekEntry 7 }
11
12  wmanIf2SsPkmTekExpiresOld OBJECT-TYPE
13      SYNTAX      DateAndTime
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "The value of this object is the actual clock time for
18          expiration of the immediate predecessor of the most recent
19          TEK for this FSM. If this FSM has only one TEK, then the
20          value is the time of activation of this FSM."
21      ::= { wmanIf2SsPkmTekEntry 8 }
22
23  wmanIf2SsPkmTekExpiresNew OBJECT-TYPE
24      SYNTAX      DateAndTime
25      MAX-ACCESS  read-only
26      STATUS      current
27      DESCRIPTION
28          "The value of this object is the actual clock time for
29          expiration of the most recent TEK for this FSM."
30      ::= { wmanIf2SsPkmTekEntry 9 }
31
32  wmanIf2SsPkmTekKeyRequests OBJECT-TYPE
33      SYNTAX      Counter32
34      MAX-ACCESS  read-only
35      STATUS      current
36      DESCRIPTION
37          "The value of this object is the count of times the SS has
38          transmitted a Key Request message."
39      ::= { wmanIf2SsPkmTekEntry 10 }
40
41  wmanIf2SsPkmTekKeyReplies OBJECT-TYPE
42      SYNTAX      Counter32
43      MAX-ACCESS  read-only
44      STATUS      current
45      DESCRIPTION
46          "The value of this object is the count of times the SS has
47          received a Key Reply message, including a message whose
48          authentication failed."
49      ::= { wmanIf2SsPkmTekEntry 11 }
50
51  wmanIf2SsPkmTekKeyRejects OBJECT-TYPE
52      SYNTAX      Counter32
53      MAX-ACCESS  read-only
54      STATUS      current

```

```

1      DESCRIPTION
2          "The value of this object is the count of times the SS has
3            received a Key Reject message, including a message whose
4            authentication failed."
5      ::= { wmanIf2SsPkmTekEntry 12 }
6
7
8      wmanIf2SsPkmTekInvalids OBJECT-TYPE
9          SYNTAX      Counter32
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "The value of this object is the count of times the SS has
14               received a TEK Invalid message, including a message whose
15               authentication failed."
16             ::= { wmanIf2SsPkmTekEntry 13 }
17
18
19
20      wmanIf2SsPkmTekAuthPends OBJECT-TYPE
21          SYNTAX      Counter32
22          MAX-ACCESS  read-only
23          STATUS      current
24          DESCRIPTION
25              "The value of this object is the count of times an
26                Authorization Pending (Auth Pend) event occurred in this
27                FSM."
28              ::= { wmanIf2SsPkmTekEntry 14 }
29
30
31
32      wmanIf2SsPkmTekKeyRejectErrorCode OBJECT-TYPE
33          SYNTAX      INTEGER {none(1),
34                               unknown(2),
35                               unauthorizedSaid(4)}
36          MAX-ACCESS  read-only
37          STATUS      current
38          DESCRIPTION
39              "The value of this object is the enumerated description of
40                the Error-Code in most recent Key Reject message received
41                by the SS. This has value unknown(2) if the last Error-Code
42                value was 0, and none(1) if no Key Reject message has been
43                received since reboot."
44              ::= { wmanIf2SsPkmTekEntry 15 }
45
46
47
48      wmanIf2SsPkmTekKeyRejectErrorString OBJECT-TYPE
49          SYNTAX      SnmpAdminString (SIZE (0..128))
50          MAX-ACCESS  read-only
51          STATUS      current
52          DESCRIPTION
53              "The value of this object is the Display-String in most
54                recent Key Reject message received by the SS. This is a
55                zero length string if no Key Reject message has been
56                received since reboot."
57              ::= { wmanIf2SsPkmTekEntry 16 }
58
59
60
61
62      wmanIf2SsPkmTekInvalidErrorCode OBJECT-TYPE
63          SYNTAX      INTEGER {none(1),
64                               unknown(2),
65

```

```

1         invalidKeySequence(6)}
2     MAX-ACCESS    read-only
3     STATUS        current
4     DESCRIPTION
5         "The value of this object is the enumerated description of
6         the Error-Code in most recent TEK Invalid message received
7         by the SS. This has value unknown(2) if the last
8         Error-Code value was 0, and none(1) if no TEK Invalid
9         message has been received since reboot."
10    ::= { wmanIf2SsPkmTekEntry 17 }
11
12
13
14 wmanIf2SsPkmTekInvalidErrorString OBJECT-TYPE
15     SYNTAX        SnmpAdminString (SIZE (0..128))
16     MAX-ACCESS    read-only
17     STATUS        current
18     DESCRIPTION
19         "The value of this object is the Display-String in most
20         recent TEK Invalid message received by the SS. This is a
21         zero length string if no TEK Invalid message has been
22         received since reboot."
23     ::= { wmanIf2SsPkmTekEntry 18 }
24
25
26
27
28 --
29 -- Table wmanIf2SsDeviceCertTable
30 --
31 wmanIf2SsDeviceCertTable OBJECT-TYPE
32     SYNTAX        SEQUENCE OF WmanIf2SsDeviceCertEntry
33     MAX-ACCESS    not-accessible
34     STATUS        current
35     DESCRIPTION
36         "This table describes the PKM device certificates for each
37         SS wireless interface."
38     ::= { wmanIf2SsPkmObjects 3 }
39
40
41
42 wmanIf2SsDeviceCertEntry OBJECT-TYPE
43     SYNTAX        WmanIf2SsDeviceCertEntry
44     MAX-ACCESS    not-accessible
45     STATUS        current
46     DESCRIPTION
47         "Each entry contains the device certificate of one SS."
48     INDEX          { ifIndex }
49     ::= { wmanIf2SsDeviceCertTable 1 }
50
51
52
53 WmanIf2SsDeviceCertEntry ::= SEQUENCE {
54     wmanIf2SsDeviceCert          OCTET STRING,
55     wmanIf2SsDeviceManufCert     OCTET STRING}
56
57
58 wmanIf2SsDeviceCert OBJECT-TYPE
59     SYNTAX        OCTET STRING (SIZE(0..65535))
60     MAX-ACCESS    read-only
61     STATUS        current
62     DESCRIPTION
63         "The X509 DER-encoded subscriber station certificate."
64     ::= { wmanIf2SsDeviceCertEntry 1 }
65

```

```

1
2 wmanIf2SsDeviceManufCert OBJECT-TYPE
3     SYNTAX      OCTET STRING (SIZE(0..65535))
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "The X509 DER-encoded manufacturer certificate which is
8         signed by the CA root authority certificate."
9     ::= { wmanIf2SsDeviceCertEntry 2 }
10
11
12
13 --
14 -- Subscriber station Notification Group
15 -- wmanIf2SsNotificationObjects contains the SS SNMP Trap objects
16 --
17 wmanIf2SsNotification OBJECT IDENTIFIER ::= { wmanIf2SsObjects 3 }
18 wmanIf2SsTrapControl OBJECT IDENTIFIER ::= { wmanIf2SsNotification 1 }
19 wmanIf2SsTrapDefinitions OBJECT IDENTIFIER ::= { wmanIf2SsNotification 2
20 }
21
22
23 -- This object groups all NOTIFICATION-TYPE objects for SS.
24 -- It is defined following RFC2758 sections 8.5 and 8.6
25 -- for the compatibility with SNMPv1.
26 wmanIf2SsTrapPrefix OBJECT IDENTIFIER ::= { wmanIf2SsTrapDefinitions 0 }
27
28
29 wmanIf2SsTrapControlRegister OBJECT-TYPE
30     SYNTAX      BITS {wmanIf2SsTlvUnknown(0),
31                     wmanIf2SsDynamicServiceFail(1),
32                     wmanIf2SsDhcpSuccess(2),
33                     wmanIf2SsRssiStatusChange(3)}
34     MAX-ACCESS  read-write
35     STATUS      current
36     DESCRIPTION
37         "The object is used to enable Subscriber Station traps.
38         From left to right, the set bit indicates the corresponding
39         Subscriber Station trap is enabled."
40     ::= { wmanIf2SsTrapControl 1 }
41
42
43
44 wmanIf2SsThresholdConfigTable OBJECT-TYPE
45     SYNTAX      SEQUENCE OF WmanIf2SsThresholdConfigEntry
46     MAX-ACCESS  not-accessible
47     STATUS      current
48     DESCRIPTION
49         "This table contains threshold objects that can be set to
50         detect the threshold crossing events."
51     ::= { wmanIf2SsTrapControl 2 }
52
53
54
55 wmanIf2SsThresholdConfigEntry OBJECT-TYPE
56     SYNTAX      WmanIf2SsThresholdConfigEntry
57     MAX-ACCESS  not-accessible
58     STATUS      current
59     DESCRIPTION
60         "This table provides one row for each Ss, and is indexed
61         by ifIndex."
62     INDEX      { ifIndex }
63
64
65

```

```

1      ::= { wmanIf2SsThresholdConfigTable 1 }
2
3
4      WmanIf2SsThresholdConfigEntry ::= SEQUENCE {
5          wmanIf2SsRssiLowThreshold      Integer32,
6          wmanIf2SsRssiHighThreshold    Integer32}
7
8      wmanIf2SsRssiLowThreshold OBJECT-TYPE
9          SYNTAX      Integer32
10         UNITS       "dBm"
11         MAX-ACCESS  read-write
12         STATUS      current
13         DESCRIPTION
14             "Low RSSI threshold for generating the RSSI alarm trap."
15         ::= { wmanIf2SsThresholdConfigEntry 1 }
16
17
18
19      wmanIf2SsRssiHighThreshold OBJECT-TYPE
20          SYNTAX      Integer32
21          UNITS       "dBm"
22          MAX-ACCESS  read-write
23          STATUS      current
24          DESCRIPTION
25              "High RSSI threshold for generating a trap to indicate
26              the RSSI is restored."
27          ::= { wmanIf2SsThresholdConfigEntry 2 }
28
29
30
31      wmanIf2SsTlvUnknownTrap NOTIFICATION-TYPE
32          OBJECTS      {ifIndex,
33                      wmanIf2SsMacAddress,
34                      wmanIf2SsUnknownTlv}
35          STATUS      current
36          DESCRIPTION
37              "Event that notifies detection of unknown TLV during
38              the TLV parsing process."
39          ::= { wmanIf2SsTrapPrefix 1 }
40
41
42
43      wmanIf2SsDynamicServiceFailTrap NOTIFICATION-TYPE
44          OBJECTS      {ifIndex,
45                      wmanIf2SsMacAddress,
46                      wmanIf2SsDynamicServiceType,
47                      wmanIf2SsDynamicServiceFailReason}
48          STATUS      current
49          DESCRIPTION
50              "An event to report the failure of a dynamic service
51              operation happened during the dynamic services process
52              and detected in the BS side."
53          ::= { wmanIf2SsTrapPrefix 2 }
54
55
56
57      wmanIf2SsDhcpSuccessTrap NOTIFICATION-TYPE
58          OBJECTS      {ifIndex,
59                      wmanIf2SsMacAddress}
60          STATUS      current
61          DESCRIPTION
62              "An event to report a successful Handshake to establish IP
63              connectivity."
64
65

```

```

1      ::= { wmanIf2SsTrapPrefix 3 }
2
3
4  wmanIf2SsRssiStatusChangeTrap NOTIFICATION-TYPE
5      OBJECTS      {ifIndex,
6                      wmanIf2SsMacAddress,
7                      wmanIf2SsRssiStatus,
8                      wmanIf2SsRssiStatusInfo}
9
10     STATUS      current
11
12     DESCRIPTION
13         "An event to report that the downlink RSSI is below
14         wmanIf2SsRssiLowThreshold, or above
15         wmanIf2SsRssiHighThreshold after restore."
16     ::= { wmanIf2SsTrapPrefix 4 }
17
18  wmanIf2SsNotificationObjectsTable OBJECT-TYPE
19      SYNTAX      SEQUENCE OF WmanIf2SsNotificationObjectsEntry
20      MAX-ACCESS  not-accessible
21      STATUS      current
22      DESCRIPTION
23          "This table contains SS notification objects that have been
24          reported by the trap."
25      ::= { wmanIf2SsTrapDefinitions 1 }
26
27
28
29  wmanIf2SsNotificationObjectsEntry OBJECT-TYPE
30      SYNTAX      WmanIf2SsNotificationObjectsEntry
31      MAX-ACCESS  not-accessible
32      STATUS      current
33      DESCRIPTION
34          "This table provides one row for each SS that has
35          generated traps, and is indexed by ifIndex."
36      INDEX      { ifIndex }
37      ::= { wmanIf2SsNotificationObjectsTable 1 }
38
39
40
41  WmanIf2SsNotificationObjectsEntry ::= SEQUENCE {
42      wmanIf2SsMacAddress      MacAddress,
43      wmanIf2SsUnknownTlv      OCTET STRING,
44      wmanIf2SsDynamicServiceType  INTEGER,
45      wmanIf2SsDynamicServiceFailReason  OCTET STRING,
46      wmanIf2SsRssiStatus      INTEGER,
47      wmanIf2SsRssiStatusInfo  OCTET STRING}
48
49
50  wmanIf2SsMacAddress OBJECT-TYPE
51      SYNTAX      MacAddress
52      MAX-ACCESS  read-only
53      STATUS      current
54      DESCRIPTION
55          "The MAC address of the SS generating the trap."
56      ::= { wmanIf2SsNotificationObjectsEntry 1 }
57
58
59
60  wmanIf2SsUnknownTlv OBJECT-TYPE
61      SYNTAX      OCTET STRING (SIZE(0..65535))
62      MAX-ACCESS  read-only
63      STATUS      current
64      DESCRIPTION
65

```

```

1         "Indicating the value of the unknown TLV."
2         ::= { wmanIf2SsNotificationObjectsEntry 2 }
3
4
5 wmanIf2SsDynamicServiceType OBJECT-TYPE
6     SYNTAX      INTEGER {ssSfCreationReq(1),
7                     ssSfCreationRsp(2),
8                     ssSfCreationAck(3)}
9
10    MAX-ACCESS   read-only
11    STATUS       current
12    DESCRIPTION
13        "This object indicates the dynamic service flow
14        creation command type."
15    ::= { wmanIf2SsNotificationObjectsEntry 3 }
16
17
18 wmanIf2SsDynamicServiceFailReason OBJECT-TYPE
19     SYNTAX      OCTET STRING (SIZE(0..255))
20     MAX-ACCESS   read-only
21     STATUS       current
22     DESCRIPTION
23         "This object indicates the reason why the service flow
24         creation has failed."
25     ::= { wmanIf2SsNotificationObjectsEntry 4 }
26
27
28
29 wmanIf2SsRssiStatus OBJECT-TYPE
30     SYNTAX      INTEGER {ssRssiAlarm(1),
31                     ssRssiNoAlarm(2)}
32
33     MAX-ACCESS   read-only
34     STATUS       current
35     DESCRIPTION
36         "A RSSI alarm is generated if the RSSI is lower than
37         wmanIf2SsRssiLowThreshold, or above
38         wmanIf2SsRssiHighThreshold after alarm is restored."
39     ::= { wmanIf2SsNotificationObjectsEntry 5 }
40
41
42 wmanIf2SsRssiStatusInfo OBJECT-TYPE
43     SYNTAX      OCTET STRING (SIZE(0..255))
44     MAX-ACCESS   read-only
45     STATUS       current
46     DESCRIPTION
47         "This object provides additional information about RSSI
48         alarm. It is implementation specific"
49     ::= { wmanIf2SsNotificationObjectsEntry 6 }
50
51
52
53 --
54 -- Subscriber station PHY Group
55 --
56 wmanIf2SsPhy OBJECT IDENTIFIER ::= { wmanIf2SsObjects 5 }
57
58
59 --
60 -- SS OFDM PHY objects
61 --
62 wmanIf2SsOfdmPhy OBJECT IDENTIFIER ::= { wmanIf2SsPhy 1 }
63
64
65 wmanIf2SsOfdmUplinkChannelTable OBJECT-TYPE

```

```

1      SYNTAX      SEQUENCE OF WmanIf2SsOfdmUplinkChannelEntry
2      MAX-ACCESS  not-accessible
3      STATUS      current
4      DESCRIPTION
5
6          "This table contains UCD channel attributes, defining the
7          transmission characteristics of uplink channels"
8      REFERENCE
9
10         "Table 349 and Table 352, in IEEE Std 802.16-2004"
11     ::= { wmanIf2SsOfdmPhy 1 }
12
13 wmanIf2SsOfdmUplinkChannelEntry OBJECT-TYPE
14     SYNTAX      WmanIf2SsOfdmUplinkChannelEntry
15     MAX-ACCESS  not-accessible
16     STATUS      current
17     DESCRIPTION
18
19         "This table provides one row for each uplink channel of
20         multi-sector BS, and is indexed by BS ifIndex. An entry
21         in this table exists for each ifEntry of BS with an
22         ifType of propBWAp2Mp."
23     INDEX { ifIndex }
24     ::= { wmanIf2SsOfdmUplinkChannelTable 1 }
25
26
27 WmanIf2SsOfdmUplinkChannelEntry ::= SEQUENCE {
28     wmanIf2SsOfdmCtBasedResvTimeout      INTEGER,
29     wmanIf2SsOfdmBwReqOppSize             INTEGER,
30     wmanIf2SsOfdmRangReqOppSize           INTEGER,
31     wmanIf2SsOfdmUplinkCenterFreq         Unsigned32,
32     wmanIf2SsOfdmNumSubChReqRegionFull    INTEGER,
33     wmanIf2SsOfdmNumSymbolsReqRegionFull  INTEGER,
34     wmanIf2SsOfdmSubChFocusCtCode         INTEGER,
35     wmanIf2SsOfdmUpLinkChannelId         INTEGER}
36
37
38 wmanIf2SsOfdmCtBasedResvTimeout OBJECT-TYPE
39     SYNTAX      INTEGER (1..255)
40     MAX-ACCESS  read-only
41     STATUS      current
42     DESCRIPTION
43
44         "The number of UL-MAPs to receive before contention-based
45         reservation is attempted again for the same connection."
46     REFERENCE
47
48         "Table 349, in IEEE Std 802.16-2004"
49     ::= { wmanIf2SsOfdmUplinkChannelEntry 1 }
50
51
52 wmanIf2SsOfdmBwReqOppSize OBJECT-TYPE
53     SYNTAX      INTEGER (1..65535)
54     UNITS       "PS"
55     MAX-ACCESS  read-only
56     STATUS      current
57     DESCRIPTION
58
59         "Size (in units of PS) of PHY payload that SS may use to
60         format and transmit a bandwidth request message in a
61         contention request opportunity. The value includes all
62         PHY overhead as well as allowance for the MAC data the
63         message may hold."
64
65

```



```

1      REFERENCE
2          "Table 349, in IEEE Std 802.16-2004"
3      ::= { wmanIf2SsOfdmUplinkChannelEntry 2 }
4
5
6  wmanIf2SsOfdmRangReqOppSize OBJECT-TYPE
7      SYNTAX      INTEGER (1..65535)
8      UNITS       "PS"
9      MAX-ACCESS  read-only
10     STATUS      current
11     DESCRIPTION
12         "Size (in units of PS) of PHY payload that SS may use to
13         format and transmit a RNG-REQ message in a contention
14         request opportunity. The value includes all PHY overhead
15         as well as allowance for the MAC data the message may
16         hold and the maximum SS/BS roundtrip propagation delay."
17     REFERENCE
18         "Table 349, in IEEE Std 802.16-2004"
19     ::= { wmanIf2SsOfdmUplinkChannelEntry 3 }
20
21
22  wmanIf2SsOfdmUplinkCenterFreq OBJECT-TYPE
23     SYNTAX      Unsigned32
24     UNITS       "kHz"
25     MAX-ACCESS  read-only
26     STATUS      current
27     DESCRIPTION
28         " Uplink center frequency (kHz)"
29     REFERENCE
30         "Table 349, in IEEE Std 802.16-2004"
31     ::= { wmanIf2SsOfdmUplinkChannelEntry 4 }
32
33
34  wmanIf2SsOfdmNumSubChReqRegionFull OBJECT-TYPE
35     SYNTAX      INTEGER {oneSubchannel(0),
36                     twoSubchannels(1),
37                     fourSubchannels(2),
38                     eightSubchannels(3),
39                     sixteenSubchannels(4)}
40     MAX-ACCESS  read-only
41     STATUS      current
42     DESCRIPTION
43         "Number of subchannels used by each transmit
44         opportunity when REQ Region-Full is allocated in
45         subchannelization region."
46     REFERENCE
47         "Table 352, in IEEE Std 802.16-2004"
48     ::= { wmanIf2SsOfdmUplinkChannelEntry 5 }
49
50
51  wmanIf2SsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
52     SYNTAX      INTEGER (0..31)
53     MAX-ACCESS  read-only
54     STATUS      current
55     DESCRIPTION
56         "Number of OFDM symbols used by each transmit
57         opportunity when REQ Region-Full is allocated in
58         subchannelization region."
59
60
61

```

```

1      REFERENCE
2          "Table 352, in IEEE Std 802.16-2004"
3      ::= { wmanIf2SsOfdmUplinkChannelEntry 6 }
4
5
6      wmanIf2SsOfdmSubChFocusCtCode OBJECT-TYPE
7          SYNTAX      INTEGER (0..8)
8          MAX-ACCESS   read-only
9          STATUS       current
10         DESCRIPTION
11             "Number of contention codes (CSE) that shall only be used to
12             request a subchannelized allocation. Default value 0.
13             Allowed values 0-8."
14         REFERENCE
15             "Table 352, in IEEE Std 802.16-2004"
16         DEFVAL       { 0 }
17         ::= { wmanIf2SsOfdmUplinkChannelEntry 7 }
18
19
20
21      wmanIf2SsOfdmUpLinkChannelId OBJECT-TYPE
22          SYNTAX      INTEGER (0..255)
23          MAX-ACCESS   read-only
24          STATUS       current
25          DESCRIPTION
26              "The identifier of the uplink channel to which this
27              message refers."
28          REFERENCE
29              "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
30          ::= { wmanIf2SsOfdmUplinkChannelEntry 8 }
31
32
33
34      wmanIf2SsOfdmDownlinkChannelTable OBJECT-TYPE
35          SYNTAX      SEQUENCE OF WmanIf2SsOfdmDownlinkChannelEntry
36          MAX-ACCESS   not-accessible
37          STATUS       current
38          DESCRIPTION
39              "This table contains DCD channel attributes, defining the
40              transmission characteristics of downlink channels"
41          REFERENCE
42              "Table 358, in IEEE Std 802.16-2004"
43          ::= { wmanIf2SsOfdmPhy 2 }
44
45
46
47      wmanIf2SsOfdmDownlinkChannelEntry OBJECT-TYPE
48          SYNTAX      WmanIf2SsOfdmDownlinkChannelEntry
49          MAX-ACCESS   not-accessible
50          STATUS       current
51          DESCRIPTION
52              "This table provides one row for each downlink channel of
53              multi-sector BS, and is indexed by BS ifIndex. An entry
54              in this table exists for each ifEntry of BS with an
55              ifType of propBWAp2Mp."
56          INDEX { ifIndex }
57          ::= { wmanIf2SsOfdmDownlinkChannelTable 1 }
58
59
60
61      WmanIf2SsOfdmDownlinkChannelEntry ::= SEQUENCE {
62          wmanIf2SsOfdmBsEIRP      INTEGER,
63          wmanIf2SsOfdmChannelNumber WmanIf2ChannelNumber,
64
65

```

```

1      wmanIf2SsOfdmTTG                INTEGER,
2      wmanIf2SsOfdmRTG                INTEGER,
3      wmanIf2SsOfdmInitRngMaxRSS      INTEGER,
4      wmanIf2SsOfdmDownlinkCenterFreq Unsigned32,
5      wmanIf2SsOfdmBsId               WmanIf2BsIdType,
6      wmanIf2SsOfdmMacVersion          WmanIf2MacVersion,
7      wmanIf2SsOfdmFrameDurationCode  INTEGER,
8      wmanIf2SsOfdmDownLinkChannelId  INTEGER}
9
10
11
12  wmanIf2SsOfdmBsEIRP OBJECT-TYPE
13      SYNTAX      INTEGER (-32768..32767)
14      UNITS       "dBm"
15      MAX-ACCESS  read-only
16      STATUS      current
17      DESCRIPTION
18          "The EIRP is the equivalent isotropic radiated power of
19           the base station, which is computed for a simple
20           single-antenna transmitter."
21      REFERENCE
22          "Table 358, in IEEE Std 802.16-2004"
23      ::= { wmanIf2SsOfdmDownlinkChannelEntry 1 }
24
25
26
27  wmanIf2SsOfdmChannelNumber OBJECT-TYPE
28      SYNTAX      WmanIf2ChannelNumber
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32          "Downlink channel number as defined in 8.5.
33           Used for license-exempt operation only."
34      REFERENCE
35          "Table 358, in IEEE Std 802.16-2004"
36      ::= { wmanIf2SsOfdmDownlinkChannelEntry 2 }
37
38
39
40
41  wmanIf2SsOfdmTTG OBJECT-TYPE
42      SYNTAX      INTEGER (0..255)
43      MAX-ACCESS  read-only
44      STATUS      current
45      DESCRIPTION
46          "Transmit / Receive Transition Gap."
47      REFERENCE
48          "Table 358, in IEEE Std 802.16-2004"
49      ::= { wmanIf2SsOfdmDownlinkChannelEntry 3 }
50
51
52
53  wmanIf2SsOfdmRTG OBJECT-TYPE
54      SYNTAX      INTEGER (0..255)
55      MAX-ACCESS  read-only
56      STATUS      current
57      DESCRIPTION
58          "Receive / Transmit Transition Gap."
59      REFERENCE
60          "Table 358, in IEEE Std 802.16-2004"
61      ::= { wmanIf2SsOfdmDownlinkChannelEntry 4 }
62
63
64
65  wmanIf2SsOfdmInitRngMaxRSS OBJECT-TYPE

```

```

1      SYNTAX      INTEGER (-32768..32767)
2      UNITS       "dBm"
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "Initial Ranging Max. equivalent isotropic received power
7          at BS Signed in units of 1 dBm."
8      REFERENCE
9          "Table 358, in IEEE Std 802.16-2004"
10     ::= { wmanIf2SsOfdmDownlinkChannelEntry 5 }
11
12 wmanIf2SsOfdmDownlinkCenterFreq OBJECT-TYPE
13     SYNTAX      Unsigned32
14     UNITS       "kHz"
15     MAX-ACCESS  read-only
16     STATUS      current
17     DESCRIPTION
18         "Downlink center frequency (kHz)."
```

```

19     REFERENCE
20         "Table 358, in IEEE Std 802.16-2004"
21     ::= { wmanIf2SsOfdmDownlinkChannelEntry 6 }
22
23 wmanIf2SsOfdmBsId OBJECT-TYPE
24     SYNTAX      WmanIf2BsIdType
25     MAX-ACCESS  read-only
26     STATUS      current
27     DESCRIPTION
28         "Base station ID."
29     REFERENCE
30         "Table 358, in IEEE Std 802.16-2004"
31     ::= { wmanIf2SsOfdmDownlinkChannelEntry 7 }
32
33 wmanIf2SsOfdmMacVersion OBJECT-TYPE
34     SYNTAX      WmanIf2MacVersion
35     MAX-ACCESS  read-only
36     STATUS      current
37     DESCRIPTION
38         "This parameter specifies the version of 802.16 to which
39         the message originator conforms."
40     REFERENCE
41         "Table 358, in IEEE Std 802.16-2004"
42     ::= { wmanIf2SsOfdmDownlinkChannelEntry 8 }
43
44 wmanIf2SsOfdmFrameDurationCode OBJECT-TYPE
45     SYNTAX      INTEGER {duration2dot5ms(0),
46                          duration4ms(1),
47                          duration5ms(2),
48                          duration8ms(3),
49                          duration10ms(4),
50                          duration12dot5ms(5),
51                          duration20ms(6)}
52     MAX-ACCESS  read-only
53     STATUS      current
54     DESCRIPTION

```

```

1         "The duration of the frame. The frame duration code
2         values are specified in Table 230."
3     REFERENCE
4         "Subclause 11.4.1, Table 230, in IEEE Std 802.16-2004"
5     ::= { wmanIf2SsOfdmDownlinkChannelEntry 9 }
6
7
8     wmanIf2SsOfdmDownLinkChannelId OBJECT-TYPE
9         SYNTAX      INTEGER (0..255)
10        MAX-ACCESS   read-only
11        STATUS       current
12        DESCRIPTION
13            "The identifier of the downlink channel to which this
14            message refers."
15        REFERENCE
16            "Subclause 6.3.2.3.1, Table 15, in IEEE Std 802.16-2004"
17        ::= { wmanIf2SsOfdmDownlinkChannelEntry 10 }
18
19
20
21     wmanIf2SsOfdmUcdBurstProfileTable OBJECT-TYPE
22         SYNTAX      SEQUENCE OF WmanIf2SsOfdmUcdBurstProfileEntry
23         MAX-ACCESS   not-accessible
24         STATUS       current
25         DESCRIPTION
26             "This table contains UCD burst profiles for each uplink
27             channel"
28         REFERENCE
29             "Table 356, in IEEE Std 802.16-2004"
30         ::= { wmanIf2SsOfdmPhy 3 }
31
32
33
34     wmanIf2SsOfdmUcdBurstProfileEntry OBJECT-TYPE
35         SYNTAX      WmanIf2SsOfdmUcdBurstProfileEntry
36         MAX-ACCESS   not-accessible
37         STATUS       current
38         DESCRIPTION
39             "This table provides one row for each UCD burst profile.
40             This table is double indexed. The primary index is an
41             ifIndex with an ifType of propBWA2Mp. The secondary index
42             is wmanIf2SsOfdmUcdBurstProfIndex."
43         INDEX { ifIndex, wmanIf2SsOfdmUiucIndex }
44         ::= { wmanIf2SsOfdmUcdBurstProfileTable 1 }
45
46
47
48     WmanIf2SsOfdmUcdBurstProfileEntry ::= SEQUENCE {
49         wmanIf2SsOfdmUiucIndex          INTEGER,
50         wmanIf2SsOfdmUcdFecCodeType     WmanIf2OfdmFecCodeType,
51         wmanIf2SsOfdmFocusCtPowerBoost  INTEGER,
52         wmanIf2SsOfdmUcdTcsEnable       INTEGER}
53
54
55
56     wmanIf2SsOfdmUiucIndex OBJECT-TYPE
57         SYNTAX      INTEGER (5 .. 12)
58         MAX-ACCESS   not-accessible
59         STATUS       current
60         DESCRIPTION
61             "The Uplink Interval Usage Code indicates the uplink burst
62             profile in the UCD message, and is used along with ifIndex
63             to identify an entry in the
64
65

```

```

1         wmanIf2SsOfdmUcdBurstProfileTable."
2     REFERENCE
3         "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
4         ::= { wmanIf2SsOfdmUcdBurstProfileEntry 1 }
5
6
7     wmanIf2SsOfdmUcdFecCodeType OBJECT-TYPE
8         SYNTAX      WmanIf2OfdmFecCodeType
9         MAX-ACCESS   read-only
10        STATUS      current
11        DESCRIPTION
12            "Uplink FEC code type and modulation type"
13        REFERENCE
14            "Table 356, in IEEE Std 802.16-2004"
15            ::= { wmanIf2SsOfdmUcdBurstProfileEntry 2 }
16
17
18
19    wmanIf2SsOfdmFocusCtPowerBoost OBJECT-TYPE
20        SYNTAX      INTEGER (0 .. 255)
21        MAX-ACCESS   read-only
22        STATUS      current
23        DESCRIPTION
24            "The power boost in dB of focused contention carriers, as
25            described in 8.3.6.3.3."
26        REFERENCE
27            "Table 356, in IEEE Std 802.16-2004"
28            ::= { wmanIf2SsOfdmUcdBurstProfileEntry 3 }
29
30
31
32    wmanIf2SsOfdmUcdTcsEnable OBJECT-TYPE
33        SYNTAX      INTEGER {tcsDisabled(0),
34                        tcsEnabled(1)}
35        MAX-ACCESS   read-only
36        STATUS      current
37        DESCRIPTION
38            "This parameter determines the transmission convergence
39            sublayer, as described in 8.1.4.3, can be enabled on a
40            per-burst basis for both uplink and downlink. through
41            DIUC/UIUC messages."
42        REFERENCE
43            "Table 356, in IEEE Std 802.16-2004"
44            ::= { wmanIf2SsOfdmUcdBurstProfileEntry 4 }
45
46
47
48
49    wmanIf2SsOfdmDcdBurstProfileTable OBJECT-TYPE
50        SYNTAX      SEQUENCE OF WmanIf2SsOfdmDcdBurstProfileEntry
51        MAX-ACCESS   not-accessible
52        STATUS      current
53        DESCRIPTION
54            "This table provides one row for each DCD burst profile.
55            This table is double indexed. The primary index is an
56            ifIndex with an ifType of propBWAp2Mp. The secondary
57            index is wmanIf2SsOfdmDiucIndex."
58        REFERENCE
59            "Table 362, in IEEE Std 802.16-2004"
60            ::= { wmanIf2SsOfdmPhy 4 }
61
62
63
64    wmanIf2SsOfdmDcdBurstProfileEntry OBJECT-TYPE
65

```

```

1      SYNTAX      WmanIf2SsOfdmDcdBurstProfileEntry
2      MAX-ACCESS  not-accessible
3      STATUS      current
4      DESCRIPTION
5
6          "This table provides one row for each DCD burst profile.
7          This table is double indexed. The primary index is an
8          ifIndex with an ifType of propBWA2Mp. The secondary index
9          is wmanIf2SsOfdmDcdBurstProfIndex."
10     INDEX { ifIndex, wmanIf2SsOfdmDiucIndex }
11     ::= { wmanIf2SsOfdmDcdBurstProfileTable 1 }
12
13
14     WmanIf2SsOfdmDcdBurstProfileEntry ::= SEQUENCE {
15         wmanIf2SsOfdmDiucIndex          INTEGER,
16         wmanIf2SsOfdmDownlinkFrequency  Unsigned32,
17         wmanIf2SsOfdmDcdFecCodeType     WmanIf2OfdmFecCodeType,
18         wmanIf2SsOfdmDiucMandatoryExitThresh  INTEGER,
19         wmanIf2SsOfdmDiucMinEntryThresh  INTEGER,
20         wmanIf2SsOfdmTcsEnable           INTEGER}
21
22
23
24     wmanIf2SsOfdmDiucIndex OBJECT-TYPE
25         SYNTAX      INTEGER (1..11)
26         MAX-ACCESS  not-accessible
27         STATUS      current
28         DESCRIPTION
29
30             "The Downlink Interval Usage Code indicates the downlink
31             burst profile in the DCD message, and is used along with
32             ifIndex to identify an entry in the
33             wmanIf2SsOfdmDcdBurstProfileTable."
34         REFERENCE
35
36             "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
37         ::= { wmanIf2SsOfdmDcdBurstProfileEntry 1 }
38
39
40     wmanIf2SsOfdmDownlinkFrequency OBJECT-TYPE
41         SYNTAX      Unsigned32
42         UNITS        "kHz"
43         MAX-ACCESS  read-only
44         STATUS      current
45         DESCRIPTION
46
47             "Downlink Frequency (kHz)."

```

```

1      SYNTAX      INTEGER (0..255)
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or
6          below where this DIUC can no longer be used and where this
7          change to a more robust DIUC is required in 0.25 dB units."
8      REFERENCE
9          "Table 362, in IEEE Std 802.16-2004"
10     ::= { wmanIf2SsOfdmDcdBurstProfileEntry 4 }
11
12 wmanIf2SsOfdmDiucMinEntryThresh OBJECT-TYPE
13     SYNTAX      INTEGER (0..255)
14     MAX-ACCESS  read-only
15     STATUS      current
16     DESCRIPTION
17         "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
18         required to start using this DIUC when changing from a more
19         robust DIUC is required, in 0.25 dB units."
20     REFERENCE
21         "Table 362, in IEEE Std 802.16-2004"
22     ::= { wmanIf2SsOfdmDcdBurstProfileEntry 5 }
23
24 wmanIf2SsOfdmTcsEnable OBJECT-TYPE
25     SYNTAX      INTEGER {tcsDisabled (0),
26                     tcsEnabled (1)}
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30         "Indicates whether Transmission CONvergence Sublayer
31         is enabled or disabled."
32     REFERENCE
33         "Table 362, in IEEE Std 802.16-2004"
34     ::= { wmanIf2SsOfdmDcdBurstProfileEntry 6 }
35
36 --
37 -- SS OFDMA PHY objects
38 --
39 wmanIf2SsOfdmaPhy OBJECT IDENTIFIER ::= { wmanIf2SsPhy 2 }
40
41 wmanIf2SsOfdmaUplinkChannelTable OBJECT-TYPE
42     SYNTAX      SEQUENCE OF WmanIf2SsOfdmaUplinkChannelEntry
43     MAX-ACCESS  not-accessible
44     STATUS      current
45     DESCRIPTION
46         "This table contains UCD channel attributes, defining the
47         transmission characteristics of uplink channels"
48     REFERENCE
49         "Subclause 11.3.1, Table 349 and Table 353, in IEEE Std
50         802.16-2004"
51     ::= { wmanIf2SsOfdmaPhy 1 }
52
53 wmanIf2SsOfdmaUplinkChannelEntry OBJECT-TYPE
54     SYNTAX      WmanIf2SsOfdmaUplinkChannelEntry

```



```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table provides one row for each uplink channel of
5          multi-sector BS, and is indexed by BS ifIndex. An entry
6          in this table exists for each ifEntry of BS with an
7          ifType of propBWA2Mp."
8      INDEX          { ifIndex }
9      ::= { wmanIf2SsOfdmaUplinkChannelTable 1 }
10
11
12
13  WmanIf2SsOfdmaUplinkChannelEntry ::= SEQUENCE {
14      wmanIf2SsOfdmaCtBasedResvTimeout      INTEGER,
15      wmanIf2SsOfdmaBwReqOppSize             INTEGER,
16      wmanIf2SsOfdmaRangReqOppSize           INTEGER,
17      wmanIf2SsOfdmaUplinkCenterFreq         Unsigned32,
18      wmanIf2SsOfdmaInitRngCodes             INTEGER,
19      wmanIf2SsOfdmaPeriodicRngCodes         INTEGER,
20      wmanIf2SsOfdmaBWRngCodes              INTEGER,
21      wmanIf2SsOfdmaPerRngBackoffStart       INTEGER,
22      wmanIf2SsOfdmaPerRngBackoffEnd        INTEGER,
23      wmanIf2SsOfdmaStartOfRngCodes          INTEGER,
24      wmanIf2SsOfdmaPermutationBase          INTEGER,
25      wmanIf2SsOfdmaULAllocSubchBitmap       OCTET STRING,
26      wmanIf2SsOfdmaOptPermULAllocSubchBitmap OCTET STRING,
27      wmanIf2SsOfdmaBandAMCAllocThreshold    INTEGER,
28      wmanIf2SsOfdmaBandAMCReleaseThreshold  INTEGER,
29      wmanIf2SsOfdmaBandAMCAllocTimer        INTEGER,
30      wmanIf2SsOfdmaBandAMCReleaseTimer      INTEGER,
31      wmanIf2SsOfdmaBandStatRepMAXPeriod     INTEGER,
32      wmanIf2SsOfdmaBandAMCRetryTimer        INTEGER,
33      wmanIf2SsOfdmaSafetyChAllocThreshold   INTEGER,
34      wmanIf2SsOfdmaSafetyChReleaseThreshold INTEGER,
35      wmanIf2SsOfdmaSafetyChAllocTimer       INTEGER,
36      wmanIf2SsOfdmaSafetyChReleaseTimer     INTEGER,
37      wmanIf2SsOfdmaBinStatRepMAXPeriod      INTEGER,
38      wmanIf2SsOfdmaSafetyChARetryTimer      INTEGER,
39      wmanIf2SsOfdmaHARQAackDelayULBurst     INTEGER,
40      wmanIf2SsOfdmaCQICHBandAMCTranaDelay   INTEGER}
41
42
43
44
45
46
47
48  wmanIf2SsOfdmaCtBasedResvTimeout OBJECT-TYPE
49      SYNTAX      INTEGER (1..255)
50      MAX-ACCESS  read-only
51      STATUS      current
52      DESCRIPTION
53          "The number of UL-MAPs to receive before contention-based
54          reservation is attempted again for the same connection."
55      REFERENCE
56          "Table 349, in IEEE Std 802.16-2004"
57      ::= { wmanIf2SsOfdmaUplinkChannelEntry 1 }
58
59
60
61  wmanIf2SsOfdmaBwReqOppSize OBJECT-TYPE
62      SYNTAX      INTEGER (1..65535)
63      UNITS       "PS"
64      MAX-ACCESS  read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "Size (in units of PS) of PHY payload that SS may use to
4          format and transmit a bandwidth request message in a
5          contention request opportunity. The value includes all
6          PHY overhead as well as allowance for the MAC data the
7          message may hold."
8
9      REFERENCE
10         "Table 349, in IEEE Std 802.16-2004"
11         ::= { wmanIf2SsOfdmaUplinkChannelEntry 2 }
12
13
14 wmanIf2SsOfdmaRangReqOppSize OBJECT-TYPE
15     SYNTAX      INTEGER (1..65535)
16     UNITS       "PS"
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20         "Size (in units of PS) of PHY payload that SS may use to
21         format and transmit a RNG-REQ message in a contention
22         request opportunity. The value includes all PHY overhead
23         as well as allowance for the MAC data the message may
24         hold and the maximum SS/BS roundtrip propagation delay."
25
26     REFERENCE
27         "Table 349, in IEEE Std 802.16-2004"
28         ::= { wmanIf2SsOfdmaUplinkChannelEntry 3 }
29
30
31
32 wmanIf2SsOfdmaUplinkCenterFreq OBJECT-TYPE
33     SYNTAX      Unsigned32
34     UNITS       "kHz"
35     MAX-ACCESS  read-only
36     STATUS      current
37     DESCRIPTION
38         " Uplink center frequency (kHz)"
39
40     REFERENCE
41         "Table 349, in IEEE Std 802.16-2004"
42         ::= { wmanIf2SsOfdmaUplinkChannelEntry 4 }
43
44
45
46 wmanIf2SsOfdmaInitRngCodes OBJECT-TYPE
47     SYNTAX      INTEGER (0..255)
48     MAX-ACCESS  read-only
49     STATUS      current
50     DESCRIPTION
51         "Number of initial ranging CDMA codes. Possible values are
52         0..255. The total number of wmanIf2SsOfdmaInitRngCodes,
53         wmanIf2SsOfdmaPeriodicRngCodes and wmanIf2SsOfdmaBWRngCodes
54         shall be equal or less than 256."
55
56     REFERENCE
57         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
58
59     DEFVAL      { 30 }
60     ::= { wmanIf2SsOfdmaUplinkChannelEntry 5 }
61
62
63 wmanIf2SsOfdmaPeriodicRngCodes OBJECT-TYPE
64     SYNTAX      INTEGER (0..255)
65     MAX-ACCESS  read-only

```

```

1      STATUS      current
2      DESCRIPTION
3          "Number of periodic ranging CDMA codes. Possible values are
4              0..255. The total number of wmanIf2SsOfdmaInitRngCodes,
5              wmanIf2SsOfdmaPeriodicRngCodes and wmanIf2SsOfdmaBWReqCodes
6              shall be equal or less than 256."
7
8      REFERENCE
9          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
10
11     DEFVAL      { 30 }
12     ::= { wmanIf2SsOfdmaUplinkChannelEntry 6 }
13
14 wmanIf2SsOfdmaBWReqCodes OBJECT-TYPE
15     SYNTAX      INTEGER (0..255)
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "Number of bandwidth request codes. Possible values are
20             0..255. The total number of wmanIf2SsOfdmaInitRngCodes,
21             wmanIf2SsOfdmaPeriodicRngCodes and wmanIf2SsOfdmaBWReqCodes
22             shall be equal or less than 256."
23
24     REFERENCE
25         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
26
27     DEFVAL      { 30 }
28     ::= { wmanIf2SsOfdmaUplinkChannelEntry 7 }
29
30
31 wmanIf2SsOfdmaPerRngBackoffStart OBJECT-TYPE
32     SYNTAX      INTEGER (0..15)
33     MAX-ACCESS  read-only
34     STATUS      current
35     DESCRIPTION
36         "Initial backoff window size for periodic ranging
37             contention, expressed as a power of 2."
38
39     REFERENCE
40         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
41
42     DEFVAL      { 0 }
43     ::= { wmanIf2SsOfdmaUplinkChannelEntry 8 }
44
45
46 wmanIf2SsOfdmaPerRngBackoffEnd OBJECT-TYPE
47     SYNTAX      INTEGER (0 .. 15)
48     MAX-ACCESS  read-only
49     STATUS      current
50     DESCRIPTION
51         "Final backoff window size for periodic ranging contention,
52             expressed as a power of 2."
53
54     REFERENCE
55         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
56
57     DEFVAL      { 15 }
58     ::= { wmanIf2SsOfdmaUplinkChannelEntry 9 }
59
60 wmanIf2SsOfdmaStartOfRngCodes OBJECT-TYPE
61     SYNTAX      INTEGER (0..255)
62     MAX-ACCESS  read-only
63     STATUS      current
64     DESCRIPTION
65

```

1 "Indicates the starting number, S, of the group of codes
2 used for this uplink. All the ranging codes used on this
3 uplink will be between S and $((S+N+M+L) \bmod 256)$. Where,
4 N is the number of initial-ranging codes M is the number
5 of periodic-ranging codes L is the number of
6 bandwidth-request codes The range of values is 0 S255"
7
8 REFERENCE
9 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
10 DEFVAL { 0 }
11 ::= { wmanIf2SsOfdmaUplinkChannelEntry 10 }
12
13
14 wmanIf2SsOfdmaPermutationBase OBJECT-TYPE
15 SYNTAX INTEGER (0..255)
16 MAX-ACCESS read-only
17 STATUS current
18 DESCRIPTION
19 "Determines the UL_IDcell parameter for the subcarrier
20 permutation to be used on this uplink channel"
21
22 REFERENCE
23 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
24 DEFVAL { 0 }
25 ::= { wmanIf2SsOfdmaUplinkChannelEntry 11 }
26
27
28
29 wmanIf2SsOfdmaULAllocSubchBitmap OBJECT-TYPE
30 SYNTAX OCTET STRING (SIZE (9))
31 MAX-ACCESS read-only
32 STATUS current
33 DESCRIPTION
34 "This is a bitmap describing the sub-channels allocated
35 to the segment in the UL, when using the uplink PUSC
36 permutation. The LSB of the first byte shall correspond to
37 subchannel 0. For any bit that is not set,
38 the corresponding subchannel shall not be used by the SS
39 on that segment"
40
41 REFERENCE
42 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
43 ::= { wmanIf2SsOfdmaUplinkChannelEntry 12 }
44
45
46
47 wmanIf2SsOfdmaOptPermULAllocSubchBitmap OBJECT-TYPE
48 SYNTAX OCTET STRING (SIZE (13))
49 MAX-ACCESS read-only
50 STATUS current
51 DESCRIPTION
52 "This is a bitmap describing the sub-channels allocated to
53 the segment in the UL, when using the uplink optional PUSC
54 permutation (see 8.4.6.2.5 in IEEE Std 802.16-2004). The
55 LSB of the first byte shall correspond to subchannel 0.
56 For any bit that is not set, the corresponding subchannel
57 shall not be used by the SS on that segment"
58
59 REFERENCE
60 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
61 ::= { wmanIf2SsOfdmaUplinkChannelEntry 13 }
62
63
64
65 wmanIf2SsOfdmaBandAMCAllocThreshold OBJECT-TYPE

```

1      SYNTAX      INTEGER (0 .. 255)
2      UNITS       "dB"
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This object defines the OFDMA band AMC allocation
7          threshold."
8      REFERENCE
9          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
10     ::= { wmanIf2SsOfdmaUplinkChannelEntry 14 }
11
12 wmanIf2SsOfdmaBandAMCReleaseThreshold OBJECT-TYPE
13     SYNTAX      INTEGER (0 .. 255)
14     UNITS       "dB"
15     MAX-ACCESS  read-only
16     STATUS      current
17     DESCRIPTION
18         "This object defines the OFDMA band AMC release
19         threshold."
20     REFERENCE
21         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
22     ::= { wmanIf2SsOfdmaUplinkChannelEntry 15 }
23
24 wmanIf2SsOfdmaBandAMCAllocTimer OBJECT-TYPE
25     SYNTAX      INTEGER (0 .. 255)
26     UNITS       "Frame"
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30         "This object defines the OFDMA band AMC allocation
31         timer."
32     REFERENCE
33         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
34     ::= { wmanIf2SsOfdmaUplinkChannelEntry 16 }
35
36 wmanIf2SsOfdmaBandAMCReleaseTimer OBJECT-TYPE
37     SYNTAX      INTEGER (0 .. 255)
38     UNITS       "Frame"
39     MAX-ACCESS  read-only
40     STATUS      current
41     DESCRIPTION
42         "This object defines the OFDMA band AMC release
43         timer."
44     REFERENCE
45         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
46     ::= { wmanIf2SsOfdmaUplinkChannelEntry 17 }
47
48 wmanIf2SsOfdmaBandStatRepMAXPeriod OBJECT-TYPE
49     SYNTAX      INTEGER (0 .. 255)
50     UNITS       "Frame"
51     MAX-ACCESS  read-only
52     STATUS      current
53     DESCRIPTION
54         "This object defines the OFDMA band status reporting
55         threshold."
56
57
58
59
60
61
62
63
64
65

```

```

1         maximum period."
2     REFERENCE
3         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
4     ::= { wmanIf2SsOfdmaUplinkChannelEntry 18 }
5
6
7 wmanIf2SsOfdmaBandAMCRetryTimer OBJECT-TYPE
8     SYNTAX      INTEGER (0 .. 255)
9     UNITS       "Frame"
10    MAX-ACCESS  read-only
11    STATUS      current
12    DESCRIPTION
13        "This object defines the OFDMA band AMC retry
14        timer."
15    REFERENCE
16        "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
17    ::= { wmanIf2SsOfdmaUplinkChannelEntry 19 }
18
19
20
21 wmanIf2SsOfdmaSafetyChAllocThreshold OBJECT-TYPE
22     SYNTAX      INTEGER (0 .. 255)
23     UNITS       "dB"
24     MAX-ACCESS  read-only
25     STATUS      current
26     DESCRIPTION
27         "This object defines the OFDMA safety channel allocation
28         threshold."
29     REFERENCE
30         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
31     ::= { wmanIf2SsOfdmaUplinkChannelEntry 20 }
32
33
34
35 wmanIf2SsOfdmaSafetyChReleaseThreshold OBJECT-TYPE
36     SYNTAX      INTEGER (0 .. 255)
37     UNITS       "dB"
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41         "This object defines the OFDMA safety channel release
42         threshold."
43     REFERENCE
44         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
45     ::= { wmanIf2SsOfdmaUplinkChannelEntry 21 }
46
47
48
49 wmanIf2SsOfdmaSafetyChAllocTimer OBJECT-TYPE
50     SYNTAX      INTEGER (0 .. 255)
51     UNITS       "Frame"
52     MAX-ACCESS  read-only
53     STATUS      current
54     DESCRIPTION
55         "This object defines the OFDMA safety channel allocation
56         timer."
57     REFERENCE
58         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
59     ::= { wmanIf2SsOfdmaUplinkChannelEntry 22 }
60
61
62
63 wmanIf2SsOfdmaSafetyChReleaseTimer OBJECT-TYPE
64
65

```

```

1      SYNTAX      INTEGER (0 .. 255)
2      UNITS       "Frame"
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This object defines the OFDMA safety channel release
7          timer."
8
9      REFERENCE
10         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
11         ::= { wmanIf2SsOfdmaUplinkChannelEntry 23 }
12
13
14 wmanIf2SsOfdmaBinStatRepMAXPeriod OBJECT-TYPE
15     SYNTAX      INTEGER (0 .. 255)
16     UNITS       "Frame"
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20         "This object defines the OFDMA bin status reporting
21         maximum period."
22
23     REFERENCE
24         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
25         ::= { wmanIf2SsOfdmaUplinkChannelEntry 24 }
26
27
28 wmanIf2SsOfdmaSafetyChaRetryTimer OBJECT-TYPE
29     SYNTAX      INTEGER (0 .. 255)
30     UNITS       "Frame"
31     MAX-ACCESS  read-only
32     STATUS      current
33     DESCRIPTION
34         "This object defines the OFDMA safety channel retry
35         timer."
36
37     REFERENCE
38         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
39         ::= { wmanIf2SsOfdmaUplinkChannelEntry 25 }
40
41
42 wmanIf2SsOfdmaHARQAackDelayULBurst OBJECT-TYPE
43     SYNTAX      INTEGER {oneframeoffset(1),
44                        twoframesoffset(2),
45                        threeframesoffset(3)}
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49         "This object defines the OFDMA H-ARQ ACK delay for UL burst.
50         1 = one frame offset
51         2 = two frames offset
52         3 = three frames offset"
53
54     REFERENCE
55         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
56         ::= { wmanIf2SsOfdmaUplinkChannelEntry 26 }
57
58
59 wmanIf2SsOfdmaCQICHBandAMCTranaDelay OBJECT-TYPE
60     SYNTAX      INTEGER (0 .. 255)
61     UNITS       "Frame"
62     MAX-ACCESS  read-only
63
64
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This object defines the OFDMA CQICH band AMC transition
4          delay."
5      REFERENCE
6          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
7      ::= { wmanIf2SsOfdmaUplinkChannelEntry 27 }
8
9
10
11  wmanIf2SsOfdmaDownlinkChannelTable OBJECT-TYPE
12      SYNTAX      SEQUENCE OF WmanIf2SsOfdmaDownlinkChannelEntry
13      MAX-ACCESS  not-accessible
14      STATUS      current
15      DESCRIPTION
16          "This table contains DCD channel attributes, defining the
17          transmission characteristics of downlink channels"
18      REFERENCE
19          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
20      ::= { wmanIf2SsOfdmaPhy 2 }
21
22
23
24  wmanIf2SsOfdmaDownlinkChannelEntry OBJECT-TYPE
25      SYNTAX      WmanIf2SsOfdmaDownlinkChannelEntry
26      MAX-ACCESS  not-accessible
27      STATUS      current
28      DESCRIPTION
29          "This table provides one row for each downlink channel of
30          multi-sector BS, and is indexed by BS ifIndex. An entry in
31          this table exists for each ifEntry of BS with an ifType of
32          propBWA2Mp."
33      INDEX       { ifIndex }
34      ::= { wmanIf2SsOfdmaDownlinkChannelTable 1 }
35
36
37
38  WmanIf2SsOfdmaDownlinkChannelEntry ::= SEQUENCE {
39      wmanIf2SsOfdmaBsEIRP                INTEGER,
40      wmanIf2SsOfdmaChannelNumber          WmanIf2ChannelNumber,
41      wmanIf2SsOfdmaATTG                   INTEGER,
42      wmanIf2SsOfdmaARTG                   INTEGER,
43      wmanIf2SsOfdmaInitRngMaxRSS          INTEGER,
44      wmanIf2SsOfdmaDownlinkCenterFreq     Unsigned32,
45      wmanIf2SsOfdmaBsId                   WmanIf2BsIdType,
46      wmanIf2SsOfdmaMacVersion              WmanIf2MacVersion,
47      wmanIf2SsOfdmaFrameDurationCode      INTEGER,
48      wmanIf2SsOfdmaSizeCqichIdField       INTEGER,
49      wmanIf2SsOfdmaHARQAackDelayBurst     INTEGER}
50
51
52
53
54  wmanIf2SsOfdmaBsEIRP OBJECT-TYPE
55      SYNTAX      INTEGER (-32768..32767)
56      UNITS       "dBm"
57      MAX-ACCESS  read-only
58      STATUS      current
59      DESCRIPTION
60          "The EIRP is the equivalent isotropic radiated power of
61          the base station, which is computed for a simple
62          single-antenna transmitter."
63      REFERENCE
64
65

```



```

1         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
2         ::= { wmanIf2SsOfdmaDownlinkChannelEntry 1 }
3
4
5 wmanIf2SsOfdmaChannelNumber OBJECT-TYPE
6     SYNTAX      WmanIf2ChannelNumber
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "Downlink channel number as defined in 8.5. Used for
11        license-exempt operation only."
12
13     REFERENCE
14        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
15        ::= { wmanIf2SsOfdmaDownlinkChannelEntry 2 }
16
17
18 wmanIf2SsOfdmaTTG OBJECT-TYPE
19     SYNTAX      INTEGER (0..255)
20     MAX-ACCESS  read-only
21     STATUS      current
22     DESCRIPTION
23        "Transmit / Receive Transition Gap."
24
25     REFERENCE
26        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
27        ::= { wmanIf2SsOfdmaDownlinkChannelEntry 3 }
28
29
30 wmanIf2SsOfdmaRTG OBJECT-TYPE
31     SYNTAX      INTEGER (0..255)
32     MAX-ACCESS  read-only
33     STATUS      current
34     DESCRIPTION
35        "Receive / Transmit Transition Gap."
36
37     REFERENCE
38        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
39        ::= { wmanIf2SsOfdmaDownlinkChannelEntry 4 }
40
41
42 wmanIf2SsOfdmaInitRngMaxRSS OBJECT-TYPE
43     SYNTAX      INTEGER (-32768..32767)
44     UNITS       "dBm"
45     MAX-ACCESS  read-only
46     STATUS      current
47     DESCRIPTION
48        "Initial Ranging Max. equivalent isotropic received power
49        at BS Signed in units of 1 dBm."
50
51     REFERENCE
52        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
53        ::= { wmanIf2SsOfdmaDownlinkChannelEntry 5 }
54
55
56 wmanIf2SsOfdmaDownlinkCenterFreq OBJECT-TYPE
57     SYNTAX      Unsigned32
58     UNITS       "kHz"
59     MAX-ACCESS  read-only
60     STATUS      current
61     DESCRIPTION
62        "Downlink center frequency (kHz)."
63
64     REFERENCE
65

```

```

1         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
2         ::= { wmanIf2SsOfdmaDownlinkChannelEntry 6 }
3
4
5 wmanIf2SsOfdmaBsId OBJECT-TYPE
6     SYNTAX      WmanIf2BsIdType
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "Base station ID."
11
12    REFERENCE
13        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
14    ::= { wmanIf2SsOfdmaDownlinkChannelEntry 7 }
15
16
17 wmanIf2SsOfdmaMacVersion OBJECT-TYPE
18     SYNTAX      WmanIf2MacVersion
19     MAX-ACCESS  read-only
20     STATUS      current
21     DESCRIPTION
22        "This parameter specifies the version of 802.16 to which
23         the message originator conforms."
24
25    REFERENCE
26        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
27    ::= { wmanIf2SsOfdmaDownlinkChannelEntry 8 }
28
29
30 wmanIf2SsOfdmaFrameDurationCode OBJECT-TYPE
31     SYNTAX      INTEGER { aASGap(0),
32                          duration2ms(1),
33                          duration2dot5ms(2),
34                          duration4ms(3),
35                          duration5ms(4),
36                          duration8ms(5),
37                          duration10ms(6),
38                          duration12dot5ms(7),
39                          duration20ms(8) }
40
41     MAX-ACCESS  read-only
42     STATUS      current
43     DESCRIPTION
44        "The duration of the frame. The frame duration code values
45         are specified in Table 232 in IEEE Std 802.16-2004."
46
47    REFERENCE
48        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
49    ::= { wmanIf2SsOfdmaDownlinkChannelEntry 9 }
50
51
52
53 wmanIf2SsOfdmaSizeCqichIdField OBJECT-TYPE
54     SYNTAX      INTEGER { threebits(1),
55                          fourbits(2),
56                          fivebits(3),
57                          sixbits(4),
58                          sevenbits(5),
59                          eightbits(6),
60                          ninebits(7) }
61
62     MAX-ACCESS  read-only
63     STATUS      current
64     DESCRIPTION
65

```

```

1         "This object defines the size of CQICH ID field.
2         0 = Reserved
3         1 = 3 bits
4         2 = 4 bits
5         3 = 5 bits
6         4 = 6 bits
7         5 = 7 bits
8         6 = 8 bits
9         7 = 9 bits
10        8...255 = Reserved"
11
12    REFERENCE
13        "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
14    ::= { wmanIf2SsOfdmaDownlinkChannelEntry 10 }
15
16
17    wmanIf2SsOfdmaHARQAackDelayBurst OBJECT-TYPE
18        SYNTAX      INTEGER {oneframeoffset(1),
19                        twoframesoffset(2),
20                        threeframesoffset(3)}
21
22        MAX-ACCESS   read-only
23        STATUS       current
24        DESCRIPTION
25            "This object defines the OFDMA H-ARQ ACK delay for DL burst.
26            1 = one frame offset
27            2 = two frames offset
28            3 = three frames offset"
29
30        REFERENCE
31            "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
32        ::= { wmanIf2SsOfdmaDownlinkChannelEntry 11 }
33
34
35    wmanIf2SsOfdmaUcdBurstProfileTable OBJECT-TYPE
36        SYNTAX      SEQUENCE OF WmanIf2SsOfdmaUcdBurstProfileEntry
37        MAX-ACCESS   not-accessible
38        STATUS       current
39        DESCRIPTION
40            "This table contains UCD burst profiles for each uplink
41            channel"
42
43        REFERENCE
44            "Subclause 11.3.1.1, Table 288 and Table 357, in IEEE
45            Std 802.16-2004"
46        ::= { wmanIf2SsOfdmaPhy 3 }
47
48
49    wmanIf2SsOfdmaUcdBurstProfileEntry OBJECT-TYPE
50        SYNTAX      WmanIf2SsOfdmaUcdBurstProfileEntry
51        MAX-ACCESS   not-accessible
52        STATUS       current
53        DESCRIPTION
54            "This table provides one row for each UCD burst profile.
55            This table is double indexed. The primary index is an
56            ifIndex with an ifType of propBWApm2Mp. The secondary index
57            is wmanIf2SsOfdmaUiucIndex."
58
59        INDEX        { ifIndex, wmanIf2SsOfdmaUiucIndex }
60        ::= { wmanIf2SsOfdmaUcdBurstProfileTable 1 }
61
62
63    WmanIf2SsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
64
65

```

```

1      wmanIf2SsOfdmaUiucIndex          INTEGER,
2      wmanIf2SsOfdmaUcdFecCodeType      WmanIf2OfdmaFecCodeType,
3      wmanIf2SsOfdmaRangingDataRatio    INTEGER,
4      wmanIf2SsOfdmaNorCOverNOVERRIDE   OCTET STRING}
5
6
7  wmanIf2SsOfdmaUiucIndex OBJECT-TYPE
8      SYNTAX      INTEGER (1 .. 10)
9      MAX-ACCESS  read-only
10     STATUS      current
11     DESCRIPTION
12         "The Uplink Interval Usage Code indicates the uplink burst
13         profile in the UCD message, and is used along with ifIndex
14         to identify an entry in the
15         wmanIf2SsOfdmaUcdBurstProfileTable."
16     REFERENCE
17         "Subclause 8.4.5.4.1, in IEEE Std 802.16-2004"
18     ::= { wmanIf2SsOfdmaUcdBurstProfileEntry 1 }
19
20
21  wmanIf2SsOfdmaUcdFecCodeType OBJECT-TYPE
22     SYNTAX      WmanIf2OfdmaFecCodeType
23     MAX-ACCESS  read-only
24     STATUS      current
25     DESCRIPTION
26         "Uplink FEC code type and modulation type"
27     REFERENCE
28         "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
29     ::= { wmanIf2SsOfdmaUcdBurstProfileEntry 2 }
30
31
32  wmanIf2SsOfdmaRangingDataRatio OBJECT-TYPE
33     SYNTAX      INTEGER (0 .. 255)
34     MAX-ACCESS  read-only
35     STATUS      current
36     DESCRIPTION
37         "Reducing factor in units of 1 dB, between the power used
38         for this burst and power should be used for CDMA Ranging."
39     REFERENCE
40         "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
41     ::= { wmanIf2SsOfdmaUcdBurstProfileEntry 3 }
42
43
44  wmanIf2SsOfdmaNorCOverNOVERRIDE OBJECT-TYPE
45     SYNTAX OCTET STRING (SIZE (5))
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49         "This is a list of numbers, where each number is encoded by
50         one nibble, and interpreted as a signed integer. The nibbles
51         correspond in order to the list define by Table 334 in IEEE
52         Std 802.16-2004 starting from the second line, such that
53         the LS nibble of the first byte corresponds to the second
54         line in the table. The number encoded by each nibble
55         represents the difference in normalized C/N relative to the
56         previous line in the table"
57     REFERENCE
58         "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
59
60
61
62
63
64
65

```

```

1      ::= { wmanIf2SsOfdmaUcdBurstProfileEntry 4 }
2
3
4  wmanIf2SsOfdmaDcdBurstProfileTable OBJECT-TYPE
5      SYNTAX      SEQUENCE OF WmanIf2SsOfdmaDcdBurstProfileEntry
6      MAX-ACCESS  not-accessible
7      STATUS      current
8      DESCRIPTION
9          "This table provides one row for each DCD burst profile.
10         This table is double indexed. The primary index is an
11         ifIndex with an ifType of propBWA2Mp. The secondary index
12         is wmanIf2SsOfdmaDiucIndex."
13
14     ::= { wmanIf2SsOfdmaPhy 4 }
15
16
17  wmanIf2SsOfdmaDcdBurstProfileEntry OBJECT-TYPE
18      SYNTAX      WmanIf2SsOfdmaDcdBurstProfileEntry
19      MAX-ACCESS  not-accessible
20      STATUS      current
21      DESCRIPTION
22          "This table provides one row for each DCD burst profile,
23          and is double indexed. The primary index is an ifIndex
24          with an ifType of propBWA2Mp. The secondary index is
25          wmanIf2SsOfdmaDiucIndex."
26
27      INDEX      { ifIndex, wmanIf2SsOfdmaDiucIndex }
28
29     ::= { wmanIf2SsOfdmaDcdBurstProfileTable 1 }
30
31
32  WmanIf2SsOfdmaDcdBurstProfileEntry ::= SEQUENCE {
33      wmanIf2SsOfdmaDiucIndex          INTEGER,
34      wmanIf2SsOfdmaDownlinkFrequency  Unsigned32,
35      wmanIf2SsOfdmaDcdFecCodeType     WmanIf2OfdmaFecCodeType,
36      wmanIf2SsOfdmaDiucMandatoryExitThresh  INTEGER,
37      wmanIf2SsOfdmaDiucMinEntryThresh  INTEGER}
38
39
40  wmanIf2SsOfdmaDiucIndex OBJECT-TYPE
41      SYNTAX      INTEGER (0 .. 12)
42      MAX-ACCESS  read-only
43      STATUS      current
44      DESCRIPTION
45          "The Downlink Interval Usage Code indicates the downlink
46          burst profile in the DCD message, and is used
47          along with ifIndex to identify an entry in the
48          wmanIf2SsOfdmaDcdBurstProfileTable."
49
50      REFERENCE
51          "Subclause 8.4.5.3.1, in IEEE Std 802.16-2004"
52
53     ::= { wmanIf2SsOfdmaDcdBurstProfileEntry 1 }
54
55
56  wmanIf2SsOfdmaDownlinkFrequency OBJECT-TYPE
57      SYNTAX      Unsigned32
58      UNITS       "kHz"
59      MAX-ACCESS  read-only
60      STATUS      current
61      DESCRIPTION
62          "Downlink Frequency (kHz)."
```

```

63      REFERENCE
64          "Subclause 11.4.2, Table 359, in IEEE Std 802.16-2004"
```

```

1      ::= { wmanIf2SsOfdmaDcdBurstProfileEntry 2 }
2
3
4  wmanIf2SsOfdmaDcdFecCodeType OBJECT-TYPE
5      SYNTAX      WmanIf2OfdmaFecCodeType
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "Downlink FEC code type and modulation type"
10     REFERENCE
11         "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
12     ::= { wmanIf2SsOfdmaDcdBurstProfileEntry 3 }
13
14
15  wmanIf2SsOfdmaDiucMandatoryExitThresh OBJECT-TYPE
16      SYNTAX      INTEGER (0..255)
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20          "DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or
21          below where this DIUC can no longer be used and where this
22          change to a more robust DIUC is required in 0.25 dB units."
23     REFERENCE
24         "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
25     ::= { wmanIf2SsOfdmaDcdBurstProfileEntry 4 }
26
27
28  wmanIf2SsOfdmaDiucMinEntryThresh OBJECT-TYPE
29      SYNTAX      INTEGER (0..255)
30      MAX-ACCESS  read-only
31      STATUS      current
32      DESCRIPTION
33          "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
34          required to start using this DIUC when changing from a more
35          robust DIUC is required, in 0.25 dB units."
36     REFERENCE
37         "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
38     ::= { wmanIf2SsOfdmaDcdBurstProfileEntry 5 }
39
40
41  --
42  -- Common object group - containing common tables and objects to be
43  -- implemented in both Base Station and Subscriber Station
44  --
45  -- wmanIf2CmnPacketCs contain the Packet Convergence Sublayer objects
46  -- that are common to both Base Station and Subscriber Station
47  --
48  wmanIf2CmnPacketCs OBJECT IDENTIFIER ::= { wmanIf2CommonObjects 1 }
49
50
51  wmanIf2CmnClassifierRuleTable OBJECT-TYPE
52      SYNTAX      SEQUENCE OF WmanIf2CmnClassifierRuleEntry
53      MAX-ACCESS  not-accessible
54      STATUS      current
55      DESCRIPTION
56          "This table contains packet classifier rules associated
57          with service flows."
58     ::= { wmanIf2CmnPacketCs 1 }
59
60
61
62
63
64
65

```

```

1  wmanIf2CmnClassifierRuleEntry OBJECT-TYPE
2      SYNTAX      WmanIf2CmnClassifierRuleEntry
3      MAX-ACCESS  not-accessible
4      STATUS      current
5      DESCRIPTION
6          "This table provides one row for each packet classifier
7          rule, and is indexed by ifIndex, wmanIf2CmnCpsSfId, and
8          wmanIf2CmnClassifierRuleIndex. ifIndex is associated with
9          the BS sector. wmanIf2CmnCpsSfId identifies the service
10         flow, and wmanIf2CmnClassifierRuleIndex identifies the
11         packet classifier rule."
12     INDEX { ifIndex, wmanIf2CmnCpsSfId,
13             wmanIf2CmnClassifierRuleIndex }
14     ::= { wmanIf2CmnClassifierRuleTable 1 }
15
16 WmanIf2CmnClassifierRuleEntry ::= SEQUENCE {
17     wmanIf2CmnClassifierRuleIndex      Unsigned32,
18     wmanIf2CmnClassifierRulePriority    INTEGER,
19     wmanIf2CmnClassifierRuleIpTosLow   INTEGER,
20     wmanIf2CmnClassifierRuleIpTosHigh  INTEGER,
21     wmanIf2CmnClassifierRuleIpTosMask  INTEGER,
22     wmanIf2CmnClassifierRuleIpProtocol Integer32,
23     wmanIf2CmnClassifierRuleIpSourceAddr InetAddress,
24     wmanIf2CmnClassifierRuleIpSourceMask InetAddress,
25     wmanIf2CmnClassifierRuleIpDestAddr  InetAddress,
26     wmanIf2CmnClassifierRuleIpDestMask  InetAddress,
27     wmanIf2CmnClassifierRuleSourcePortStart Integer32,
28     wmanIf2CmnClassifierRuleSourcePortEnd Integer32,
29     wmanIf2CmnClassifierRuleDestPortStart Integer32,
30     wmanIf2CmnClassifierRuleDestPortEnd Integer32,
31     wmanIf2CmnClassifierRuleDestMacAddr MacAddress,
32     wmanIf2CmnClassifierRuleDestMacMask MacAddress,
33     wmanIf2CmnClassifierRuleSourceMacAddr MacAddress,
34     wmanIf2CmnClassifierRuleSourceMacMask MacAddress,
35     wmanIf2CmnClassifierRuleEnetProtocolType INTEGER,
36     wmanIf2CmnClassifierRuleEnetProtocol Integer32,
37     wmanIf2CmnClassifierRuleUserPriLow Integer32,
38     wmanIf2CmnClassifierRuleUserPriHigh Integer32,
39     wmanIf2CmnClassifierRuleVlanId      Integer32,
40     wmanIf2CmnClassifierRuleState        INTEGER,
41     wmanIf2CmnClassifierRulePkts         Counter64,
42     wmanIf2CmnClassifierRuleIpv6FlowLabel WmanIf2Ipv6FlowLabel,
43     wmanIf2CmnClassifierRuleBitMap       WmanIf2ClassifierBitMap
44 }
45
46 wmanIf2CmnClassifierRuleIndex OBJECT-TYPE
47     SYNTAX      Unsigned32 (1..4294967295)
48     MAX-ACCESS  not-accessible
49     STATUS      current
50     DESCRIPTION
51         "An index is assigned to each classifier in the classifiers
52         table"
53     ::= { wmanIf2CmnClassifierRuleEntry 1 }
54
55
56
57
58
59
60
61
62
63
64
65

```

```

1  wmanIf2CmnClassifierRulePriority OBJECT-TYPE
2      SYNTAX          INTEGER (0..255)
3      MAX-ACCESS      read-only
4      STATUS          current
5      DESCRIPTION
6          "The value specifies the order of evaluation of the
7           classifiers. The higher the value the higher the
8           priority. The value of 0 is used as default in
9           provisioned service flows classifiers. The default
10          value of 64 is used for dynamic service flow classifiers.
11          If the referenced parameter is not present in a classifier
12          , this object reports the default value as defined above"
13      REFERENCE
14          "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
15      DEFVAL          { 0 }
16      ::= { wmanIf2CmnClassifierRuleEntry 2 }
17
18  wmanIf2CmnClassifierRuleIpTosLow OBJECT-TYPE
19      SYNTAX          INTEGER (0 .. 255)
20      MAX-ACCESS      read-only
21      STATUS          current
22      DESCRIPTION
23          "The low value of a range of TOS byte values. If the
24          referenced parameter is not present in a classifier, this
25          object reports the value of 0."
26      REFERENCE
27          "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
28      ::= { wmanIf2CmnClassifierRuleEntry 3 }
29
30  wmanIf2CmnClassifierRuleIpTosHigh OBJECT-TYPE
31      SYNTAX          INTEGER (0 .. 255)
32      MAX-ACCESS      read-only
33      STATUS          current
34      DESCRIPTION
35          "The 8-bit high value of a range of TOS byte values.
36          If the referenced parameter is not present in a classifier
37          , this object reports the value of 0."
38      REFERENCE
39          "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
40      ::= { wmanIf2CmnClassifierRuleEntry 4 }
41
42  wmanIf2CmnClassifierRuleIpTosMask OBJECT-TYPE
43      SYNTAX          INTEGER (0 .. 255)
44      MAX-ACCESS      read-only
45      STATUS          current
46      DESCRIPTION
47          "The mask value is bitwise ANDed with TOS byte in an IP
48          packet and this value is used for the range checking of
49          TosLow and TosHigh. If the referenced parameter is not
50          present in a classifier, this object reports the value
51          of 0."
52      REFERENCE
53          "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
54      ::= { wmanIf2CmnClassifierRuleEntry 5 }
55
56
57
58
59
60
61
62
63
64
65

```



```

1
2 wmanIf2CmnClassifierRuleIpProtocol OBJECT-TYPE
3     SYNTAX      Integer32 (0..255)
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "This object indicates the value of the IP Protocol field
8         required for IP packets to match this rule. If the
9         referenced parameter is not present in a classifier, this
10        object reports the value of 0."
11
12 REFERENCE
13     "Subclause 11.13.19.3.4.3 in IEEE Std 802.16-2004"
14 ::= { wmanIf2CmnClassifierRuleEntry 6 }
15
16 wmanIf2CmnClassifierRuleIpSourceAddr OBJECT-TYPE
17     SYNTAX      InetAddress
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21         "This object specifies the value of the IP Source Address
22         required for packets to match this rule. An IP packet
23         matches the rule when the packet ip source address bitwise
24         ANDed with the wmanIf2CmnClassifierRuleIpSourceMask value
25         equals the wmanIf2CmnClassifierRuleIpSourceAddr value.
26         If the referenced parameter is not present in a classifier
27         , this object reports the value of 0.0.0.0."
28
29 REFERENCE
30     "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
31 ::= { wmanIf2CmnClassifierRuleEntry 7 }
32
33 wmanIf2CmnClassifierRuleIpSourceMask OBJECT-TYPE
34     SYNTAX      InetAddress
35     MAX-ACCESS  read-only
36     STATUS      current
37     DESCRIPTION
38         "This object specifies which bits of a packet's IP Source
39         Address that are compared to match this rule. An IP packet
40         matches the rule when the packet source address bitwise
41         ANDed with the
42         wmanIf2CmnClassifierRuleIpSourceMask value equals the
43         wmanIf2CmnClassifierRuleIpSourceAddr value.
44         If the referenced parameter is not present in a classifier
45         , this object reports the value of 0.0.0.0."
46
47 REFERENCE
48     "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
49 ::= { wmanIf2CmnClassifierRuleEntry 8 }
50
51 wmanIf2CmnClassifierRuleIpDestAddr OBJECT-TYPE
52     SYNTAX      InetAddress
53     MAX-ACCESS  read-only
54     STATUS      current
55     DESCRIPTION
56         "This object specifies the value of the IP Destination
57         Address required for packets to match this rule. An IP
58

```

packet matches the rule when the packet IP destination address bitwise ANDed with the wmanIf2CmnClassifierRuleIpDestMask value equals the wmanIf2CmnClassifierRuleIpDestAddr value. If the referenced parameter is not present in a classifier, this object reports the value of 0.0.0.0."

REFERENCE

"Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnClassifierRuleEntry 9 }

wmanIf2CmnClassifierRuleIpDestMask OBJECT-TYPE

SYNTAX InetAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object specifies which bits of a packet's IP Destination Address that are compared to match this rule. An IP packet matches the rule when the packet destination address bitwise ANDed with the wmanIf2CmnClassifierRuleIpDestMask value equals the wmanIf2CmnClassifierRuleIpDestAddr value. If the referenced parameter is not present in a classifier, this object reports the value of 0.0.0.0."

REFERENCE

"Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnClassifierRuleEntry 10 }

wmanIf2CmnClassifierRuleSourcePortStart OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object specifies the low end inclusive range of TCP/UDP source port numbers to which a packet is compared. This object is irrelevant for non-TCP/UDP IP packets. If the referenced parameter is not present in a classifier, this object reports the value of 0."

REFERENCE

"Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnClassifierRuleEntry 11 }

wmanIf2CmnClassifierRuleSourcePortEnd OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object specifies the high end inclusive range of TCP/UDP source port numbers to which a packet is compared. This object is irrelevant for non-TCP/UDP IP packets. If the referenced parameter is not present in a classifier, this object reports the value of 65535."

REFERENCE

"Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnClassifierRuleEntry 12 }

```

1
2 wmanIf2CmnClassifierRuleDestPortStart OBJECT-TYPE
3     SYNTAX      Integer32 (0..65535)
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "This object specifies the low end inclusive range of
8         TCP/UDP destination port numbers to which a packet is
9         compared. If the referenced parameter is not present
10        in a classifier, this object reports the value of 0."
11
12 REFERENCE
13     "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
14 ::= { wmanIf2CmnClassifierRuleEntry 13 }
15
16 wmanIf2CmnClassifierRuleDestPortEnd OBJECT-TYPE
17     SYNTAX      Integer32 (0..65535)
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21         "This object specifies the high end inclusive range of
22         TCP/UDP destination port numbers to which a packet is
23         compared. If the referenced parameter is not present
24         in a classifier, this object reports the value of
25         65535."
26
27 REFERENCE
28     "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
29 ::= { wmanIf2CmnClassifierRuleEntry 14 }
30
31 wmanIf2CmnClassifierRuleDestMacAddr OBJECT-TYPE
32     SYNTAX      MacAddress
33     MAX-ACCESS  read-only
34     STATUS      current
35     DESCRIPTION
36         "An Ethernet packet matches an entry when its destination
37         MAC address bitwise ANDed with
38         wmanIf2CmnClassifierRuleDestMacMask equals the value of
39         wmanIf2CmnClassifierRuleDestMacAddr. If the referenced
40         parameter is not present in a classifier, this object
41         reports the value of '000000000000'H."
42
43 REFERENCE
44     "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
45 ::= { wmanIf2CmnClassifierRuleEntry 15 }
46
47 wmanIf2CmnClassifierRuleDestMacMask OBJECT-TYPE
48     SYNTAX      MacAddress
49     MAX-ACCESS  read-only
50     STATUS      current
51     DESCRIPTION
52         "An Ethernet packet matches an entry when its destination
53         MAC address bitwise ANDed with
54         wmanIf2CmnClassifierRuleDestMacMask equals the value of
55         wmanIf2CmnClassifierRuleDestMacAddr. If the referenced
56         parameter is not present in a classifier, this object
57         reports the value of '000000000000'H."
58
59
60
61
62
63
64
65

```

REFERENCE

```

        "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnClassifierRuleEntry 16 }

```

```
::= { wmanIf2CmnClassifierRuleEntry 16 }
```

wmanIf2CmnClassifierRuleSourceMacAddr OBJECT-TYPE

SYNTAX MacAddress

MAX-ACCESS read-only

STATUS	current
--------	---------

DESCRIPTION

"An Ethernet packet matches this entry when its source MAC address bitwise ANDed with wmanIf2CmnClassifierRuleSourceMacMask equals the value of wmanIf2CmnClassifierRuleSourceMacAddr. If the referenced parameter is not present in a classifier, this object reports the value of '000000000000'H."

REFERENCE

```

        "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnClassifierRuleEntry 17 }

```

```
::= { wmanIf2CmnClassifierRuleEntry 17 }
```

wmanIf2CmnClassifierRuleSourceMacMask OBJECT-TYPE

SYNTAX MacAddress

MAX-ACCESS read-only

STATUS	current
--------	---------

DESCRIPTION

```
"An Ethernet packet matches an entry when its destination
MAC address bitwise ANDed with
wmanIf2CmnClassifierRuleSourceMacMask equals the value of
wmanIf2CmnClassifierRuleSourceMacAddr. If the referenced
parameter is not present in a classifier, this object
reports the value of '000000000000'H."
```

REFERENCE

```

        "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnClassifierRuleEntry 18 }

```

```
::= { wmanIf2CmnClassifierRuleEntry 18 }
```

```
wmanIf2CmnClassifierRuleEnetProtocolType OBJECT-TYPE
```

```
SYNTAX      INTEGER {none(0),
                   ethernet(1),
                   dsap(2) }
```

ethertype(1),

```
dsap(2) }
```

MAX-ACCESS read-only

```
STATUS      current
```

DESCRIPTION

"This object indicates the format of the layer 3 protocol id in the Ethernet packet. A value of none(0) means that the rule does not use the layer 3 protocol type as a matching criteria. A value of ethertype(1) means that the rule applies only to frames which contains an EtherType value. Ethertype values are contained in packets using the Dec-Intel-Xerox (DIX) encapsulation or the RFC1042 Sub-Network Access Protocol (SNAP) encapsulation formats. A value of dsap(2) means that the rule applies only to frames using the IEEE802.3 encapsulation format with a Destination Service Access Point (DSAP) other than 0xAA (which is reserved for SNAP). If the Ethernet frame contains an 802.1P/Q Tag header (i.e. EtherType 0x8100),

```

1         this object applies to the embedded EtherType field within
2         the 802.1P/Q header. If the referenced parameter is not
3         present in a classifier, this object reports the value of
4         0."
5
6     REFERENCE
7         "Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
8     ::= { wmanIf2CmnClassifierRuleEntry 19 }
9
10
11 wmanIf2CmnClassifierRuleEnetProtocol OBJECT-TYPE
12     SYNTAX      Integer32 (0..65535)
13     MAX-ACCESS  read-only
14     STATUS      current
15     DESCRIPTION
16         "If wmanIf2CmnClassifierRuleEnetProtocolType is none(0),
17         this object is ignored when considering whether a packet
18         matches the current rule.
19         If wmanIf2CmnClassifierRuleEnetProtocolType is ethertype(1)
20         ,this object gives the 16-bit value of the EtherType that
21         the packet must match in order to match the rule.
22         If wmanIf2CmnClassifierRuleEnetProtocolType is dsap(2), the
23         lower 8 bits of this object's value must match the DSAP
24         byte of the packet in order to match the rule.
25         If the Ethernet frame contains an 802.1P/Q Tag header
26         (i.e. EtherType 0x8100), this object applies to the
27         embedded EtherType field within the 802.1P/Q header.
28         If the referenced parameter is not present in the
29         classifier, the value of this object is reported as 0."
30
31     REFERENCE
32         "Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
33     ::= { wmanIf2CmnClassifierRuleEntry 20 }
34
35
36 wmanIf2CmnClassifierRuleUserPriLow OBJECT-TYPE
37     SYNTAX      Integer32 (0..7)
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41         "This object applies only to Ethernet frames using the
42         802.1P/Q tag header (indicated with EtherType 0x8100).
43         Such frames include a 16-bit Tag that contains a 3 bit
44         Priority field and a 12 bit VLAN number.
45         Tagged Ethernet packets must have a 3-bit Priority field
46         within the range of wmanIf2CmnClassifierRulePriLow and
47         wmanIf2CmnClassifierRulePriHigh in order to match this
48         rule.
49         If the referenced parameter is not present in the
50         classifier, the value of this object is reported as 0."
51
52     REFERENCE
53         "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
54     ::= { wmanIf2CmnClassifierRuleEntry 21 }
55
56
57 wmanIf2CmnClassifierRuleUserPriHigh OBJECT-TYPE
58     SYNTAX      Integer32 (0..7)
59     MAX-ACCESS  read-only
60     STATUS      current
61
62
63
64
65

```

DESCRIPTION

"This object applies only to Ethernet frames using the 802.1P/Q tag header (indicated with EtherType 0x8100). Such frames include a 16-bit Tag that contains a 3 bit Priority field and a 12 bit VLAN number. Tagged Ethernet packets must have a 3-bit Priority field within the range of wmanIf2CmnClassifierRulePriLow and wmanIf2CmnClassifierRulePriHigh in order to match this rule. If the referenced parameter is not present in the classifier, the value of this object is reported as 7."

REFERENCE

"Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"

::= { wmanIf2CmnClassifierRuleEntry 22 }

wmanIf2CmnClassifierRuleVlanId OBJECT-TYPE

SYNTAX Integer32 (0..4095)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object applies only to Ethernet frames using the 802.1P/Q tag header. If this object's value is nonzero, tagged packets must have a VLAN Identifier that matches the value in order to match the rule. Only the least significant 12 bits of this object's value are valid. If the referenced parameter is not present in the classifier, the value of this object is reported as 0."

REFERENCE

"Subclause 11.13.19.3.4.12 in IEEE Std 802.16-2004"

::= { wmanIf2CmnClassifierRuleEntry 23 }

wmanIf2CmnClassifierRuleState OBJECT-TYPE

SYNTAX INTEGER {active(1),
inactive(2)}

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"This object indicates whether or not the classifier is enabled to classify packets to a Service Flow. If the referenced parameter is not present in the classifier, the value of this object is reported as active(1)."

::= { wmanIf2CmnClassifierRuleEntry 24 }

wmanIf2CmnClassifierRulePkts OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object counts the number of packets that have been classified using this entry."

::= { wmanIf2CmnClassifierRuleEntry 25 }

```

1
2 wmanIf2CmnClassifierRuleIpv6FlowLabel OBJECT-TYPE
3     SYNTAX      WmanIf2Ipv6FlowLabel
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "The value of this field specifies the matching values for
8         the IPv6 Flow label field."
9     ::= { wmanIf2CmnClassifierRuleEntry 26 }
10
11
12
13 wmanIf2CmnClassifierRuleBitMap OBJECT-TYPE
14     SYNTAX      WmanIf2ClassifierBitMap
15     MAX-ACCESS  read-only
16     STATUS      current
17     DESCRIPTION
18         "This object indicates which parameter encodings were
19         actually present in the entry. A bit set to '1' indicates
20         the corresponding classifier encoding is present, and '0'
21         means otherwise"
22     ::= { wmanIf2CmnClassifierRuleEntry 27 }
23
24
25
26 wmanIf2CmnPhsRuleTable OBJECT-TYPE
27     SYNTAX      SEQUENCE OF WmanIf2CmnPhsRuleEntry
28     MAX-ACCESS  not-accessible
29     STATUS      current
30     DESCRIPTION
31         "This table contains PHS rule dictionary entries. Each
32         entry contains the data of the header to be suppressed
33         along with its identification - PHSI. The classifier
34         uniquely maps packets to its associated PHS Rule. The
35         receiving entity uses the CID and the PHSI to restore the
36         PHSF. Once a PHSF has been assigned to a PHSI, it shall
37         not be changed. To change the value of a PHSF on a
38         service flow, a new PHS rule shall be defined, the old
39         rule is removed from the service flow, and the new rule
40         is added. When a classifier is deleted, any associated
41         PHS rule shall also be deleted."
42     REFERENCE
43         "Subclause 5.2.3 in IEEE Std 802.16-2004"
44     ::= { wmanIf2CmnPacketCs 2 }
45
46
47
48
49
50 wmanIf2CmnPhsRuleEntry OBJECT-TYPE
51     SYNTAX      WmanIf2CmnPhsRuleEntry
52     MAX-ACCESS  not-accessible
53     STATUS      current
54     DESCRIPTION
55         "This table provides one row for each PHS rule created
56         dynamically by the BS and SS on a given service flow. The
57         PHS rule is defined by the pair (PHSS, PHSM) for each
58         distinct header data. It is indexed by IfIndex,
59         wmanIf2CmnCpsSfId, and wmanIf2CmnPhsIndex. The table is
60         read-only for NMS. "
61     INDEX      { ifIndex, wmanIf2CmnCpsSfId,
62                  wmanIf2CmnPhsRulePhsIndex }
63
64
65

```

```

1      ::= { wmanIf2CmnPhsRuleTable 1 }
2
3
4  WmanIf2CmnPhsRuleEntry ::= SEQUENCE {
5      wmanIf2CmnPhsRulePhsIndex      INTEGER,
6      wmanIf2CmnPhsRulePhsField      OCTET STRING,
7      wmanIf2CmnPhsRulePhsMask      OCTET STRING,
8      wmanIf2CmnPhsRulePhsSize      Integer32,
9      wmanIf2CmnPhsRulePhsVerify    WmanIf2PhsRuleVerify}
10
11
12  wmanIf2CmnPhsRulePhsIndex OBJECT-TYPE
13      SYNTAX      INTEGER (1..255)
14      MAX-ACCESS  not-accessible
15      STATUS      current
16      DESCRIPTION
17          "The PHSI (PHS Index) has a value between 1 and 255, which
18           uniquely references the suppressed byte string. The index
19           is unique per service flow. The uplink and downlink PHSI
20           values are independent of each other."
21      REFERENCE
22          "Subclause 11.13.19.3.7.1 in IEEE Std 802.16-2004"
23      ::= { wmanIf2CmnPhsRuleEntry 1 }
24
25
26
27  wmanIf2CmnPhsRulePhsField OBJECT-TYPE
28      SYNTAX      OCTET STRING (SIZE(0..65535))
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32          "The PHSF (PHS Field) is a string of bytes containing the
33           header information to be suppressed by the sending CS and
34           reconstructed by the receiving CS. The most significant
35           byte of the string corresponds to the first byte of the
36           CS-SDU."
37      REFERENCE
38          "Subclause 11.13.19.3.7.2 in IEEE Std 802.16-2004"
39      ::= { wmanIf2CmnPhsRuleEntry 2 }
40
41
42
43
44  wmanIf2CmnPhsRulePhsMask OBJECT-TYPE
45      SYNTAX      OCTET STRING (SIZE(0..65535))
46      MAX-ACCESS  read-only
47      STATUS      current
48      DESCRIPTION
49          "The PHSM An 8-bit mask that indicates which bytes in the
50           PHS Field (PHSF) to suppress and which bytes to not
51           suppress. The PHSM allows fields, such as sequence numbers
52           or checksums (which vary in value), to be excluded from
53           suppression with the constant bytes around them suppressed.
54           It is encoded as follows:
55           bit 0:
56               0 = don't suppress the 1st byte of the suppression field
57               1 = suppress first byte of the suppression field
58           bit 1:
59               0 = don't suppress the 2nd byte of the suppression field
60               1 = suppress second byte of the suppression field
61           bit x:
62
63
64
65

```



```

1           0 = don't suppress the (x+1) byte of the suppression
2             field
3           1 = suppress (x+1) byte of the suppression field
4             where the length of the octet string is ceiling
5             (wmanIf2CmnPhsRulePhsSize/8)."
6
7   REFERENCE
8       "Subclause 11.13.19.3.7.3 in IEEE Std 802.16-2004"
9   ::= { wmanIf2CmnPhsRuleEntry 3 }
10
11
12   wmanIf2CmnPhsRulePhsSize OBJECT-TYPE
13       SYNTAX      Integer32 (0..255)
14       UNITS       "byte"
15       MAX-ACCESS  read-only
16       STATUS      current
17       DESCRIPTION
18           "The value of this field - PHSS is the total number of bytes
19             in the header to be suppressed and then restored in a
20             service flow that uses PHS."
21
22   REFERENCE
23       "Subclause 11.13.19.3.7.4 in IEEE Std 802.16-2004"
24   DEFVAL      { 0 }
25   ::= { wmanIf2CmnPhsRuleEntry 4 }
26
27
28   wmanIf2CmnPhsRulePhsVerify OBJECT-TYPE
29       SYNTAX      WmanIf2PhsRuleVerify
30       MAX-ACCESS  read-only
31       STATUS      current
32       DESCRIPTION
33           "The value of this field indicates to the sending entity
34             whether or not the packet header contents are to be
35             verified prior to performing suppression."
36
37   DEFVAL      { phsVerifyEnable }
38   ::= { wmanIf2CmnPhsRuleEntry 5 }
39
40
41   --
42   -- wmanIf2CmnCps contain the Common Part Sublayer objects that are
43   -- common to both Base Station and Subscriber Station
44   --
45
46   wmanIf2CmnCps OBJECT IDENTIFIER ::= { wmanIf2CommonObjects 2 }
47
48
49   wmanIf2CmnCpsServiceFlowTable OBJECT-TYPE
50       SYNTAX      SEQUENCE OF WmanIf2CmnCpsServiceFlowEntry
51       MAX-ACCESS  not-accessible
52       STATUS      current
53       DESCRIPTION
54           "This table contains Service Flow managed objects that
55             are common in BS and SS."
56
57   ::= { wmanIf2CmnCps 1 }
58
59
60   wmanIf2CmnCpsServiceFlowEntry OBJECT-TYPE
61       SYNTAX      WmanIf2CmnCpsServiceFlowEntry
62       MAX-ACCESS  not-accessible
63       STATUS      current
64       DESCRIPTION
65

```

```

1      "This table provides one row for each created service
2      flow for a given MacAddress, and is indexed by ifIndex,
3      wmanIf2CmnCpsCpsSfMacAddress, and wmanIf2CmnCpsSfId.
4      IfIndex is associated with the BS sector."
5  INDEX      { ifIndex, wmanIf2CmnCpsSfMacAddress,
6              wmanIf2CmnCpsSfId }
7
8  ::= { wmanIf2CmnCpsServiceFlowTable 1 }
9
10
11  WmanIf2CmnCpsServiceFlowEntry ::= SEQUENCE {
12      wmanIf2CmnCpsSfMacAddress      MacAddress,
13      wmanIf2CmnCpsSfId              Unsigned32,
14      wmanIf2CmnCpsSfCid             WmanIf2CidType,
15      wmanIf2CmnCpsSfDirection      INTEGER,
16      wmanIf2CmnCpsSfState           WmanIf2SfState,
17      wmanIf2CmnCpsTrafficPriority    INTEGER,
18      wmanIf2CmnCpsMaxSustainedRate  Unsigned32,
19      wmanIf2CmnCpsMaxTrafficBurst   Unsigned32,
20      wmanIf2CmnCpsMinReservedRate   Unsigned32,
21      wmanIf2CmnCpsToleratedJitter   Unsigned32,
22      wmanIf2CmnCpsMaxLatency        Unsigned32,
23      wmanIf2CmnCpsFixedVsVariableSduInd  INTEGER,
24      wmanIf2CmnCpsSduSize           Unsigned32,
25      wmanIf2CmnCpsSfSchedulingType   WmanIf2SfSchedulingType,
26      wmanIf2CmnCpsArqEnable         TruthValue,
27      wmanIf2CmnCpsArqWindowSize     INTEGER,
28      wmanIf2CmnCpsArqBlockLifetime  INTEGER,
29      wmanIf2CmnCpsArqSyncLossTimeout  INTEGER,
30      wmanIf2CmnCpsArqDeliverInOrder  TruthValue,
31      wmanIf2CmnCpsArqRxPurgeTimeout  INTEGER,
32      wmanIf2CmnCpsArqBlockSize      INTEGER,
33      wmanIf2CmnCpsMinRsvdTolerableRate  Unsigned32,
34      wmanIf2CmnCpsReqTxPolicy        BITS,
35      wmanIf2CmnSfCsSpecification     WmanIf2CsSpecification,
36      wmanIf2CmnCpsTargetSaid         INTEGER}
37
38
39  wmanIf2CmnCpsSfMacAddress OBJECT-TYPE
40      SYNTAX      MacAddress
41      MAX-ACCESS  not-accessible
42      STATUS      current
43      DESCRIPTION
44          "When this table is implemented on the basestation, this
45          object contains the SS Mac address, the reported service
46          flow was created for. On the SS, the value returned is
47          the SS's own Mac address."
48      ::= { wmanIf2CmnCpsServiceFlowEntry 1 }
49
50
51  wmanIf2CmnCpsSfId OBJECT-TYPE
52      SYNTAX      Unsigned32 ( 1 .. 4294967295)
53      MAX-ACCESS  read-only
54      STATUS      current
55      DESCRIPTION
56          "A 32 bit quantity that uniquely identifies a service flow
57          to both the subscriber station and base station (BS)."
```

```

58      ::= { wmanIf2CmnCpsServiceFlowEntry 2 }
```

```

1
2 wmanIf2CmnCpsSfCid OBJECT-TYPE
3     SYNTAX      WmanIf2CidType
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "A 16 bit channel identifier to identify the connection
8         being created by DSA."
9     ::= { wmanIf2CmnCpsServiceFlowEntry 3 }
10
11
12
13 wmanIf2CmnCpsSfDirection OBJECT-TYPE
14     SYNTAX      INTEGER {downstream(1),
15                        upstream(2)}
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "An attribute indicating the service flow is downstream or
20         upstream."
21     ::= { wmanIf2CmnCpsServiceFlowEntry 4 }
22
23
24
25 wmanIf2CmnCpsSfState OBJECT-TYPE
26     SYNTAX      WmanIf2SfState
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30         "wmanIf2CmnCpsSfState indicates the service flow state:
31         Authorized (1), Admitted (2), and Active (3) service
32         flow state."
33     REFERENCE
34         "Subclause 6.3.14.6, in IEEE Std 802.16-2004"
35     ::= { wmanIf2CmnCpsServiceFlowEntry 5 }
36
37
38
39 wmanIf2CmnCpsTrafficPriority OBJECT-TYPE
40     SYNTAX      INTEGER (0 .. 7)
41     MAX-ACCESS  read-only
42     STATUS      current
43     DESCRIPTION
44         "The value of this parameter specifies the priority
45         assigned to a service flow. For uplink service flows,
46         the BS should use this parameter when determining
47         precedence in request service and grant generation,
48         and the SS shall preferentially select contention
49         Request opportunities for Priority Request CIDs
50         based on this priority"
51     REFERENCE
52         "Subclause 11.13.5 in IEEE Std 802.16-2004"
53     ::= { wmanIf2CmnCpsServiceFlowEntry 6 }
54
55
56
57
58
59 wmanIf2CmnCpsMaxSustainedRate OBJECT-TYPE
60     SYNTAX      Unsigned32
61     UNITS       "b/s"
62     MAX-ACCESS  read-only
63     STATUS      current
64     DESCRIPTION
65

```

```

1         "This parameter defines the peak information rate
2         of the service. The rate is expressed in bits per
3         second and pertains to the SDUs at the input to
4         the system."
5
6     REFERENCE
7         "Subclause 11.13.6 in IEEE Std 802.16-2004"
8     ::= { wmanIf2CmnCpsServiceFlowEntry 7 }
9
10
11 wmanIf2CmnCpsMaxTrafficBurst OBJECT-TYPE
12     SYNTAX      Unsigned32
13     UNITS       "byte"
14     MAX-ACCESS  read-only
15     STATUS      current
16     DESCRIPTION
17         "This parameter defines the maximum burst size that
18         must be accommodated for the service."
19
20     REFERENCE
21         "Subclause 11.13.7 in IEEE Std 802.16-2004"
22     ::= { wmanIf2CmnCpsServiceFlowEntry 8 }
23
24
25 wmanIf2CmnCpsMinReservedRate OBJECT-TYPE
26     SYNTAX      Unsigned32
27     UNITS       "byte"
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31         "This parameter specifies the minimum rate reserved
32         for this service flow."
33
34     REFERENCE
35         "Subclause 11.13.8 in IEEE Std 802.16-2004"
36     ::= { wmanIf2CmnCpsServiceFlowEntry 9 }
37
38
39 wmanIf2CmnCpsToleratedJitter OBJECT-TYPE
40     SYNTAX      Unsigned32
41     UNITS       "millisecond"
42     MAX-ACCESS  read-only
43     STATUS      current
44     DESCRIPTION
45         "This parameter defines the Maximum delay
46         variation (jitter) for the connection."
47
48     REFERENCE
49         "Subclause 11.13.13 in IEEE Std 802.16-2004"
50     ::= { wmanIf2CmnCpsServiceFlowEntry 10 }
51
52
53
54 wmanIf2CmnCpsMaxLatency OBJECT-TYPE
55     SYNTAX      Unsigned32
56     UNITS       "millisecond"
57     MAX-ACCESS  read-only
58     STATUS      current
59     DESCRIPTION
60         "The value of this parameter specifies the maximum
61         latency between the reception of a packet by the BS
62         or SS on its network interface and the forwarding
63         of the packet to its RF Interface."
64
65

```

REFERENCE

"Subclause 11.13.14 in IEEE Std 802.16-2004"
 ::= { wmanIf2CmnCpsServiceFlowEntry 11 }

wmanIf2CmnCpsFixedVsVariableSduInd OBJECT-TYPE

SYNTAX INTEGER {variableLength(0),
 fixedLength(1)}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of this parameter specifies whether the SDUs on the service flow are variable-length (0) or fixed-length (1). The parameter is used only if packing is on for the service flow. The default value is 0, i.e., variable-length SDUs."

REFERENCE

"Subclause 11.13.15 in IEEE Std 802.16-2004"
 DEFVAL { variableLength }
 ::= { wmanIf2CmnCpsServiceFlowEntry 12 }

wmanIf2CmnCpsSduSize OBJECT-TYPE

SYNTAX Unsigned32

UNITS "byte"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of this parameter specifies the length of the SDU for a fixed-length SDU service flow. This parameter is used only if packing is on and the service flow is indicated as carrying fixed-length SDUs. The default value is 49 bytes, i.e., VC-switched ATM cells with PHS. The parameter is relevant for both ATM and Packet Convergence Sublayers."

REFERENCE

"Subclause 11.13.16 in IEEE Std 802.16-2004"
 DEFVAL { 49 }
 ::= { wmanIf2CmnCpsServiceFlowEntry 13 }

wmanIf2CmnCpsSfsSchedulingType OBJECT-TYPE

SYNTAX WmanIf2SfsSchedulingType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Specifies the upstream scheduling service used for upstream service flow. If the referenced parameter is not present in the corresponding 802.16 QoS Parameter Set of an upstream service flow, the default value of this object is bestEffort(2)."

REFERENCE

"Subclause 11.13.11 in IEEE Std 802.16-2004"
 DEFVAL { bestEffort }
 ::= { wmanIf2CmnCpsServiceFlowEntry 14 }

wmanIf2CmnCpsArqEnable OBJECT-TYPE

```

1      SYNTAX      TruthValue
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "True(1) ARQ enabling is requested for the connection."
6      ::= { wmanIf2CmnCpsServiceFlowEntry 15 }
7
8
9
10     wmanIf2CmnCpsArqWindowSize OBJECT-TYPE
11         SYNTAX      INTEGER (1..1024)
12         MAX-ACCESS  read-only
13         STATUS      current
14         DESCRIPTION
15             "Indicates the maximum number of unacknowledged
16             fragments at any time."
17         ::= { wmanIf2CmnCpsServiceFlowEntry 16 }
18
19
20     wmanIf2CmnCpsArqBlockLifetime OBJECT-TYPE
21         SYNTAX      INTEGER (0 .. 65535)
22         UNITS        "10 us"
23         MAX-ACCESS  read-only
24         STATUS      current
25         DESCRIPTION
26             "The maximum time interval an ARQ fragment will be
27             managed by the transmitter ARQ machine, once
28             initial transmission of the fragment has occurred.
29             If transmission or retransmission of the fragment
30             is not acknowledged by the receiver before the
31             time limit is reached, the fragment is discarded.
32             A value of 0 means Infinite."
33         ::= { wmanIf2CmnCpsServiceFlowEntry 17 }
34
35
36
37
38     wmanIf2CmnCpsArqSyncLossTimeout OBJECT-TYPE
39         SYNTAX      INTEGER (0 .. 65535 )
40         UNITS        "10 us"
41         MAX-ACCESS  read-only
42         STATUS      current
43         DESCRIPTION
44             "The maximum interval before declaring a loss
45             of synchronization of the sender and receiver
46             state machines. A value of 0 means Infinite."
47         ::= { wmanIf2CmnCpsServiceFlowEntry 18 }
48
49
50
51
52     wmanIf2CmnCpsArqDeliverInOrder OBJECT-TYPE
53         SYNTAX      TruthValue
54         MAX-ACCESS  read-only
55         STATUS      current
56         DESCRIPTION
57             "Indicates whether or not data is to be delivered
58             by the receiving MAC to its client application
59             in the order in which data was handed off to the
60             originating MAC."
61         ::= { wmanIf2CmnCpsServiceFlowEntry 19 }
62
63
64
65     wmanIf2CmnCpsArqRxPurgeTimeout OBJECT-TYPE

```

```

1      SYNTAX      INTEGER (0 .. 65535)
2      UNITS       "10 us"
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "Indicates the time interval the ARQ window is advanced
7              after a fragment is received. A value of 0 means
8              Infinite."
9      ::= { wmanIf2CmnCpsServiceFlowEntry 20 }
10
11
12
13  wmanIf2CmnCpsArqBlockSize OBJECT-TYPE
14      SYNTAX      INTEGER (1..2040)
15      UNITS       "byte"
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "This value of this parameter specifies the size of an
20              ARQ block. This parameter shall be established by
21              negotiation during the connection creation dialog."
22      REFERENCE
23          "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
24      ::= { wmanIf2CmnCpsServiceFlowEntry 21 }
25
26
27
28
29  wmanIf2CmnCpsMinRsvdTolerableRate OBJECT-TYPE
30      SYNTAX      Unsigned32
31      UNITS       "b/s"
32      MAX-ACCESS  read-only
33      STATUS      current
34      DESCRIPTION
35          "Minimum Tolerable Traffic Rate = R (bits/sec) with
36              time base T(sec) means the following. Let S denote
37              additional demand accumulated at the MAC SAP of the
38              transmitter during an arbitrary time interval of the
39              length T. Then the amount of data forwarded at the
40              receiver to CS (in bits) during this interval should
41              be not less than min {S, R * T}."
42      REFERENCE
43          "Subclause 11.13.9 in IEEE Std 802.16-2004"
44      ::= { wmanIf2CmnCpsServiceFlowEntry 22 }
45
46
47
48
49  wmanIf2CmnCpsReqTxPolicy OBJECT-TYPE
50      SYNTAX      BITS {noBroadcastBwReq(0),
51                          reserved1(1),
52                          noPiggybackReq(2),
53                          noFragmentData(3),
54                          noPHS(4),
55                          noSduPacking(5),
56                          noCrc(6),
57                          reserved2(7)}
58      MAX-ACCESS  read-only
59      STATUS      current
60      DESCRIPTION
61          "The value of this parameter provides the capability to
62              specify certain attributes for the associated service
63
64
65

```

```

1         flow. An attribute is enabled by setting the
2         corresponding bit position to 1."
3
4     REFERENCE
5         "Subclause 11.13.12 in IEEE Std 802.16-2004"
6     ::= { wmanIf2CmnCpsServiceFlowEntry 23 }
7
8
9     wmanIf2CmnSfCsSpecification OBJECT-TYPE
10         SYNTAX      WmanIf2CsSpecification
11         MAX-ACCESS   read-only
12         STATUS       current
13         DESCRIPTION
14             "This parameter specifies the convergence sublayer
15             encapsulation mode."
16         REFERENCE
17             "Subclause 11.13.19.1 in IEEE Std 802.16-2004"
18         ::= { wmanIf2CmnCpsServiceFlowEntry 24 }
19
20
21     wmanIf2CmnCpsTargetSaid OBJECT-TYPE
22         SYNTAX      INTEGER (0 .. 65535)
23         MAX-ACCESS   read-only
24         STATUS       current
25         DESCRIPTION
26             "The target SAID parameter indicates the SAID onto
27             which the service flow being set up shall be mapped."
28         REFERENCE
29             "Subclause 11.13.17 in IEEE Std 802.16-2004"
30         ::= { wmanIf2CmnCpsServiceFlowEntry 25 }
31
32
33
34     --
35     -- wmanIf2CmnBsSsConfigurationTable contains global parameters
36     -- common in BS and SS
37     --
38
39     wmanIf2CmnBsSsConfigurationTable OBJECT-TYPE
40         SYNTAX      SEQUENCE OF WmanIf2CmnBsSsConfigurationEntry
41         MAX-ACCESS   not-accessible
42         STATUS       current
43         DESCRIPTION
44             "This table provides one row for each BS sector that
45             contains the system parameters common in both SS and
46             BS. All SSs shall have the same parameters as the BS
47             to which the SSs are associated."
48         REFERENCE
49             "Subclause 10.1 in IEEE Std 802.16-2004"
50         ::= { wmanIf2CmnCps 2 }
51
52
53
54
55     wmanIf2CmnBsSsConfigurationEntry OBJECT-TYPE
56         SYNTAX      WmanIf2CmnBsSsConfigurationEntry
57         MAX-ACCESS   not-accessible
58         STATUS       current
59         DESCRIPTION
60             "This table is indexed by ifIndex, indicating BS
61             sector."
62         INDEX        { ifIndex }
63         ::= { wmanIf2CmnBsSsConfigurationTable 1 }
64
65

```



```

1
2 WmanIf2CmnBsSsConfigurationEntry ::= SEQUENCE {
3     wmanIf2CmnInvitedRangRetries      INTEGER,
4     wmanIf2CmnDSxReqRetries           Unsigned32,
5     wmanIf2CmnDSxRespRetries          Unsigned32,
6     wmanIf2CmnT7Timeout               INTEGER,
7     wmanIf2CmnT8Timeout               INTEGER,
8     wmanIf2CmnT10Timeout              INTEGER,
9     wmanIf2CmnT22Timeout              INTEGER}
10
11
12
13 wmanIf2CmnInvitedRangRetries OBJECT-TYPE
14     SYNTAX      INTEGER (16..65535)
15     MAX-ACCESS  read-write
16     STATUS      current
17     DESCRIPTION
18         "Number of retries on inviting Ranging Requests."
19     ::= { wmanIf2CmnBsSsConfigurationEntry 1 }
20
21
22
23 wmanIf2CmnDSxReqRetries OBJECT-TYPE
24     SYNTAX      Unsigned32
25     MAX-ACCESS  read-write
26     STATUS      current
27     DESCRIPTION
28         "Number of Timeout Retries on DSA/DSC/DSD Requests."
29     DEFVAL      { 3 }
30     ::= { wmanIf2CmnBsSsConfigurationEntry 2 }
31
32
33
34 wmanIf2CmnDSxRespRetries OBJECT-TYPE
35     SYNTAX      Unsigned32
36     MAX-ACCESS  read-write
37     STATUS      current
38     DESCRIPTION
39         "Number of Timeout Retries on DSA/DSC/DSD Responses."
40     DEFVAL      { 3 }
41     ::= { wmanIf2CmnBsSsConfigurationEntry 3 }
42
43
44
45 wmanIf2CmnT7Timeout OBJECT-TYPE
46     SYNTAX      INTEGER (0 .. 1000)
47     UNITS        "milliseconds"
48     MAX-ACCESS  read-write
49     STATUS      current
50     DESCRIPTION
51         "Wait for DSA/DSC/DSD Response Timeout in ms."
52     ::= { wmanIf2CmnBsSsConfigurationEntry 4 }
53
54
55
56 wmanIf2CmnT8Timeout OBJECT-TYPE
57     SYNTAX      INTEGER (0 .. 300)
58     UNITS        "milliseconds"
59     MAX-ACCESS  read-write
60     STATUS      current
61     DESCRIPTION
62         "Wait for DSA/DSC/DSD Acknowledge Timeout in ms."
63     ::= { wmanIf2CmnBsSsConfigurationEntry 5 }
64
65

```

```

1  wmanIf2CmnT10Timeout OBJECT-TYPE
2      SYNTAX      INTEGER (0 .. 3000)
3      UNITS       "milliseconds"
4      MAX-ACCESS  read-write
5      STATUS      current
6      DESCRIPTION
7          "Wait for Transaction End timeout in ms."
8      ::= { wmanIf2CmnBsSsConfigurationEntry 6 }
9
10
11  wmanIf2CmnT22Timeout OBJECT-TYPE
12      SYNTAX      INTEGER (0 .. 500)
13      UNITS       "milliseconds"
14      MAX-ACCESS  read-write
15      STATUS      current
16      DESCRIPTION
17          "Wait for ARQ Reset in ms."
18      ::= { wmanIf2CmnBsSsConfigurationEntry 7 }
19
20
21  -- Common PKM group
22  -- wmanIf2CmnPkmObjects contain the Privacy Sublayer objects that are
23  -- common to both Base Station and Subscriber Station
24  --
25  wmanIf2CmnPkmObjects OBJECT IDENTIFIER ::= { wmanIf2CommonObjects 3 }
26
27
28  --
29  -- Table wmanIf2CmnCryptoSuiteTable
30  --
31  wmanIf2CmnCryptoSuiteTable OBJECT-TYPE
32      SYNTAX      SEQUENCE OF WmanIf2CmnCryptoSuiteEntry
33      MAX-ACCESS  not-accessible
34      STATUS      current
35      DESCRIPTION
36          "This table describes the PKM cryptographic suite
37          capabilites for each SS or BS wireless interface."
38      ::= { wmanIf2CmnPkmObjects 1 }
39
40
41  wmanIf2CmnCryptoSuiteEntry OBJECT-TYPE
42      SYNTAX      WmanIf2CmnCryptoSuiteEntry
43      MAX-ACCESS  not-accessible
44      STATUS      current
45      DESCRIPTION
46          "Each entry contains the cryptographic suite pair that SS
47          or BS supports."
48      INDEX      { ifIndex, wmanIf2CmnCryptoSuiteIndex }
49      ::= { wmanIf2CmnCryptoSuiteTable 1 }
50
51
52  WmanIf2CmnCryptoSuiteEntry ::= SEQUENCE {
53      wmanIf2CmnCryptoSuiteIndex      Integer32,
54      wmanIf2CmnCryptoSuiteDataEncryptAlg  WmanIf2DataEncryptAlgId,
55      wmanIf2CmnCryptoSuiteDataAuthAlg    WmanIf2DataAuthAlgId,
56      wmanIf2CmnCryptoSuiteTekEncryptAlg  WmanIf2TekEncryptAlgId}
57
58
59  wmanIf2CmnCryptoSuiteIndex OBJECT-TYPE
60      SYNTAX      Integer32 (1 .. 1000)
61
62
63
64
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "The index for a cryptographic suite row."
5      ::= { wmanIf2CmnCryptoSuiteEntry 1 }
6
7
8  wmanIf2CmnCryptoSuiteDataEncryptAlg OBJECT-TYPE
9      SYNTAX          WmanIf2DataEncryptAlgId
10     MAX-ACCESS      read-only
11     STATUS          current
12     DESCRIPTION
13         "The value of this object is the data encryption algorithm
14         for this cryptographic suite capability."
15     REFERENCE
16         "Table 375, IEEE Std 802.16-2004"
17     ::= { wmanIf2CmnCryptoSuiteEntry 2 }
18
19
20
21  wmanIf2CmnCryptoSuiteDataAuthentAlg OBJECT-TYPE
22     SYNTAX          WmanIf2DataAuthAlgId
23     MAX-ACCESS      read-only
24     STATUS          current
25     DESCRIPTION
26         "The value of this object is the data authentication
27         algorithm for this cryptographic suite capability."
28     REFERENCE
29         "Table 376, IEEE Std 802.16-2004"
30     ::= { wmanIf2CmnCryptoSuiteEntry 3 }
31
32
33
34
35  wmanIf2CmnCryptoSuiteTekEncryptAlg OBJECT-TYPE
36     SYNTAX          WmanIf2TekEncryptAlgId
37     MAX-ACCESS      read-only
38     STATUS          current
39     DESCRIPTION
40         "The value of this object is the TEK key encryption
41         algorithm for this cryptographic suite capability."
42     REFERENCE
43         "Table 377, IEEE Std 802.16-2004"
44     ::= { wmanIf2CmnCryptoSuiteEntry 4 }
45
46
47
48  --
49  -- Conformance Information
50  --
51  wmanIf2MibConformance OBJECT IDENTIFIER ::= {wmanIf2Mib 2}
52  wmanIf2MibGroups       OBJECT IDENTIFIER ::= {wmanIf2MibConformance 1}
53  wmanIf2MibCompliances  OBJECT IDENTIFIER ::= {wmanIf2MibConformance 2}
54
55
56  -- compliance statements
57  wmanIf2MibCompliance MODULE-COMPLIANCE
58      STATUS          current
59      DESCRIPTION
60          "The compliance statement for devices that implement
61          Wireless MAN interfaces as defined in IEEE Std 802.16-2004."
62
63
64  MODULE -- wmanIf2Mib
65

```

```

1
2 MANDATORY-GROUPS -- unconditionally mandatory groups
3 { wmanIf2MibCommonGroup }
4
5
6 GROUP wmanIf2MibQoSGroup -- unconditionally mandatory group
7 DESCRIPTION
8 "This group is mandatory for Base Station and subscriber
9 station."
10
11
12 GROUP wmanIf2MibBsGroup -- conditionally mandatory group
13 DESCRIPTION
14 "This group is mandatory for Base Station."
15
16
17 GROUP wmanIf2MibBsAasGroup -- optional group
18 DESCRIPTION
19 "This group is mandatory for Base Station."
20
21
22 GROUP wmanIf2MibSsGroup -- conditionally mandatory group
23 DESCRIPTION
24 "This group is mandatory for Subscriber Station."
25
26
27 GROUP wmanIf2MibBsOfdmGroup -- conditionally mandatory group
28 DESCRIPTION
29 "This group is mandatory for Base Station
30 implementaing the OFDM PHY."
31
32
33 GROUP wmanIf2MibSsOfdmGroup -- conditionally mandatory group
34 DESCRIPTION
35 "This group is mandatory for Subscriber Station
36 implementing the OFDM PHY."
37
38
39 GROUP wmanIf2MibBsOfdmaGroup -- conditionally mandatory group
40 DESCRIPTION
41 "This group is mandatory for Base Station
42 implementaing the OFDMA PHY."
43
44
45 GROUP wmanIf2MibSsOfdmaGroup -- conditionally mandatory group
46 DESCRIPTION
47 "This group is mandatory for Subscriber Station
48 implementing the OFDMA PHY."
49
50
51 GROUP wmanIf2MibBsNotificationGroup -- unconditionally
52 -- mandatory groups
53 DESCRIPTION
54 "This group is mandatory for Base Station."
55
56
57 GROUP wmanIf2MibSsNotificationGroup -- optional group
58 DESCRIPTION
59 "This group is optional for Subscriber Station."
60
61
62 GROUP wmanIf2MibCmnPhsGroup -- optional group
63 DESCRIPTION
64 "This group is optional for Base Station and
65 Subscriber Station."

```

```

1
2      GROUP wmanIf2MibBsPhsGroup -- optional group
3      DESCRIPTION
4          "This group is optional for Base Station."
5      ::= { wmanIf2MibCompliances 1 }
6
7
8      wmanIf2MibCommonGroup      OBJECT-GROUP
9      OBJECTS {-- Classification
10          wmanIf2CmnClassifierRulePriority,
11          wmanIf2CmnClassifierRuleIpTosLow,
12          wmanIf2CmnClassifierRuleIpTosHigh,
13          wmanIf2CmnClassifierRuleIpTosMask,
14          wmanIf2CmnClassifierRuleIpProtocol,
15          wmanIf2CmnClassifierRuleIpSourceAddr,
16          wmanIf2CmnClassifierRuleIpSourceMask,
17          wmanIf2CmnClassifierRuleIpDestAddr,
18          wmanIf2CmnClassifierRuleIpDestMask,
19          wmanIf2CmnClassifierRuleSourcePortStart,
20          wmanIf2CmnClassifierRuleSourcePortEnd,
21          wmanIf2CmnClassifierRuleDestPortStart,
22          wmanIf2CmnClassifierRuleDestPortEnd,
23          wmanIf2CmnClassifierRuleDestMacAddr,
24          wmanIf2CmnClassifierRuleDestMacMask,
25          wmanIf2CmnClassifierRuleSourceMacAddr,
26          wmanIf2CmnClassifierRuleSourceMacMask,
27          wmanIf2CmnClassifierRuleEnetProtocolType,
28          wmanIf2CmnClassifierRuleEnetProtocol,
29          wmanIf2CmnClassifierRuleUserPriLow,
30          wmanIf2CmnClassifierRuleUserPriHigh,
31          wmanIf2CmnClassifierRuleVlanId,
32          wmanIf2CmnClassifierRuleState,
33          wmanIf2CmnClassifierRulePkts,
34          wmanIf2CmnClassifierRuleIpv6FlowLabel,
35          wmanIf2CmnClassifierRuleBitMap,
36
37          -- Configuration parameters
38          wmanIf2CmnCpsTargetSaid,
39          wmanIf2CmnInvitedRangRetries,
40          wmanIf2CmnDSxReqRetries,
41          wmanIf2CmnDSxRespRetries,
42          wmanIf2CmnT7Timeout,
43          wmanIf2CmnT8Timeout,
44          wmanIf2CmnT10Timeout,
45          wmanIf2CmnT22Timeout,
46          wmanIf2CmnCryptoSuiteDataEncryptAlg,
47          wmanIf2CmnCryptoSuiteDataAuthentAlg,
48          wmanIf2CmnCryptoSuiteTekEncryptAlg}
49
50      STATUS      current
51      DESCRIPTION
52          "This group contains objects for both BS and SS,
53          and are independent of PHY."
54      ::= { wmanIf2MibGroups 1 }
55
56
57      wmanIf2MibQoSGroup      OBJECT-GROUP
58
59
60
61
62
63
64
65

```

```

1      OBJECTS {wmanIf2CmnCpsSfId,
2                wmanIf2CmnCpsSfCid,
3                wmanIf2CmnCpsSfDirection,
4                wmanIf2CmnCpsSfState,
5                wmanIf2CmnCpsTrafficPriority,
6                wmanIf2CmnCpsMaxSustainedRate,
7                wmanIf2CmnCpsMaxTrafficBurst,
8                wmanIf2CmnCpsMinReservedRate,
9                wmanIf2CmnCpsToleratedJitter,
10               wmanIf2CmnCpsMaxLatency,
11               wmanIf2CmnCpsFixedVsVariableSduInd,
12               wmanIf2CmnCpsSduSize,
13               wmanIf2CmnCpsSfSchedulingType,
14               wmanIf2CmnCpsArqEnable,
15               wmanIf2CmnCpsArqWindowSize,
16               wmanIf2CmnCpsArqBlockLifetime,
17               wmanIf2CmnCpsArqSyncLossTimeout,
18               wmanIf2CmnCpsArqDeliverInOrder,
19               wmanIf2CmnCpsArqRxPurgeTimeout,
20               wmanIf2CmnCpsArqBlockSize,
21               wmanIf2CmnCpsMinRsvdTolerableRate,
22               wmanIf2CmnCpsReqTxPolicy,
23               wmanIf2CmnSfCsSpecification}
24
25      STATUS      current
26
27      DESCRIPTION
28
29      "This group contains QoS objects for both BS and SS."
30
31      ::= { wmanIf2MibGroups 2 }
32
33
34
35      wmanIf2MibBsGroup      OBJECT-GROUP
36      OBJECTS {-- Service classes
37                wmanIf2BsSfDirection,
38                wmanIf2BsServiceClassIndex,
39                wmanIf2BsSfState,
40                wmanIf2BsSfProvisionedTime,
41                wmanIf2BsProvisionedSfRowStatus,
42                wmanIf2BsSsProvisionedForSfRowStatus,
43                wmanIf2BsSfCsSpecification,
44                wmanIf2BsQosServiceClassName,
45                wmanIf2BsQoSSTrafficPriority,
46                wmanIf2BsQoSMaxSustainedRate,
47                wmanIf2BsQoSMaxTrafficBurst,
48                wmanIf2BsQoSMinReservedRate,
49                wmanIf2BsQOSToleratedJitter,
50                wmanIf2BsQoSMaxLatency,
51                wmanIf2BsQoSFixedVsVariableSduInd,
52                wmanIf2BsQOSSduSize,
53                wmanIf2BsQoSScSchedulingType,
54                wmanIf2BsQoSScArqEnable,
55                wmanIf2BsQoSScArqWindowSize,
56                wmanIf2BsQoSScArqBlockLifetime,
57                wmanIf2BsQoSScArqSyncLossTimeout,
58                wmanIf2BsQoSScArqDeliverInOrder,
59                wmanIf2BsQoSScArqRxPurgeTimeout,
60                wmanIf2BsQoSScArqBlockSize,
61
62
63
64
65

```

```

1      wmanIf2BsQoSSCMinRsvdTolerableRate,
2      wmanIf2BsQoSReqTxPolicy,
3      wmanIf2BsQoSServiceClassRowStatus,
4
5      -- Classification
6      wmanIf2BsClassifierRulePriority,
7      wmanIf2BsClassifierRuleIpTosLow,
8      wmanIf2BsClassifierRuleIpTosHigh,
9      wmanIf2BsClassifierRuleIpTosMask,
10     wmanIf2BsClassifierRuleIpProtocol,
11     wmanIf2BsClassifierRuleIpSourceAddr,
12     wmanIf2BsClassifierRuleIpSourceMask,
13     wmanIf2BsClassifierRuleIpDestAddr,
14     wmanIf2BsClassifierRuleIpDestMask,
15     wmanIf2BsClassifierRuleSourcePortStart,
16     wmanIf2BsClassifierRuleSourcePortEnd,
17     wmanIf2BsClassifierRuleDestPortStart,
18     wmanIf2BsClassifierRuleDestPortEnd,
19     wmanIf2BsClassifierRuleDestMacAddr,
20     wmanIf2BsClassifierRuleDestMacMask,
21     wmanIf2BsClassifierRuleSourceMacAddr,
22     wmanIf2BsClassifierRuleSourceMacMask,
23     wmanIf2BsClassifierRuleEnetProtocolType,
24     wmanIf2BsClassifierRuleEnetProtocol,
25     wmanIf2BsClassifierRuleUserPriLow,
26     wmanIf2BsClassifierRuleUserPriHigh,
27     wmanIf2BsClassifierRuleVlanId,
28     wmanIf2BsClassifierRuleState,
29     wmanIf2BsClassifierRulePhsSize,
30     wmanIf2BsClassifierRulePhsMask,
31     wmanIf2BsClassifierRulePhsVerify,
32     wmanIf2BsClassifierRuleIpv6FlowLabel,
33     wmanIf2BsClassifierRuleBitMap,
34     wmanIf2BsClassifierRuleRowStatus,
35
36     -- Packet counters
37     wmanIf2BsSsMacSduCount,
38     wmanIf2BsSsOctetCount,
39     wmanIf2BsSsResetCounter,
40     wmanIf2BsSsResetCounterTime,
41
42     -- Capability negotiation
43     wmanIf2BsSsBasicCid,
44     wmanIf2BsSsPrimaryCid,
45     wmanIf2BsSsSecondaryCid,
46     wmanIf2BsSsManagementSupport,
47     wmanIf2BsSsIpManagementMode,
48     wmanIf2Bs2ndMgmtDlQoSProfileIndex,
49     wmanIf2Bs2ndMgmtUlQoSProfileIndex,
50     wmanIf2BsAutoSfidEnabled,
51     wmanIf2BsAutoSfidRangeMin,
52     wmanIf2BsAutoSfidRangeMax,
53     wmanIf2BsResetSector,
54     wmanIf2BsSs2ndMgmtArqEnable,

```

```

1      wmanIf2BsSs2ndMgmtArqWindowSize,
2      wmanIf2BsSs2ndMgmtArqDnLinkTxDelay,
3      wmanIf2BsSs2ndMgmtArqUpLinkTxDelay,
4      wmanIf2BsSs2ndMgmtArqDnLinkRxDelay,
5      wmanIf2BsSs2ndMgmtArqUpLinkRxDelay,
6      wmanIf2BsSs2ndMgmtArqBlockLifetime,
7      wmanIf2BsSs2ndMgmtArqSyncLossTimeout,
8      wmanIf2BsSs2ndMgmtArqDeliverInOrder,
9      wmanIf2BsSs2ndMgmtArqRxPurgeTimeout,
10     wmanIf2BsSs2ndMgmtArqBlockSize,
11     wmanIf2BsSsVendorIdEncoding,
12     wmanIf2BsSsAasBroadcastPermission,
13     wmanIf2BsSsMaxTxPowerBpsk,
14     wmanIf2BsSsMaxTxPowerQpsk,
15     wmanIf2BsSsMaxTxPower16Qam,
16     wmanIf2BsSsMaxTxPower64Qam,
17
18     -- Configuration parameters
19     wmanIf2BsSsMacVersion,
20     wmanIf2BsDcdInterval,
21     wmanIf2BsUcdInterval,
22     wmanIf2BsUcdTransition,
23     wmanIf2BsDcdTransition,
24     wmanIf2BsInitialRangingInterval,
25     wmanIf2BsSsULMapProcTime,
26     wmanIf2BsSsRangRespProcTime,
27     wmanIf2BsT5Timeout,
28     wmanIf2BsT9Timeout,
29     wmanIf2BsT13Timeout,
30     wmanIf2BsT15Timeout,
31     wmanIf2BsT17Timeout,
32     wmanIf2BsT27IdleTimer,
33     wmanIf2BsT27ActiveTimer,
34
35     -- Performance monitoring
36     wmanIf2BsHistogramIndex,
37     wmanIf2BsChannelNumber,
38     wmanIf2BsStartFrame,
39     wmanIf2BsDuration,
40     wmanIf2BsBasicReport,
41     wmanIf2BsMeanCinrReport,
42     wmanIf2BsMeanRssiReport,
43     wmanIf2BsStdDeviationCinrReport,
44     wmanIf2BsStdDeviationRssiReport,
45
46     -- Capability negotiation
47     wmanIf2BsSsReqCapUplinkCidSupport,
48     wmanIf2BsSsReqCapArqSupport,
49     wmanIf2BsSsReqCapDsxFowControl,
50     wmanIf2BsSsReqCapMacCrcSupport,
51     wmanIf2BsSsReqCapMcaFlowControl,
52     wmanIf2BsSsReqCapMcpGroupCidSupport,
53     wmanIf2BsSsReqCapPkmFlowControl,
54     wmanIf2BsSsReqCapAuthPolicyControl,
55
56
57
58
59
60
61
62
63
64
65

```



```

1      wmanIf2BsSsReqCapMaxNumOfSupportedSA,
2      wmanIf2BsSsReqCapIpVersion,
3      wmanIf2BsSsReqCapMacCsSupportBitMap,
4      wmanIf2BsSsReqCapMaxNumOfClassifier,
5      wmanIf2BsSsReqCapPhsSupport,
6      wmanIf2BsSsReqCapBandwidthAllocSupport,
7      wmanIf2BsSsReqCapPduConstruction,
8      wmanIf2BsSsReqCapTtgTransitionGap,
9      wmanIf2BsSsReqCapRtgTransitionGap,
10     wmanIf2BsSsRspCapUplinkCidSupport,
11     wmanIf2BsSsRspCapArqSupport,
12     wmanIf2BsSsRspCapDsxFowControl,
13     wmanIf2BsSsRspCapMacCrcSupport,
14     wmanIf2BsSsRspCapMcaFlowControl,
15     wmanIf2BsSsRspCapMcpGroupCidSupport,
16     wmanIf2BsSsRspCapPkmFlowControl,
17     wmanIf2BsSsRspCapAuthPolicyControl,
18     wmanIf2BsSsRspCapMaxNumOfSupportedSA,
19     wmanIf2BsSsRspCapIpVersion,
20     wmanIf2BsSsRspCapMacCsSupportBitMap,
21     wmanIf2BsSsRspCapMaxNumOfClassifier,
22     wmanIf2BsSsRspCapPhsSupport,
23     wmanIf2BsSsRspCapBandwidthAllocSupport,
24     wmanIf2BsSsRspCapPduConstruction,
25     wmanIf2BsSsRspCapTtgTransitionGap,
26     wmanIf2BsSsRspCapRtgTransitionGap,
27     wmanIf2BsCapUplinkCidSupport,
28     wmanIf2BsCapArqSupport,
29     wmanIf2BsCapDsxFowControl,
30     wmanIf2BsCapMacCrcSupport,
31     wmanIf2BsCapMcaFlowControl,
32     wmanIf2BsCapMcpGroupCidSupport,
33     wmanIf2BsCapPkmFlowControl,
34     wmanIf2BsCapAuthPolicyControl,
35     wmanIf2BsCapMaxNumOfSupportedSA,
36     wmanIf2BsCapIpVersion,
37     wmanIf2BsCapMacCsSupportBitMap,
38     wmanIf2BsCapMaxNumOfClassifier,
39     wmanIf2BsCapPhsSupport,
40     wmanIf2BsCapBandwidthAllocSupport,
41     wmanIf2BsCapPduConstruction,
42     wmanIf2BsCapTtgTransitionGap,
43     wmanIf2BsCapRtgTransitionGap,
44     wmanIf2BsCapCfgUplinkCidSupport,
45     wmanIf2BsCapCfgArqSupport,
46     wmanIf2BsCapCfgDsxFowControl,
47     wmanIf2BsCapCfgMacCrcSupport,
48     wmanIf2BsCapCfgMcaFlowControl,
49     wmanIf2BsCapCfgMcpGroupCidSupport,
50     wmanIf2BsCapCfgPkmFlowControl,
51     wmanIf2BsCapCfgAuthPolicyControl,
52     wmanIf2BsCapCfgMaxNumOfSupportedSA,
53     wmanIf2BsCapCfgIpVersion,
54     wmanIf2BsCapCfgMacCsSupportBitMap,
55

```

```

1      wmanIf2BsCapCfgMaxNumOfClassifier,
2      wmanIf2BsCapCfgPhsSupport,
3      wmanIf2BsCapCfgBandwidthAllocSupport,
4      wmanIf2BsCapCfgPduConstruction,
5      wmanIf2BsCapCfgTtgTransitionGap,
6      wmanIf2BsCapCfgRtgTransitionGap,
7      wmanIf2BsSsActionsResetSs,
8      wmanIf2BsSsActionsAbortSs,
9      wmanIf2BsSsActionsOverrideDnFreq,
10     wmanIf2BsSsActionsOverrideChannelId,
11     wmanIf2BsSsActionsDeReRegSs,
12     wmanIf2BsSsActionsDeReRegSsCode,
13     wmanIf2BsSsActionsRowStatus,
14
15     -- Privacy sublayer
16     wmanIf2BsPkmDefaultAuthLifetime,
17     wmanIf2BsPkmDefaultTekLifetime,
18     wmanIf2BsPkmDefaultSelfSigManufCertTrust,
19     wmanIf2BsPkmCheckCertValidityPeriods,
20     wmanIf2BsPkmAuthentInfos,
21     wmanIf2BsPkmAuthRequests,
22     wmanIf2BsPkmAuthReplies,
23     wmanIf2BsPkmAuthRejects,
24     wmanIf2BsPkmAuthInvalids,
25     wmanIf2BsSsPkmAuthKeySequenceNumber,
26     wmanIf2BsSsPkmAuthExpiresOld,
27     wmanIf2BsSsPkmAuthExpiresNew,
28     wmanIf2BsSsPkmAuthLifetime,
29     wmanIf2BsSsPkmAuthReset,
30     wmanIf2BsSsPkmAuthInfos,
31     wmanIf2BsSsPkmAuthRequests,
32     wmanIf2BsSsPkmAuthReplies,
33     wmanIf2BsSsPkmAuthRejects,
34     wmanIf2BsSsPkmAuthInvalids,
35     wmanIf2BsSsPkmAuthRejectErrorCode,
36     wmanIf2BsSsPkmAuthRejectErrorString,
37     wmanIf2BsSsPkmAuthInvalidErrorCode,
38     wmanIf2BsSsPkmAuthInvalidErrorString,
39     wmanIf2BsSsPkmAuthPrimarySAId,
40     wmanIf2BsSsPkmAuthValidStatus,
41     wmanIf2BsPkmTekSAType,
42     wmanIf2BsPkmTekDataEncryptAlg,
43     wmanIf2BsPkmTekDataAuthentAlg,
44     wmanIf2BsPkmTekEncryptAlg,
45     wmanIf2BsPkmTekLifetime,
46     wmanIf2BsPkmTekKeySequenceNumber,
47     wmanIf2BsPkmTekExpiresOld,
48     wmanIf2BsPkmTekExpiresNew,
49     wmanIf2BsPkmTekReset,
50     wmanIf2BsPkmKeyRequests,
51     wmanIf2BsPkmKeyReplies,
52     wmanIf2BsPkmKeyRejects,
53     wmanIf2BsPkmTekInvalids,
54     wmanIf2BsPkmKeyRejectErrorCode,
55
56
57
58
59
60
61
62
63
64
65

```

```

1      wmanIf2BsPkmKeyRejectErrorString,
2      wmanIf2BsPkmTekInvalidErrorCode,
3      wmanIf2BsPkmTekInvalidErrorString,
4
5      -- Notification
6      wmanIf2BsTrapControlRegister,
7      wmanIf2BsStatusTrapControlRegister,
8      wmanIf2BsRssiLowThreshold,
9      wmanIf2BsRssiHighThreshold,
10     wmanIf2BsSsNotificationMacAddr,
11     wmanIf2BsSsStatusValue,
12     wmanIf2BsSsStatusInfo,
13     wmanIf2BsDynamicServiceType,
14     wmanIf2BsDynamicServiceFailReason,
15     wmanIf2BsSsRssiStatus,
16     wmanIf2BsSsRssiStatusInfo,
17     wmanIf2BsSsRegisterStatus}
18
19 STATUS      current
20
21 DESCRIPTION
22     "This group contains objects for BS, and are
23     independent of PHY."
24 ::= { wmanIf2MibGroups 3 }
25
26
27
28
29 wmanIf2MibBsAasGroup      OBJECT-GROUP
30     OBJECTS {-- AAS Configuration parameters
31         wmanIf2BsAasChanFbckReqFreq,
32         wmanIf2BsAasBeamSelectFreq,
33         wmanIf2BsAasChanFbckReqResolution,
34         wmanIf2BsAasBeamReqResolution,
35         wmanIf2BsAasNumOptDiversityZones}
36
37 STATUS      current
38
39 DESCRIPTION
40     "This group contains objects for AAS in BS."
41 ::= { wmanIf2MibGroups 4 }
42
43
44 wmanIf2MibSsGroup          OBJECT-GROUP
45     OBJECTS {-- Configuration parameters
46         wmanIf2SsLostDLMapInterval,
47         wmanIf2SsLostULMapInterval,
48         wmanIf2SsContentionRangRetries,
49         wmanIf2SsRequestRetries,
50         wmanIf2SsRegRequestRetries,
51         wmanIf2SsTftpBackoffStart,
52         wmanIf2SsTftpBackoffEnd,
53         wmanIf2SsTftpRequestRetries,
54         wmanIf2SsTftpDownloadRetries,
55         wmanIf2SsTftpWait,
56         wmanIf2SsToDRetries,
57         wmanIf2SsToDRetryPeriod,
58         wmanIf2SsT1Timeout,
59         wmanIf2SsT2Timeout,
60         wmanIf2SsT3Timeout,
61         wmanIf2SsT4Timeout,
62         wmanIf2SsT6Timeout,
63
64
65

```

```

1      wmanIf2SsT12Timeout,
2      wmanIf2SsT14Timeout,
3      wmanIf2SsT16Timeout,
4      wmanIf2SsT18Timeout,
5      wmanIf2SsT19Timeout,
6      wmanIf2SsT20Timeout,
7      wmanIf2SsT21Timeout,
8      wmanIf2SsSBCRequestRetries,
9      wmanIf2SsTftpCpltRetries,
10     wmanIf2SsT26Timeout,
11     wmanIf2SsDLManagProcTime,
12
13
14
15     -- Performance monitoring
16     wmanIf2SsChannelNumber,
17     wmanIf2SsStartFrame ,
18     wmanIf2SsDuration,
19     wmanIf2SsBasicReport,
20     wmanIf2SsMeanCinrReport,
21     wmanIf2SsStdDeviationCinrReport,
22     wmanIf2SsMeanRssiReport,
23     wmanIf2SsStdDeviationRssiReport,
24
25
26
27     -- Privacy sublayer
28     wmanIf2SsPkmAuthState,
29     wmanIf2SsPkmAuthKeySequenceNumber,
30     wmanIf2SsPkmAuthExpiresOld,
31     wmanIf2SsPkmAuthExpiresNew ,
32     wmanIf2SsPkmAuthReset,
33     wmanIf2SsPkmAuthentInfos,
34     wmanIf2SsPkmAuthRequests,
35     wmanIf2SsPkmAuthReplies,
36     wmanIf2SsPkmAuthRejects,
37     wmanIf2SsPkmAuthInvalids,
38     wmanIf2SsPkmAuthRejectErrorCode,
39     wmanIf2SsPkmAuthRejectErrorString,
40     wmanIf2SsPkmAuthInvalidErrorCode,
41     wmanIf2SsPkmAuthInvalidErrorString ,
42     wmanIf2SsPkmAuthGraceTime,
43     wmanIf2SsPkmTekGraceTime,
44     wmanIf2SsPkmAuthWaitTimeout,
45     wmanIf2SsPkmReauthWaitTimeout,
46     wmanIf2SsPkmOpWaitTimeout,
47     wmanIf2SsPkmRekeyWaitTimeout,
48     wmanIf2SsPkmAuthRejectWaitTimeout,
49     wmanIf2SsPkmTekSAType,
50     wmanIf2SsPkmTekDataEncryptAlg,
51     wmanIf2SsPkmTekDataAuthentAlg,
52     wmanIf2SsPkmTekEncryptAlg,
53     wmanIf2SsPkmTekState,
54     wmanIf2SsPkmTekKeySequenceNumber,
55     wmanIf2SsPkmTekExpiresOld,
56     wmanIf2SsPkmTekExpiresNew,
57     wmanIf2SsPkmTekKeyRequests,
58     wmanIf2SsPkmTekKeyReplies,
59
60
61
62
63
64
65

```

```

1      wmanIf2SsPkmTekKeyRejects,
2      wmanIf2SsPkmTekInvalids,
3      wmanIf2SsPkmTekAuthPends,
4      wmanIf2SsPkmTekKeyRejectErrorCode,
5      wmanIf2SsPkmTekKeyRejectErrorString,
6      wmanIf2SsPkmTekInvalidErrorCode,
7      wmanIf2SsPkmTekInvalidErrorString,
8      wmanIf2SsDeviceCert,
9      wmanIf2SsDeviceManufCert,
10
11
12
13      -- Notofoction
14      wmanIf2SsTrapControlRegister,
15      wmanIf2SsRssiLowThreshold,
16      wmanIf2SsRssiHighThreshold,
17      wmanIf2SsMacAddress,
18      wmanIf2SsUnknownTlv,
19      wmanIf2SsDynamicServiceType,
20      wmanIf2SsDynamicServiceFailReason,
21      wmanIf2SsRssiStatus,
22      wmanIf2SsRssiStatusInfo}
23
24  STATUS          current
25
26  DESCRIPTION
27      "This group contains objects for SS, and are
28      independent of PHY."
29
30  ::= { wmanIf2MibGroups 5 }
31
32  wmanIf2MibBsOfdmGroup      OBJECT-GROUP
33      OBJECTS {wmanIf2BsOfdmCtBasedResvTimeout,
34      wmanIf2BsOfdmBwReqOppSize,
35      wmanIf2BsOfdmRangReqOppSize,
36      wmanIf2BsOfdmUplinkCenterFreq,
37      wmanIf2BsOfdmNumSubChReqRegionFull,
38      wmanIf2BsOfdmNumSymbolsReqRegionFull,
39      wmanIf2BsOfdmSubChFocusCtCode,
40      wmanIf2BsOfdmUpLinkChannelId,
41      wmanIf2BsOfdmBsEIRP,
42      wmanIf2BsOfdmChannelNumber,
43      wmanIf2BsOfdmTTG,
44      wmanIf2BsOfdmRTG,
45      wmanIf2BsOfdmInitRngMaxRSS,
46      wmanIf2BsOfdmDownlinkCenterFreq,
47      wmanIf2BsOfdmBsId,
48      wmanIf2BsOfdmMacVersion,
49      wmanIf2BsOfdmFrameDurationCode,
50      wmanIf2BsOfdmDownLinkChannelId,
51      wmanIf2BsOfdmUcdFecCodeType,
52      wmanIf2BsOfdmFocusCtPowerBoost,
53      wmanIf2BsOfdmUcdTcsEnable,
54      wmanIf2BsOfdmUcdBurstProfileRowStatus,
55      wmanIf2BsOfdmDownlinkFrequency,
56      wmanIf2BsOfdmDcdFecCodeType,
57      wmanIf2BsOfdmDiucMandatoryExitThresh,
58      wmanIf2BsOfdmDiucMinEntryThresh,
59      wmanIf2BsOfdmTcsEnable,
60
61
62
63
64
65

```

```

1      wmanIf2BsOfdmDcdBurstProfileRowStatus,
2      wmanIf2BsOfdmMinReqRegionFullTxOpp,
3      wmanIf2BsOfdmMinFocusedCtTxOpp,
4      wmanIf2BsOfdmMaxRoundTripDelay,
5      wmanIf2BsOfdmRangeAbortTimingThold,
6      wmanIf2BsOfdmRangeAbortPowerThold ,
7      wmanIf2BsOfdmRangeAbortFreqThold,
8      wmanIf2BsOfdmDnlkRateId,
9      wmanIf2BsOfdmRatioG,
10     wmanIf2BsSsOfdmReqCapFftSizes,
11     wmanIf2BsSsOfdmReqCapSsDemodulator,
12     wmanIf2BsSsOfdmReqCapSsModulator,
13     wmanIf2BsSsOfdmReqCapFocusedCtSupport,
14     wmanIf2BsSsOfdmReqCapTcSublayerSupport,
15     wmanIf2BsSsOfdmRspCapFftSizes,
16     wmanIf2BsSsOfdmRspCapSsDemodulator,
17     wmanIf2BsSsOfdmRspCapSsModulator,
18     wmanIf2BsSsOfdmRspCapFocusedCtSupport,
19     wmanIf2BsSsOfdmRspCapTcSublayerSupport,
20     wmanIf2BsOfdmCapFftSizes,
21     wmanIf2BsOfdmCapSsDemodulator,
22     wmanIf2BsOfdmCapSsModulator,
23     wmanIf2BsOfdmCapFocusedCtSupport,
24     wmanIf2BsOfdmCapTcSublayerSupport,
25     wmanIf2BsOfdmCapCfgFftSizes,
26     wmanIf2BsOfdmCapCfgSsDemodulator,
27     wmanIf2BsOfdmCapCfgSsModulator,
28     wmanIf2BsOfdmCapCfgFocusedCtSupport,
29     wmanIf2BsOfdmCapCfgTcSublayerSupport}
30
31 STATUS          current
32
33 DESCRIPTION
34     "This group contains objects for BS and OFDM PHY."
35 ::= { wmanIf2MibGroups 6 }
36
37
38
39
40
41
42 wmanIf2MibSsOfdmGroup      OBJECT-GROUP
43     OBJECTS {wmanIf2SsOfdmCtBasedResvTimeout,
44             wmanIf2SsOfdmBwReqOppSize,
45             wmanIf2SsOfdmRangReqOppSize,
46             wmanIf2SsOfdmUplinkCenterFreq,
47             wmanIf2SsOfdmNumSubChReqRegionFull,
48             wmanIf2SsOfdmNumSymbolsReqRegionFull,
49             wmanIf2SsOfdmSubChFocusCtCode,
50             wmanIf2SsOfdmUpLinkChannelId,
51             wmanIf2SsOfdmBsEIRP,
52             wmanIf2SsOfdmChannelNumber,
53             wmanIf2SsOfdmTTG,
54             wmanIf2SsOfdmRTG,
55             wmanIf2SsOfdmInitRngMaxRSS,
56             wmanIf2SsOfdmDownlinkCenterFreq,
57             wmanIf2SsOfdmBsId,
58             wmanIf2SsOfdmMacVersion,
59             wmanIf2SsOfdmFrameDurationCode,
60             wmanIf2SsOfdmDownLinkChannelId,
61             wmanIf2SsOfdmUcdFecCodeType,
62

```

```

1      wmanIf2SsOfdmFocusCtPowerBoost,
2      wmanIf2SsOfdmUcdTcsEnable,
3      wmanIf2SsOfdmDownlinkFrequency,
4      wmanIf2SsOfdmDcdFecCodeType,
5      wmanIf2SsOfdmDiucMandatoryExitThresh,
6      wmanIf2SsOfdmDiucMinEntryThresh,
7      wmanIf2SsOfdmTcsEnable}
8
9  STATUS      current
10
11  DESCRIPTION
12      "This group contains objects for SS and OFDM PHY."
13      ::= { wmanIf2MibGroups 7 }
14
15  wmanIf2MibBsOfdmaGroup      OBJECT-GROUP
16      OBJECTS {wmanIf2BsOfdmaCtBasedResvTimeout,
17      wmanIf2BsOfdmaBwReqOppSize,
18      wmanIf2BsOfdmaRangReqOppSize,
19      wmanIf2BsOfdmaUplinkCenterFreq,
20      wmanIf2BsOfdmaInitRngCodes,
21      wmanIf2BsOfdmaPeriodicRngCodes,
22      wmanIf2BsOfdmaBWReqCodes,
23      wmanIf2BsOfdmaPerRngBackoffStart,
24      wmanIf2BsOfdmaPerRngBackoffEnd,
25      wmanIf2BsOfdmaStartOfRngCodes,
26      wmanIf2BsOfdmaPermutationBase,
27      wmanIf2BsOfdmaULAllocSubchBitmap,
28      wmanIf2BsOfdmaOptPermULAllocSubchBitmap,
29      wmanIf2BsOfdmaBandAMCAllocThreshold,
30      wmanIf2BsOfdmaBandAMCReleaseThreshold,
31      wmanIf2BsOfdmaBandAMCAllocTimer,
32      wmanIf2BsOfdmaBandAMCReleaseTimer,
33      wmanIf2BsOfdmaBandStatRepMAXPeriod,
34      wmanIf2BsOfdmaBandAMCRetryTimer,
35      wmanIf2BsOfdmaSafetyChAllocThreshold,
36      wmanIf2BsOfdmaSafetyChReleaseThreshold,
37      wmanIf2BsOfdmaSafetyChAllocTimer,
38      wmanIf2BsOfdmaSafetyChReleaseTimer,
39      wmanIf2BsOfdmaBinStatRepMAXPeriod,
40      wmanIf2BsOfdmaSafetyChARetryTimer,
41      wmanIf2BsOfdmaHARQAackDelayULBurst,
42      wmanIf2BsOfdmaCQICHBandAMCTranaDelay,
43      wmanIf2BsOfdmaBsEIRP,
44      wmanIf2BsOfdmaChannelNumber,
45      wmanIf2BsOfdmaTTG,
46      wmanIf2BsOfdmaRTG,
47      wmanIf2BsOfdmaInitRngMaxRSS,
48      wmanIf2BsOfdmaDownlinkCenterFreq,
49      wmanIf2BsOfdmaBsId,
50      wmanIf2BsOfdmaMacVersion,
51      wmanIf2BsOfdmaFrameDurationCode,
52      wmanIf2BsOfdmaSizeCqichIdField,
53      wmanIf2BsOfdmaHARQAackDelayBurst,
54      wmanIf2BsOfdmaUcdFecCodeType,
55      wmanIf2BsOfdmaRangingDataRatio,
56      wmanIf2BsOfdmaNorCOVerNOOverride,
57
58
59
60
61
62
63
64
65

```

```

1          wmanIf2BsOfdmaUcdBurstProfileRowStatus,
2          wmanIf2BsOfdmaDownlinkFrequency,
3          wmanIf2BsOfdmaDcdFecCodeType,
4          wmanIf2BsOfdmaDiucMandatoryExitThresh,
5          wmanIf2BsOfdmaDiucMinEntryThresh,
6          wmanIf2BsOfdmaDcdBurstProfileRowStatus}
7
8      STATUS          current
9
10     DESCRIPTION
11         "This group contains objects for BS and OFDMA PHY."
12     ::= { wmanIf2MibGroups 8 }
13
14 wmanIf2MibSsOfdmaGroup      OBJECT-GROUP
15     OBJECTS {wmanIf2SsOfdmaCtBasedResvTimeout,
16             wmanIf2SsOfdmaBwReqOppSize,
17             wmanIf2SsOfdmaRangReqOppSize,
18             wmanIf2SsOfdmaUplinkCenterFreq,
19             wmanIf2SsOfdmaInitRngCodes,
20             wmanIf2SsOfdmaPeriodicRngCodes,
21             wmanIf2SsOfdmaBWReqCodes,
22             wmanIf2SsOfdmaPerRngBackoffStart,
23             wmanIf2SsOfdmaPerRngBackoffEnd,
24             wmanIf2SsOfdmaStartOfRngCodes,
25             wmanIf2SsOfdmaPermutationBase,
26             wmanIf2SsOfdmaULAllocSubchBitmap,
27             wmanIf2SsOfdmaOptPermULAllocSubchBitmap,
28             wmanIf2SsOfdmaBandAMCAllocThreshold,
29             wmanIf2SsOfdmaBandAMCReleaseThreshold,
30             wmanIf2SsOfdmaBandAMCAllocTimer,
31             wmanIf2SsOfdmaBandAMCReleaseTimer,
32             wmanIf2SsOfdmaBandStatRepMAXPeriod,
33             wmanIf2SsOfdmaBandAMCRetryTimer,
34             wmanIf2SsOfdmaSafetyChAllocThreshold,
35             wmanIf2SsOfdmaSafetyChReleaseThreshold,
36             wmanIf2SsOfdmaSafetyChAllocTimer,
37             wmanIf2SsOfdmaSafetyChReleaseTimer,
38             wmanIf2SsOfdmaBinStatRepMAXPeriod,
39             wmanIf2SsOfdmaSafetyChARetryTimer,
40             wmanIf2SsOfdmaHARQAackDelayULBurst,
41             wmanIf2SsOfdmaCQICHBandAMCTranaDelay,
42             wmanIf2SsOfdmaBsEIRP,
43             wmanIf2SsOfdmaChannelNumber,
44             wmanIf2SsOfdmaTTG,
45             wmanIf2SsOfdmaRTG,
46             wmanIf2SsOfdmaInitRngMaxRSS,
47             wmanIf2SsOfdmaDownlinkCenterFreq,
48             wmanIf2SsOfdmaBsId,
49             wmanIf2SsOfdmaMacVersion,
50             wmanIf2SsOfdmaFrameDurationCode,
51             wmanIf2SsOfdmaSizeCqichIdField,
52             wmanIf2SsOfdmaHARQAackDelayBurst,
53             wmanIf2SsOfdmaUiucIndex,
54             wmanIf2SsOfdmaUcdFecCodeType,
55             wmanIf2SsOfdmaRangingDataRatio,
56             wmanIf2SsOfdmaNorCOverNOverride,

```



```

1          wmanIf2SsOfdmaDiucIndex,
2          wmanIf2SsOfdmaDownlinkFrequency,
3          wmanIf2SsOfdmaDcdFecCodeType,
4          wmanIf2SsOfdmaDiucMandatoryExitThresh,
5          wmanIf2SsOfdmaDiucMinEntryThresh}
6
7      STATUS          current
8
9      DESCRIPTION
10         "This group contains objects for SS and OFDMA PHY."
11     ::= { wmanIf2MibGroups 9 }
12
13 wmanIf2MibBsNotificationGroup      NOTIFICATION-GROUP
14     NOTIFICATIONS {wmanIf2BsSsStatusNotificationTrap,
15                   wmanIf2BsSsDynamicServiceFailTrap,
16                   wmanIf2BsSsRssiStatusChangeTrap,
17                   wmanIf2BsSsPkmFailTrap,
18                   wmanIf2BsSsRegistrerTrap}
19
20     STATUS          current
21
22     DESCRIPTION
23         "This group contains event notifications for BS."
24     ::= { wmanIf2MibGroups 10 }
25
26 wmanIf2MibSsNotificationGroup      NOTIFICATION-GROUP
27     NOTIFICATIONS {wmanIf2SsTlvUnknownTrap,
28                   wmanIf2SsDynamicServiceFailTrap,
29                   wmanIf2SsDhcpSuccessTrap,
30                   wmanIf2SsRssiStatusChangeTrap}
31
32     STATUS          current
33
34     DESCRIPTION
35         "This group contains event notifications for SS."
36     ::= { wmanIf2MibGroups 11 }
37
38 wmanIf2MibCmnPhsGroup              OBJECT-GROUP
39     OBJECTS {-- Payload header supression
40             wmanIf2CmnPhsRulePhsField,
41             wmanIf2CmnPhsRulePhsMask,
42             wmanIf2CmnPhsRulePhsSize,
43             wmanIf2CmnPhsRulePhsVerify}
44
45     STATUS          current
46
47     DESCRIPTION
48         "This group contains common objects for PHS."
49     ::= { wmanIf2MibGroups 12 }
50
51 wmanIf2MibBsPhsGroup               OBJECT-GROUP
52     OBJECTS {-- Payload header supression
53             wmanIf2BsClassifierRulePhsSize,
54             wmanIf2BsClassifierRulePhsMask,
55             wmanIf2BsClassifierRulePhsVerify,
56             wmanIf2BsClassifierRuleBitMap}
57
58     STATUS          current
59
60     DESCRIPTION
61         "This group contains BS objects for PHS."
62     ::= { wmanIf2MibGroups 13 }
63
64 END
65

```

15.2.3.2 WMAN-IF2M-MIB

WMAN-IF2M-MIB DEFINITIONS ::= BEGIN

IMPORTS

```

    MODULE-IDENTITY,
    OBJECT-TYPE,
    NOTIFICATION-TYPE,
    Unsigned32, Integer32, Counter32,
    Counter64, transmission
        FROM SNMPv2-SMI
    SnmpAdminString
        FROM SNMP-FRAMEWORK-MIB
    TEXTUAL-CONVENTION,
    MacAddress, RowStatus, TruthValue,
    TimeStamp, DateAndTime
        FROM SNMPv2-TC
    InetAddressType, InetAddress
        FROM INET-ADDRESS-MIB
    OBJECT-GROUP,
    MODULE-COMPLIANCE,
    NOTIFICATION-GROUP
        FROM SNMPv2-CONF
    ifIndex
        FROM IF-MIB;
```

wmanIf2mMib MODULE-IDENTITY

```

    LAST-UPDATED      "200610160000Z" -- October 16, 2006
    ORGANIZATION      "IEEE 802.16"
    CONTACT-INFO
        "WG E-mail: stds-802-16@ieee.org
        WG Chair: Roger B. Marks
        Postal: (U.S.) National Institute
                of Standards and Technology
        E-mail: r.b.marks@ieee.org

        TGf Chair: Phillip Barber
        Postal: Huawei Technologies Co., Ltd
        E-mail: pbarber@futurewei.com

        Editor: Joey Chou
        Postal: Intel Corporation
                5000 W. Chandler Blvd,
                Chandler, AZ 85227, USA
        E-mail: joey.chou@intel.com"
```

DESCRIPTION

```

    "This material is from IEEE Std 802.16i
    Copyright (c) 2006 IEEE.
    This MIB Module defines managed objects for
    Subscriber Station and Base Station based on IEEE Std
    802.16-2004 and its amendment IEEE Std 802.16e-2005.
    The MIB contains managed objects that are specific
    to mobile Broadband Wireless Networks."
```

```

    REVISION          "200610160000Z"
```

DESCRIPTION

"The 1st revision of WMAN-IF2M-MIB module."
 ::= { ??? ??? }

END

Annex E.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

Annex F. Proposal for Adding Mobility Handover and Paging group MIBs

1. Introduction

With the mobility feature introduced, handover between BS and its neighbouring BS is inevitable. This contribution proposes to add BS handover related parameters which will help to execute smoothly handover.

Paging group configuration is also very important in the mobility scenario, proper paging group settings will make the paging procedure simple and effective. The configuration of paging group is also included in this contribution.

2. Proposed Text Introduction

2.1 wmanIfBsObjects

2.1.1 wmanIfBsMobility

2.1.1.1 wmanIfBsHandoverConfiguration

wmanIfBsHandoverConfiguration contains handover related parameters. Handover related parameters include BS configuration parameters and its neighbouring BSes configuration parameters.

2.1.1.2 wmanIfBsPagingGroupTable

wmanIfBsPagingGroupTable contains paging group related parameters

3. ASN.1 Definitions of 802.16 MIB for SNMP

```

wman2IfMibObjects OBJECT IDENTIFIER ::= { wman2IfMib 1 }
wman2IfBsObjects OBJECT IDENTIFIER ::= { wman2IfMibObjects 1 }
wman2IfMsObjects OBJECT IDENTIFIER ::= { wman2IfMibObjects 2 }
wman2IfCommonObjects OBJECT IDENTIFIER ::= { wman2IfMibObjects 3 }
wmanIfBsFm OBJECT IDENTIFIER ::= { wman2IfBsObjects x } -- e.g. x=0
wmanIfBsCm OBJECT IDENTIFIER ::= { wman2IfBsObjects x } -- e.g. x=7
wmanIfBsAm OBJECT IDENTIFIER ::= { wman2IfBsObjects x } -- e.g. x=0
wmanIfBsPm OBJECT IDENTIFIER ::= { wman2IfBsObjects x } -- e.g. x=0
wmanIfBsSm OBJECT IDENTIFIER ::= { wman2IfBsObjects x } -- e.g. x=0
wmanIfMsFm OBJECT IDENTIFIER ::= { wman2IfMsObjects x } -- e.g. x=0
wmanIfMsCm OBJECT IDENTIFIER ::= { wman2IfMsObjects x } -- e.g. x=0
wmanIfMsAm OBJECT IDENTIFIER ::= { wman2IfMsObjects x } -- e.g. x=0
wmanIfMsPm OBJECT IDENTIFIER ::= { wman2IfMsObjects x } -- e.g. x=0
wmanIfMsSm OBJECT IDENTIFIER ::= { wman2IfMsObjects x } -- e.g. x=0
wmanIfBsCmHOConfiguration OBJECT IDENTIFIER ::= { wmanIfBsCm 2 }

wmanIfBsOperatorId OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "An unique operator identifier."
    ::= { wmanIfBsHandoverConfiguration 1 }

wmanIfBsId OBJECT-TYPE
    SYNTAX WmanIfBsIdType
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "An unique BS identifier."
    ::= { wmanIfBsHandoverConfiguration 2 }

wmanIfBsHandoverSupport OBJECT-TYPE
    SYNTAX BITS
        {
            MDHO/FBSS HO not supported(0),
            FBSS/MDHO DLRf combining supported(1),
            MDHO DL soft combining supported monitoring single MAP from anchor
            BS(2),
            MDHO DL soft combining supported monitoring MAPs from active BSs(3),
            reserved1(5),
            reserved2(6),
            reserved3(7)
        }
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "The Handover supported field indicates what type(s) of HO the BS and the MS
        supports."
    ::= { wmanIfBsHandoverConfiguration 3 }

```

```

1      wmanIfBsHandoverSupport OBJECT-TYPE
2          SYNTAX BITS
3              {
4                  mdho/fbss HO not supported(0),
5                  fbss/mdho DLRF combining supported(1),
6                  mdho DL soft combining supported monitoring single MAP from anchor BS(2),
7                  mdho DL soft combining supported monitoring MAPS from active BSs(3)
8              }
9
10         MAX-ACCESS read-write
11         STATUS current
12         DESCRIPTION
13             "The Handover supported field indicates what type(s) of HO the BS and the MS
14 supports."
15         ::= { wmanIfBsHandoverConfiguration 3 }
16
17
18
19      wmanIfBsResourceRetainTime OBJECT-TYPE
20          SYNTAX Integer32
21          MAX-ACCESS read-write
22          STATUS current
23          DESCRIPTION
24              "The Resource_Retain_Time is the duration for MS s connection information
25 that will be retained in serving BS. BS shall start Resource_Retain_Time timer at MS notification
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

```

of pending HO attempt through MOB_HO-IND or by detecting an MS drop. The unit of this value is 100 milliseconds."

::= { wmanIfBsHandoverConfiguration 4 }

wmanIfBsHOProcessOptimizationMSTimer OBJECT-TYPE

SYNTAX INTEGER

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"the duration in frames MS shall wait until receipt of the next unsolicited network re-entry MAC management message as indicated in the HO Process Optimization element of the RNG-RSP message."

::= { wmanIfBsHandoverConfiguration 5 }

wmanIfBsMsHOREtransmissionTimer OBJECT-TYPE

SYNTAX INTEGER

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"After a MS transmits MOB_MSHO-REQ to initiate a handover process, it shall start MS Handover Retransmission Timer and shall not transmit another MOB_MSHO-REQ until the expiration of the MS Handover Retransmission Timer."

::= { wmanIfBsHandoverConfiguration 6 }

wmanIfBsMobilityModeSupport OBJECT-TYPE

SYNTAX BITS

{
handover support(0),
sleep-mode support(1),
idle-mode support(2)
}

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This parameter is to represent the supported mobility mode."

::= { wmanIfBsHandoverConfiguration 7 }

wmanIfBsMsHOCConnectProcessingTime OBJECT-TYPE

SYNTAX INTEGER

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Time in ms the MS needs to process information on connections provided in RNGRSP or REG-RSP message during HO."

::= { wmanIfBsHandoverConfiguration 8 }

wmanIfBsMsHoTekProcessingTime OBJECT-TYPE

SYNTAX INTEGER

MAX-ACCESS read-write

STATUS current

DESCRIPTION


```

1           "Time in ms the MS needs to completely
2           process TEK information during HO."
3       ::= { wmanIfBsHandoverConfiguration 9 }
4
5
6   wmanIfBsULPermutationBase OBJECT-TYPE
7       SYNTAX OCTET STRING
8       MAX-ACCESS read-write
9       STATUS current
10      DESCRIPTION
11          "This parameter is used for uplink subcarrier allocation."
12      ::= { wmanIfBsHandoverConfiguration 10 }
13
14
15   wmanIfBsDLPermutationBase OBJECT-TYPE
16       SYNTAX OCTET STRING
17       MAX-ACCESS read-write
18       STATUS current
19       DESCRIPTION
20          "This parameter is used for downlink subcarrier allocation."
21      ::= { wmanIfBsHandoverConfiguration 11 }
22
23
24
25   wmanIfBsPreambleIndex OBJECT-TYPE
26       SYNTAX OCTET STRING
27       MAX-ACCESS read-write
28       STATUS current
29       DESCRIPTION
30          "This parameter is used for downlink synchronization by MS."
31      ::= { wmanIfBsHandoverConfiguration 12 }
32
33
34
35   wmanIfBsSegmentNumber OBJECT-TYPE
36       SYNTAX INTEGER
37       MAX-ACCESS read-write
38       STATUS current
39       DESCRIPTION
40          "This parameter is an unique segment identifier ."
41      ::= { wmanIfBsHandoverConfiguration 13 }
42
43
44
45   wmanIfNeighbourBsTable OBJECT-TYPE
46       SYNTAX SEQUENCE OF WmanIfNeighbourBsEntry
47       MAX-ACCESS not-accessible
48       STATUS current
49       DESCRIPTION
50          "This table contains neighbouring BS related parameters."
51      ::= { wmanIfBsHandoverConfiguration 14 }
52
53
54   wmanIfNeighbourBsEntry OBJECT-TYPE
55       SYNTAX WmanIfNeighbourBsEntry
56       MAX-ACCESS not-accessible
57       STATUS current
58       DESCRIPTION
59          "This table is indexed by wmanIfNeighbourBsId."
60       INDEX { ifIndex, wmanIfNeighbourBsId }
61      ::= { wmanIfNeighbourBsTable 1 }
62
63
64   wmanIfNeighbourBsEntry ::= SEQUENCE {
65

```

```

1      wmanIfNeighbourBsId          WmanIfBsIdType,
2      wmanIfNeighbourBsFAIndex INTEGER,
3      wmanIfNeighbourBsEIRP        INTEGER (-128..127),
4      wmanIfNeighbourBsHOPProcessOptimizationInteger32,
5      wmanIfNeighbourBsSchedulingServiceSupportBITS,
6      wmanIfNeighbourBsBandwidthInteger32,
7      wmanIfNeighbourBsFFTSize     Integer32,
8      wmanIfNeighbourBsCyclePrefixInteger32,
9      wmanIfNeighbourBsFrameDurationCodeInteger32,
10     wmanIfNeighbourBsULPermutationBaseInteger32,
11     wmanIfNeighbourBsDLPermutationBaseInteger32,
12     wmanIfNeighbourBsSegmentNumberInteger32,
13     wmanIfNeighbourBsPreambleIndexInteger32
14 }
15
16
17
18
19     wmanIfNeighbourBsId OBJECT-TYPE
20         SYNTAX WmanIfBsIdType
21         MAX-ACCESS read-write
22         STATUS current
23         DESCRIPTION
24             "The neighbouring BS identifier."
25             ::= { wmanIfNeighbourBsEntry 1 }
26
27
28
29     wmanIfNeighbourBsFAIndex OBJECT-TYPE
30         SYNTAX INTEGER
31         MAX-ACCESS read-write
32         STATUS current
33         DESCRIPTION
34             "Frequency Assignment Index."
35             ::= { wmanIfNeighbourBsEntry 2 }
36
37
38
39     wmanIfNeighbourBsEIRP OBJECT-TYPE
40         SYNTAX INTEGER (-128..127)
41         MAX-ACCESS read-write
42         STATUS current
43         DESCRIPTION
44             "Neighbour BS EIRP."
45             ::= { wmanIfNeighbourBsEntry 3 }
46
47
48
49     wmanIfNeighbourBsHOPProcessOptimization OBJECT-TYPE
50         SYNTAX Integer32
51         MAX-ACCESS read-write
52         STATUS current
53         DESCRIPTION
54             "Identifies re-entry process management messages that may be omitted during
55             the current HO attempt due to the availability of MS service and operational context information,
56             and the MS service and operational status post-HO completion."
57             ::= { wmanIfNeighbourBsEntry 4 }
58
59
60
61     wmanIfNeighbourBsSchedulingServiceSupport OBJECT-TYPE
62         SYNTAX BITS
63         {
64             real-time polling service(0),
65             extended real-time polling service(1),

```

```

1          non-real-time polling service(2),
2          unsolicited grant service(3),
3          best effort(4)
4      }
5      MAX-ACCESS read-write
6      STATUS current
7      DESCRIPTION
8          "This parameter is used to indicate neighbouring BS scheduling service type."
9      ::= { wmanIfNeighbourBsEntry 5 }
10
11
12
13      wmanIfNeighbourBsBandwidth OBJECT-TYPE
14          SYNTAX Integer32
15          MAX-ACCESS read-write
16          STATUS current
17          DESCRIPTION
18              "This parameter is used to indicate neighbouring BS bandwidth."
19          ::= { wmanIfNeighbourBsEntry 6 }
20
21
22
23      wmanIfNeighbourBsFFTSIZE OBJECT-TYPE
24          SYNTAX Integer32
25          MAX-ACCESS read-write
26          STATUS current
27          DESCRIPTION
28              "This parameter is used to indicate neighbouring BS FFT size."
29          ::= { wmanIfNeighbourBsEntry 7 }
30
31
32
33      wmanIfNeighbourBsCyclePrefix OBJECT-TYPE
34          SYNTAX Integer32
35          MAX-ACCESS read-write
36          STATUS current
37          DESCRIPTION
38              "This parameter is used to indicate neighbouring BS Cycle prefix."
39          ::= { wmanIfNeighbourBsEntry 8 }
40
41
42
43      wmanIfNeighbourBsFrameDurationCode OBJECT-TYPE
44          SYNTAX Integer32
45          MAX-ACCESS read-write
46          STATUS current
47          DESCRIPTION
48              "This parameter is used to indicate neighbouring BS Frame duration code."
49          ::= { wmanIfNeighbourBsEntry 9 }
50
51
52
53      wmanIfNeighbourBsULPermutationBase OBJECT-TYPE
54          SYNTAX Integer32
55          MAX-ACCESS read-write
56          STATUS current
57          DESCRIPTION
58              "This parameter is used to indicate neighbouring BS uplink permutation base."
59          ::= { wmanIfNeighbourBsEntry 10 }
60
61
62
63      wmanIfNeighbourBsDLPermutationBase OBJECT-TYPE
64          SYNTAX Integer32
65          MAX-ACCESS read-write
66          STATUS current

```

```

1      DESCRIPTION
2          "This parameter is used to indicate neighbouring BS downlink permutation
3      base."
4          ::= { wmanIfNeighbourBsEntry 11 }
5
6
7      wmanIfNeighbourBsSegmentNumber OBJECT-TYPE
8          SYNTAX Integer32
9          MAX-ACCESS read-write
10         STATUS current
11         DESCRIPTION
12             "This parameter is used to indicate neighbouring BS segment number."
13             ::= { wmanIfNeighbourBsEntry 12 }
14
15
16
17     wmanIfNeighbourBsPreambleIndex OBJECT-TYPE
18         SYNTAX Integer32
19         MAX-ACCESS read-write
20         STATUS current
21         DESCRIPTION
22             "This parameter is used to indicate neighbouring BS preamble index."
23             ::= { wmanIfNeighbourBsEntry 13 }
24
25
26
27     wmanIfBsPagingGroupTable OBJECT-TYPE
28         SYNTAX SEQUENCE OF WmanIfBsPagingGroupEntry
29         MAX-ACCESS not-accessible
30         STATUS current
31         DESCRIPTION
32             "This table contains paging group related parameters."
33             ::= { wmanIfBsMobility 3 }
34
35
36
37     wmanIfBsPagingGroupEntry OBJECT-TYPE
38         SYNTAX WmanIfBsPagingGroupEntry
39         MAX-ACCESS not-accessible
40         STATUS current
41         DESCRIPTION
42             "This table is indexed by wmanIfBsPagingGroupId."
43         INDEX { wmanIfBsPagingGroupId }
44         ::= { wmanIfBsPagingGroupTable 1 }
45
46
47     wmanIfBsPagingGroupEntry ::= SEQUENCE {
48         wmanIfBsPagingControlId          IpAddress,
49         wmanIfBsPagingGroupId            INTEGER,
50         wmanIfBsMgmtResourceHoldingTimerInteger32,
51         wmanIfBsT46Timer                  Integer32,
52         wmanIfBsPagingRetryCount          INTEGER,
53         wmanIfBsREQDuration               INTEGER,
54         wmanIfBsMACHashSkipThresholdInteger32,
55         wmanIfBsCDMATransmissionOpportunityAssignmentINTEGER,
56         wmanIfBsPagingResponseWindow     INTEGER,
57         wmanIfBsIdleModeTimer             INTEGER,
58         wmanIfBsIdleModeSystemTimer      INTEGER,
59         wmanIfBsPagingIntervalLength     INTEGER,
60         wmanIfBsPagingCycle               INTEGER
61     }
62
63
64
65

```

```

1      wmanIfBsPagingControlId OBJECT-TYPE
2          SYNTAX IpAddress
3          MAX-ACCESS read-write
4          STATUS current
5          DESCRIPTION
6              "This parameter is used to indicate paging controller identifier connected by BS."
7              ::= { wmanIfBsPagingGroupEntry 1 }
8
9
10     wmanIfBsPagingGroupId OBJECT-TYPE
11         SYNTAX INTEGER
12         MAX-ACCESS read-write
13         STATUS current
14         DESCRIPTION
15             "This parameter is used to indicate the paging group identifier assigned to BS by
16             network."
17             ::= { wmanIfBsPagingGroupEntry 2 }
18
19
20
21     wmanIfBsMgmtResourceHoldingTimer OBJECT-TYPE
22         SYNTAX Integer32
23         MAX-ACCESS read-write
24         STATUS current
25         DESCRIPTION
26             "Time the BS maintain connection
27             information with the MS after the
28             BS send DREG-CMD to the MS"
29             ::= { wmanIfBsPagingGroupEntry 3 }
30
31
32
33     wmanIfBsT46Timer OBJECT-TYPE
34         SYNTAX Integer32
35         MAX-ACCESS read-write
36         STATUS current
37         DESCRIPTION
38             "Time the BS waits for DREGREQ
39             in case of unsolicited Idle
40             Mode initiation from BS."
41             ::= { wmanIfBsPagingGroupEntry 4 }
42
43
44
45     wmanIfBsPagingRetryCount OBJECT-TYPE
46         SYNTAX INTEGER
47         MAX-ACCESS read-write
48         STATUS current
49         DESCRIPTION
50             "Number of retries on paging
51             transmission. If the BS does not
52             receive RNG-REQ from the MS
53             until this value decreases to zero,
54             it determines that the MS is
55             unavailable."
56             ::= { wmanIfBsPagingGroupEntry 5 }
57
58
59
60
61     wmanIfBsREQDuration OBJECT-TYPE
62         SYNTAX INTEGER
63         MAX-ACCESS read-write
64         STATUS current
65

```

```

1      DESCRIPTION
2          "Waiting value for the DREG-REQ message re-transmission
3          (measured in frames)."
4      ::= { wmanIfBsPagingGroupEntry 6 }
5
6
7      wmanIfBsMACHashSkipThreshold OBJECT-TYPE
8          SYNTAX Integer32
9          MAX-ACCESS read-write
10         STATUS current
11         DESCRIPTION
12             "Maximum number of successive MOB_PAG-ADV messages
13             that may be sent from a BS without individual notification for
14             an MS for which BS is allowed to skip MS MAC Address Hash
15             when the Action Code for the MS is 0b00,'No Action Required'."
16         ::= { wmanIfBsPagingGroupEntry 7 }
17
18
19
20     wmanIfBsCDMATransmissionOpportunityAssignment OBJECT-TYPE
21         SYNTAX INTEGER
22         MAX-ACCESS read-write
23         STATUS current
24         DESCRIPTION
25             "The CDMA code and transmission opportunity
26             assignment field indicates the assigned code
27             and transmission opportunity for a MS who is
28             paged to use over dedicated CDMA ranging region."
29         ::= { wmanIfBsPagingGroupEntry 8 }
30
31
32
33     wmanIfBsPagingResponseWindow OBJECT-TYPE
34         SYNTAX INTEGER
35         MAX-ACCESS read-write
36         STATUS current
37         DESCRIPTION
38             "The Page-Response Window indicates the Page-Response window for a MS
39             who is paged to transmit
40             the assigned code for CDMA ranging channel."
41         ::= { wmanIfBsPagingGroupEntry 9 }
42
43
44
45     wmanIfBsIdleModeTimer OBJECT-TYPE
46         SYNTAX INTEGER (128..65536)
47         MAX-ACCESS read-write
48         STATUS current
49         DESCRIPTION
50             "MS timed interval to conduct
51             Location Update. Set timer to MS
52             Idle Mode Timeout capabilities
53             setting. Timer recycles on successful
54             Idle Mode Location Update."
55         ::= { wmanIfBsPagingGroupEntry 10 }
56
57
58
59     wmanIfBsIdleModeSystemTimer OBJECT-TYPE
60         SYNTAX INTEGER (128..65536)
61         MAX-ACCESS read-write
62         STATUS current
63         DESCRIPTION
64
65

```

"For BS acting as Paging Controller,
 timed interval to receive notification
 of MS Idle Mode Location Update. Set
 timer to MS Idle Mode Timeout. Timer
 recycles on successful Idle Mode
 Location Update."
 ::= { wmanIfBsPagingGroupEntry 11 }

wmanIfBsPagingIntervalLength OBJECT-TYPE
 SYNTAX INTEGER (2..5)
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "time duration of Paging Interval
 of the BS."
 ::= { wmanIfBsPagingGroupEntry 12 }

wmanIfBsPagingCycle OBJECT-TYPE
 SYNTAX INTEGER
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "Cycle in which the paging message is transmitted
 within the paging group."
 ::= { wmanIfBsPagingGroupEntry 13 }

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65