

2006-08-08

IEEE 802.16i-06/001r3, August 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, *802.16f Chief Technical Editor*

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 wman2IfMib	49
15.2.1.1 wman2IfBsObjects.....	50
15.2.1.2 wman2IfSsObjects	56
15.2.1.3 wman2IfCommonObjects	59
15.2.2 ASN.1 Definitions of 802.16 MIB for SNMP	61
Annex E.	278
Annex F. Proposal for Adding Mobility Handover and Paging group MIBs	279
1. Introduction.....	279
2. Proposed Text Introduction	279
2.1 wmanIfBsObjects.....	279
2.1.1 wmanIfBsMobility	279
2.1.1.1 wmanIfBsHandoverConfiguration.....	279
2.1.1.2 wmanIfBsPagingGroupTable	279
3. ASN.1 Definitions of 802.16 MIB for SNMP	280

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—wman2IfMib structure	50
Figure 19—wman2IfBsPacketCs structure	50
Figure 20—wman2IfBsCps structure	51
Figure 21—wman2IfBsPkmObjects structure.....	53
Figure 22—wman2IfBsNotification structure	53
Figure 23—wman2IfBsPhy structure	54
Figure 24—wman2IfSsCps structure.....	56
Figure 25—wman2IfSsPkmObjects structure	57
Figure 26—wman2IfSsNotification structure	57
Figure 27—wman2IfSsPhy structure.....	58
Figure 28—wman2IfCmnPacketCs structure	59
Figure 29—wman2IfCmnCps structure.....	59
Figure 30—wman2IfCmnPkmObjects structure	60

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.

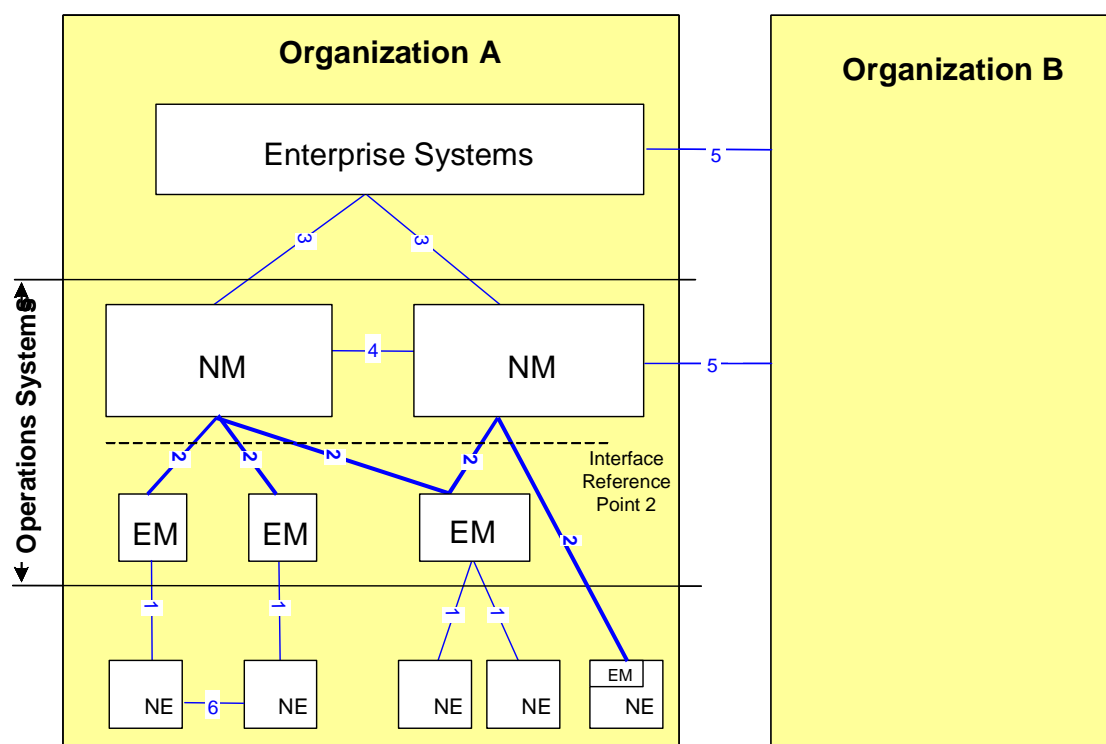


Figure 1—Mobile BWA Network Management Reference Model

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.

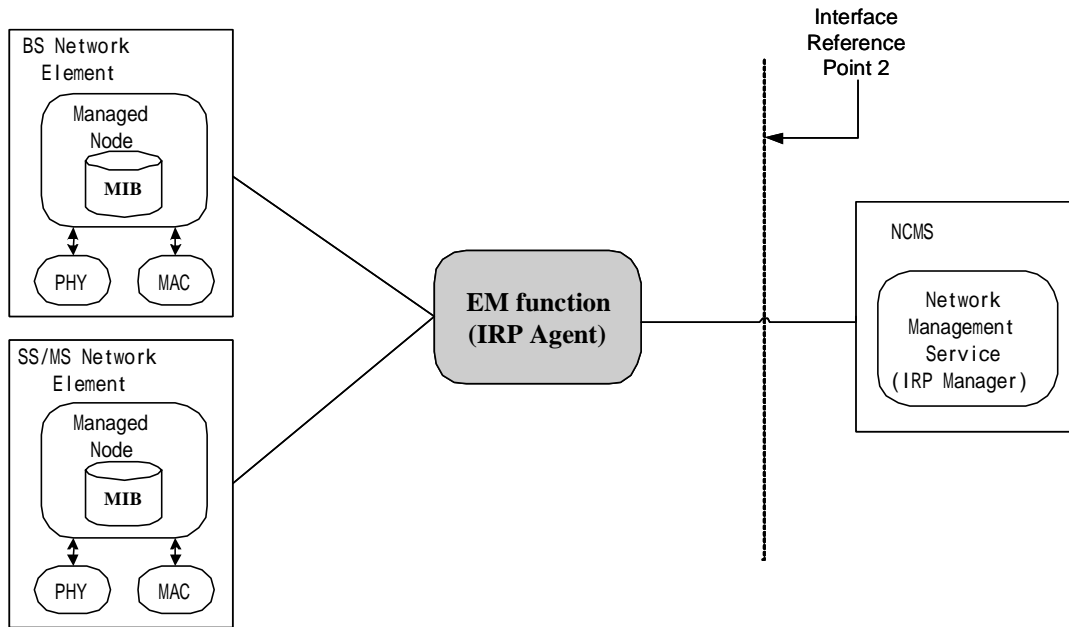


Figure 2—Mobile BWA Network Management Architecture - Context A

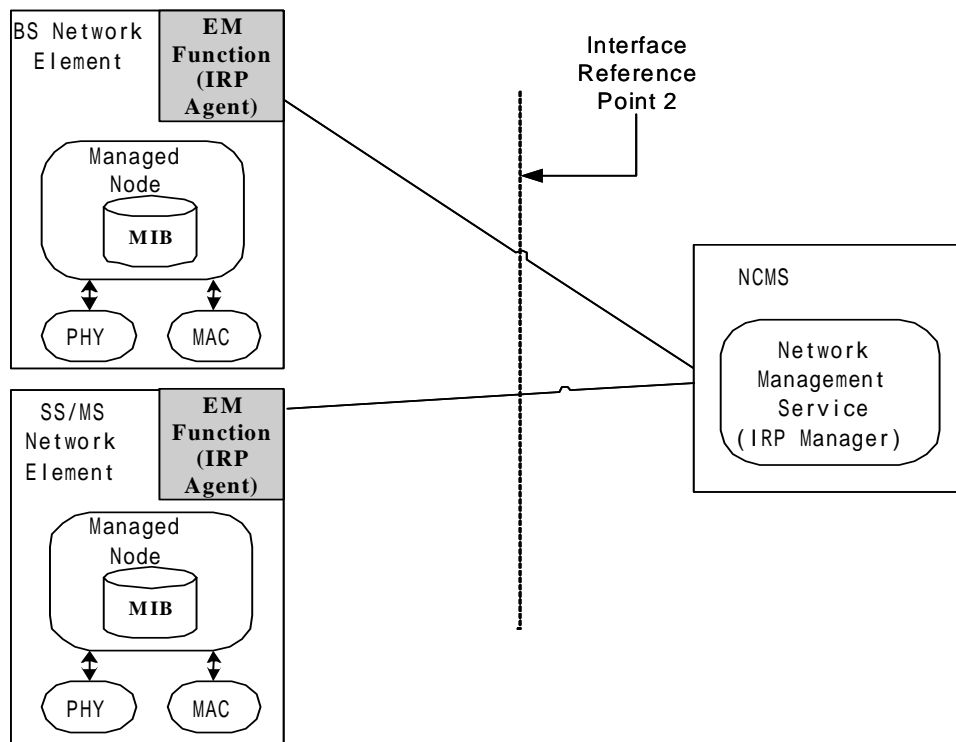


Figure 3—Mobile BWA Network Management Architecture - Context B

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
20 | RDN Relative Distinguished Name
21

22 | RP Integration Reference Point
23

24 | IS Information Service
25

26 | NRM Network Resource Model
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

wman2IfMib is located under MIB-II subtree. A submission will be sent to the Internet Assigned Numbers Authority (IANA) to assign ieee80216WMAN for wman2IfMib.

```

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

```

Pending on IETF approval, wman2IfMib 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.

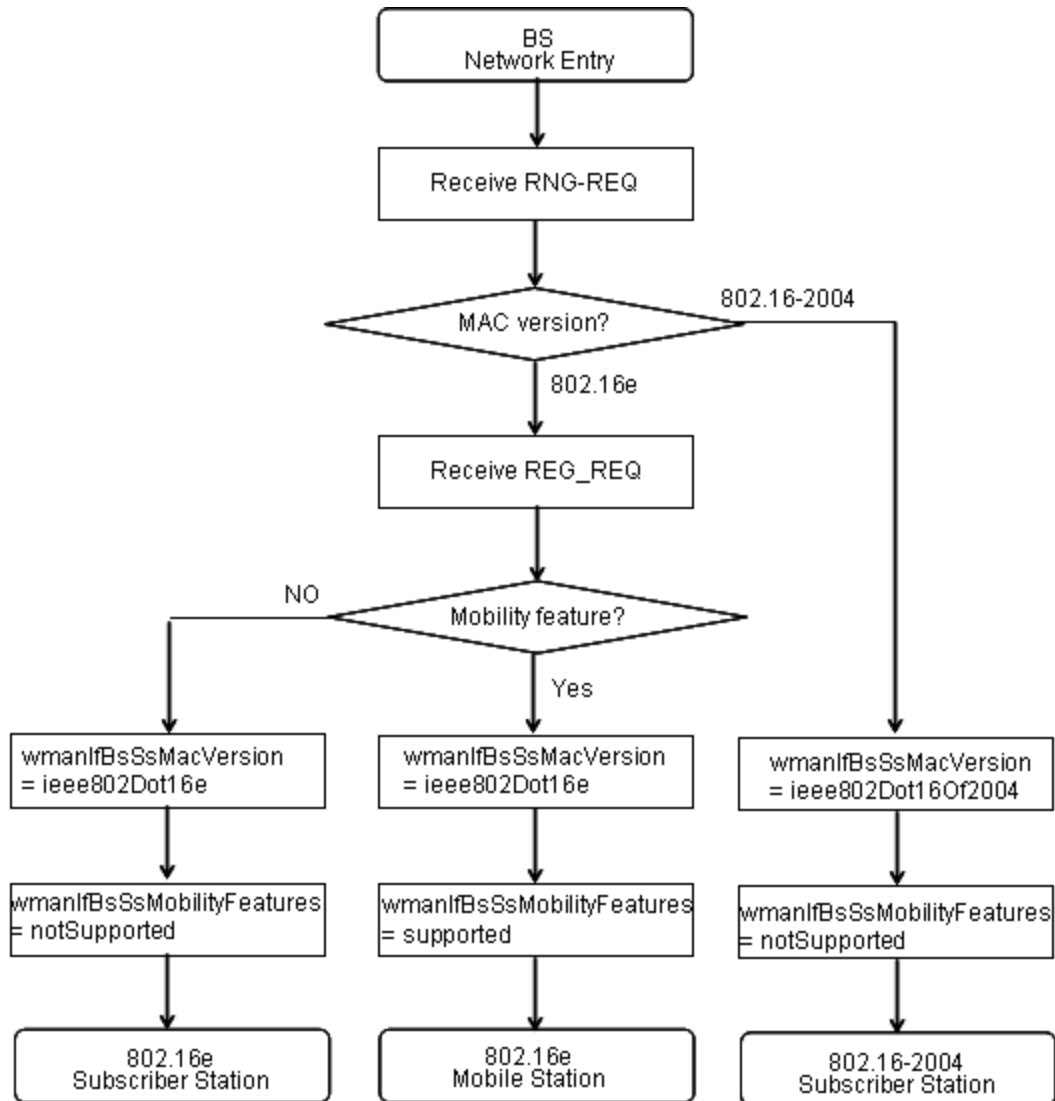


Figure 4—SS / MS Network Entry

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

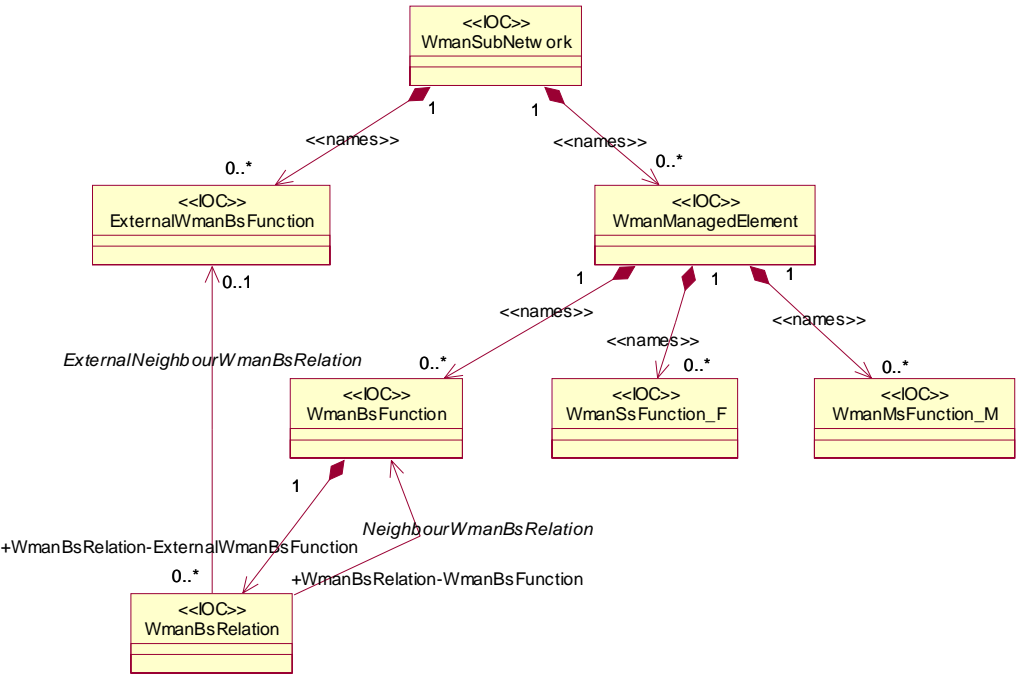


Figure 5—WmanSubNetwork Containment/Naming and Association Diagram

15.1.1.2.1.2 Bs Object Relationships

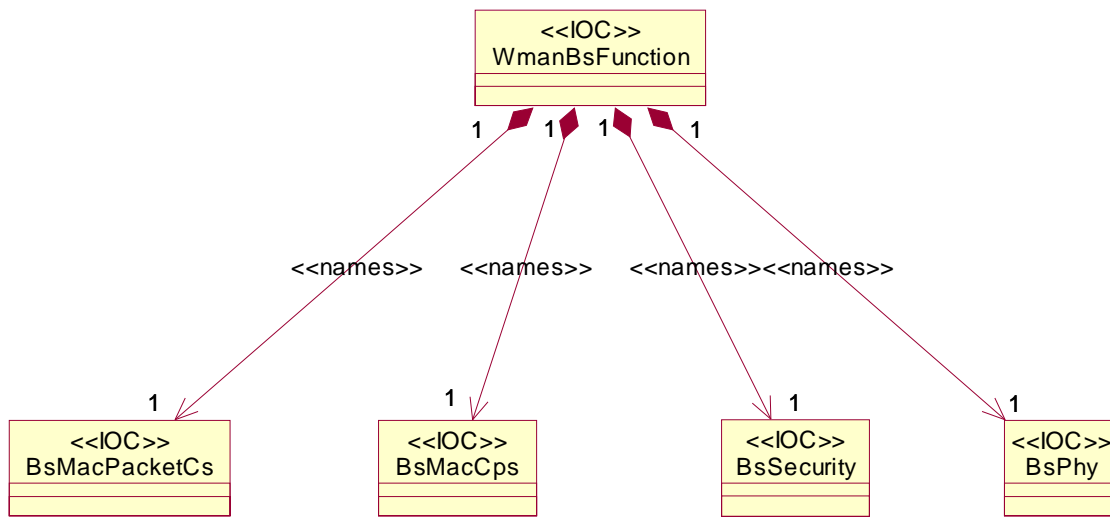


Figure 6—WmanBsFunction Containment/Naming and Association Diagram

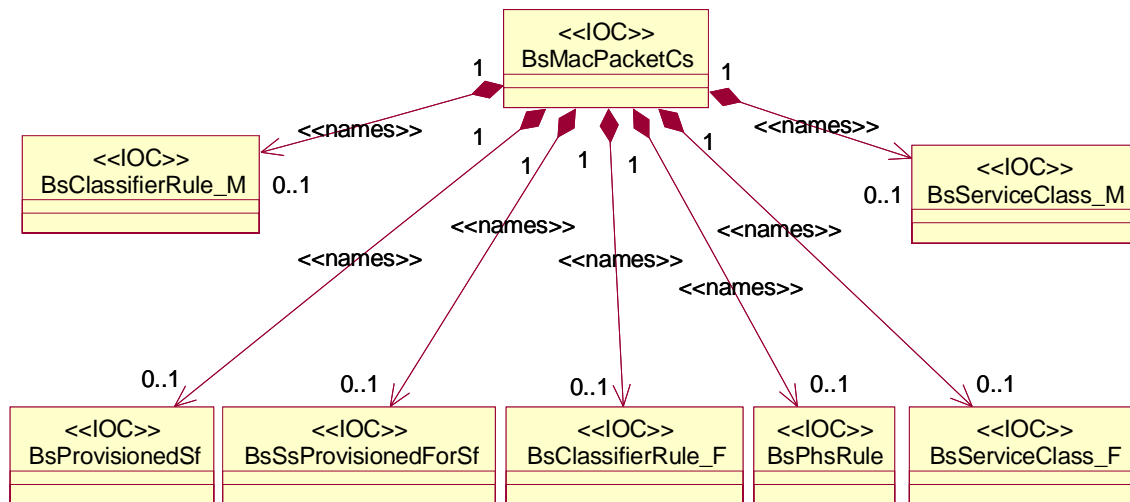


Figure 7—BsMacPacketCs Containment/Naming and Association Diagram

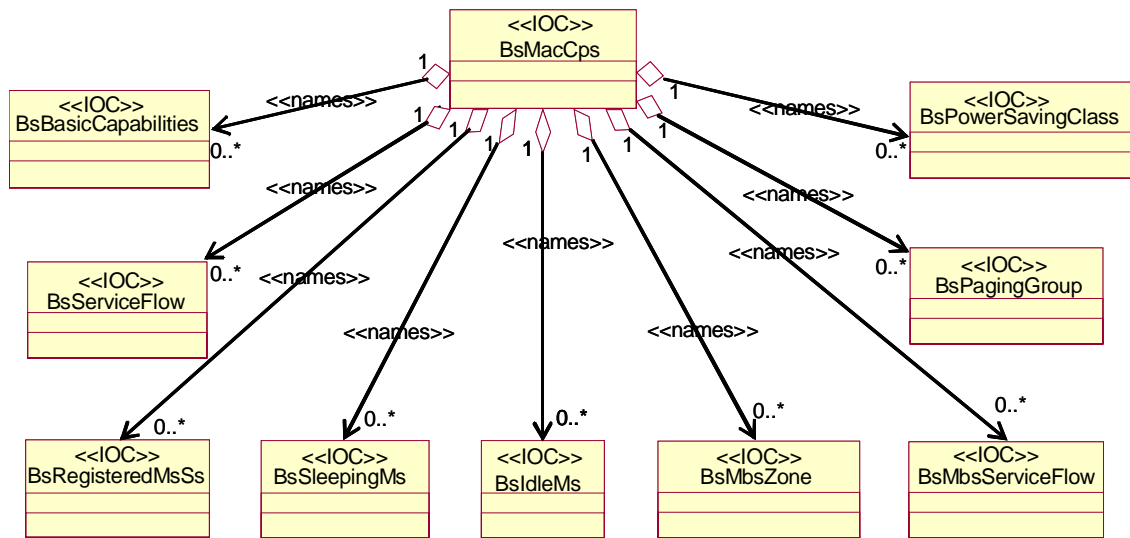


Figure 8—BsMacCps Containment/Naming and Association Diagram

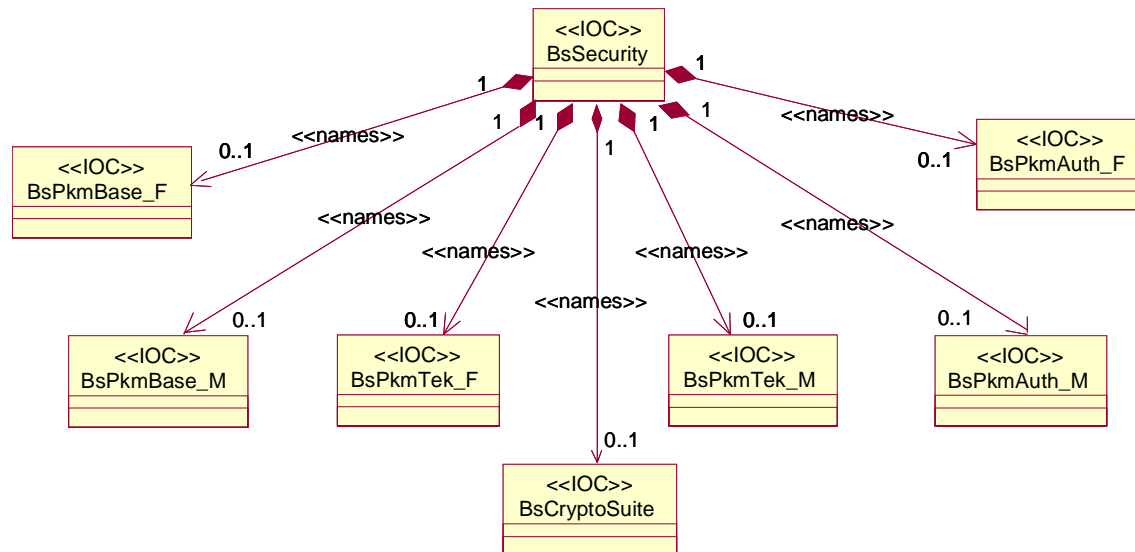


Figure 9—BsSecurity Containment/Naming and Association Diagram

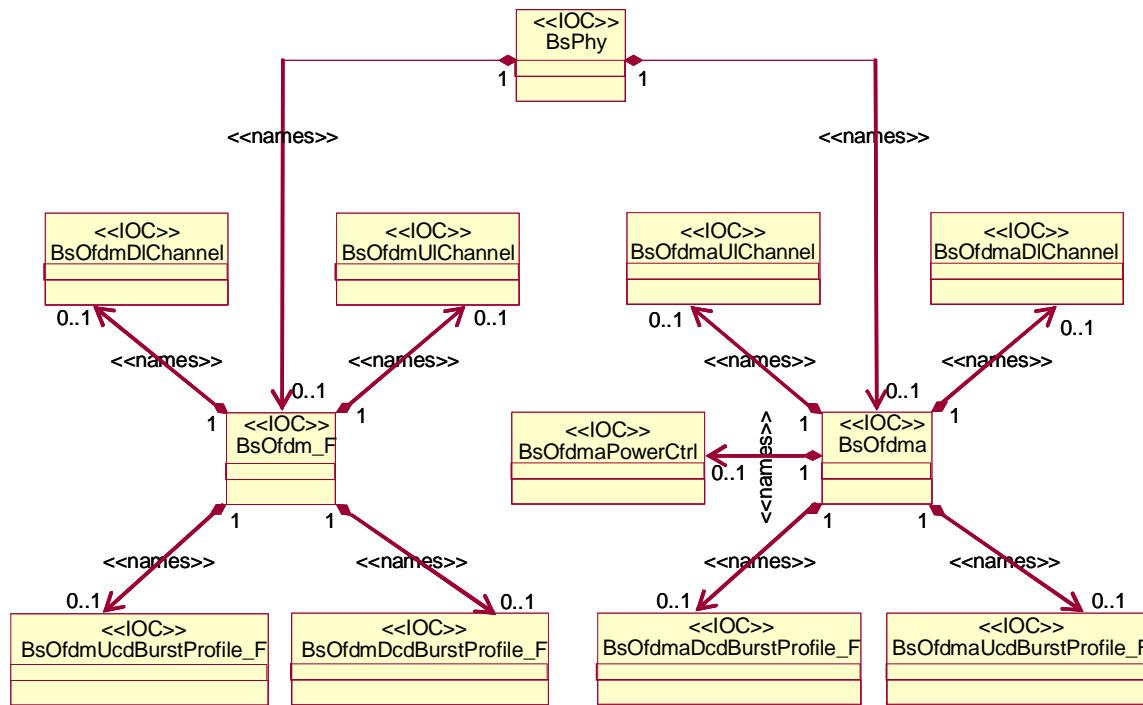


Figure 10—BsPhy Containment/Naming and Association Diagram

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

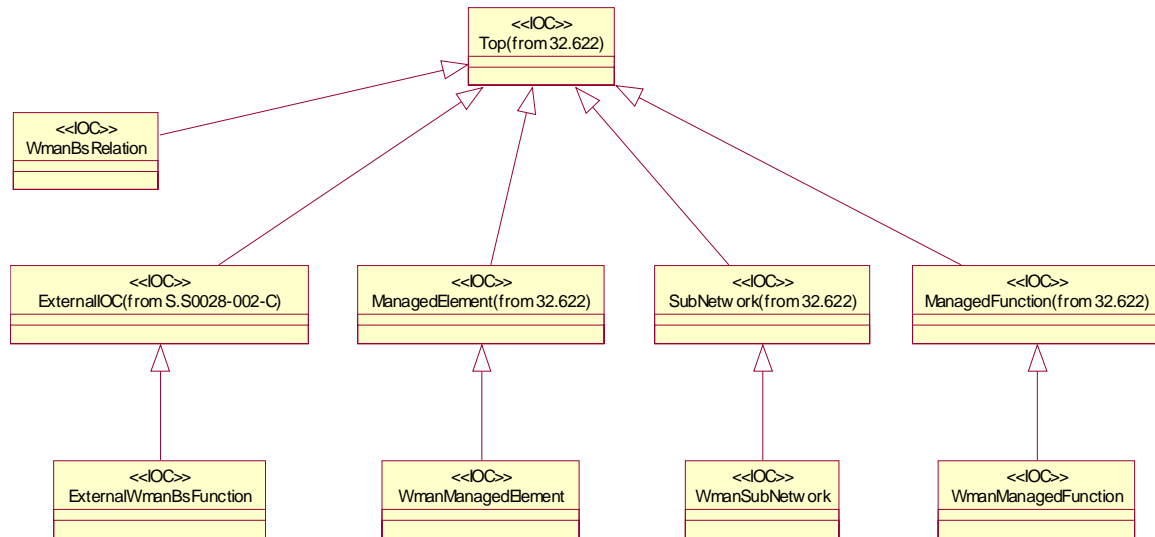


Figure 11—Top Inheritance Hierarchy Diagram

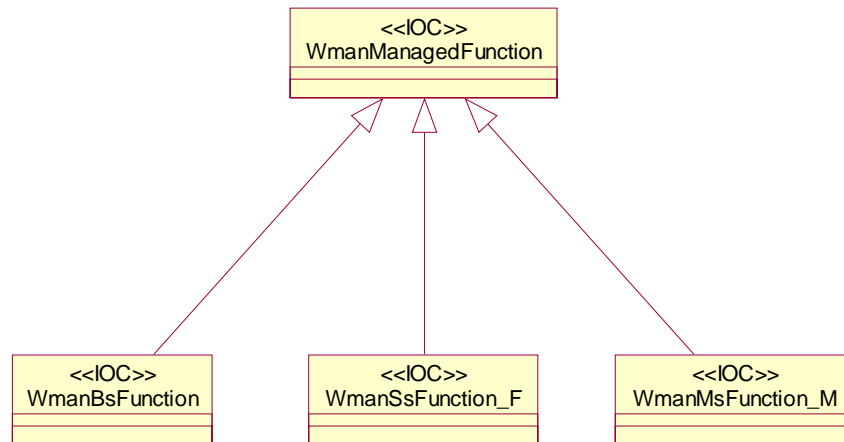


Figure 12—WmanManagedFunction Inheritance Hierarchy Diagram

15.1.1.2.2.2 Bs Object Inheritance

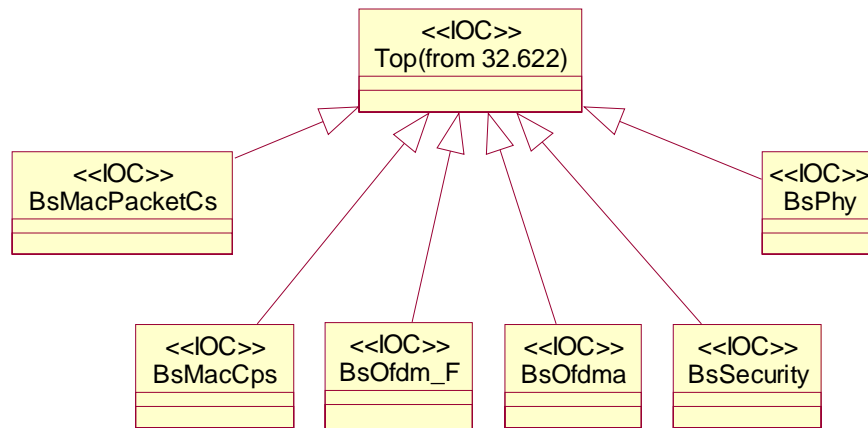


Figure 13—Bs Inheritance Hierarchy Diagram

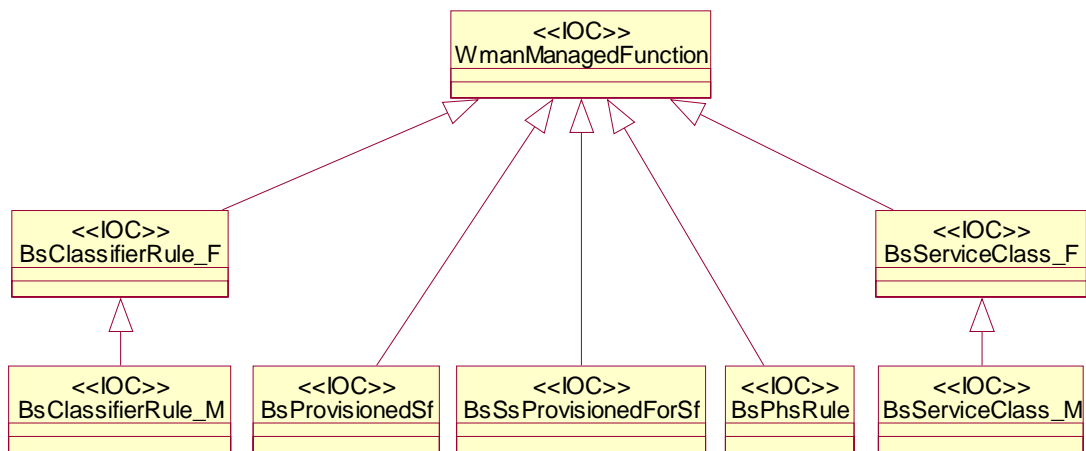


Figure 14—Bs PacketCs Inheritance Hierarchy Diagram

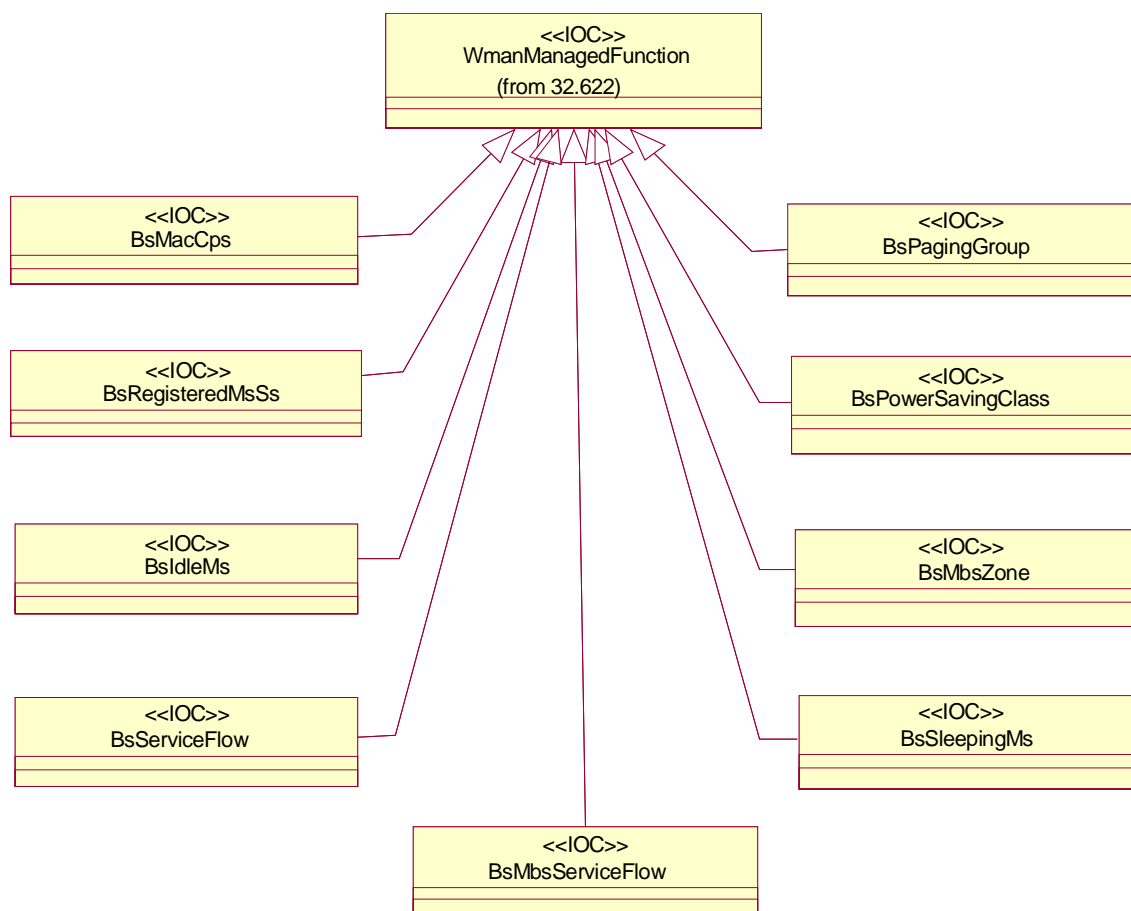


Figure 15—Bs MacCps Inheritance Hierarchy Diagram

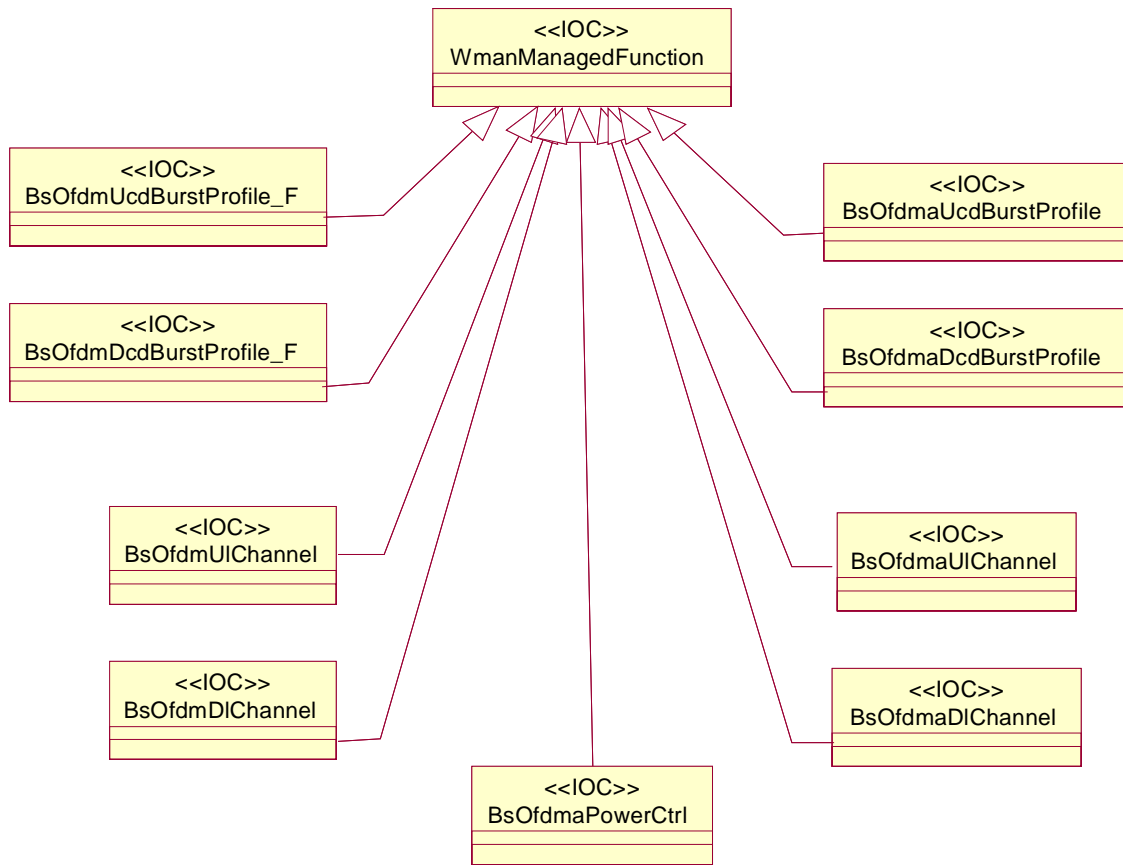


Figure 16—Bs Phy Inheritance Hierarchy Diagram

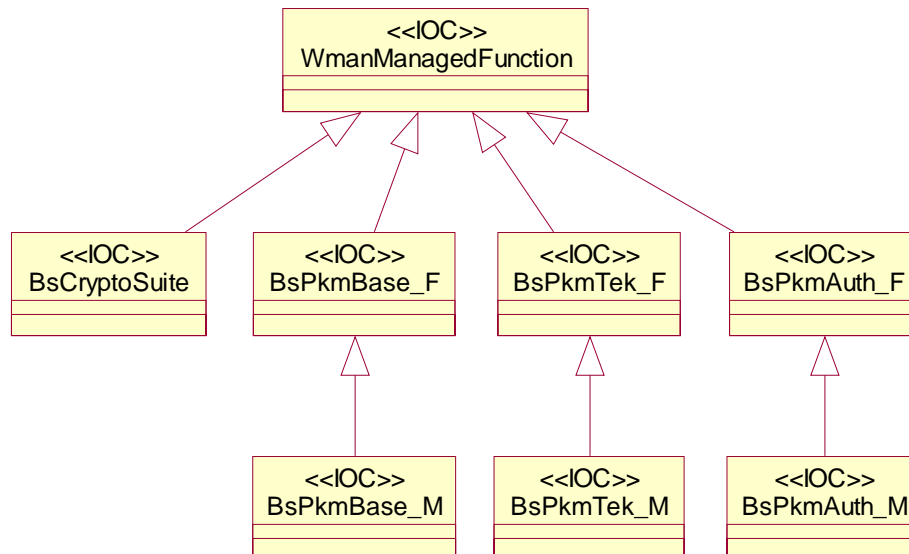


Figure 17—Bs Security Inheritance Hierarchy Diagram

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 wman2IfMib

Figure 19 shows the high level MIB structure of wman2IfMib for 802.16. The MIB structure is organized based on the the reference model as defined in IEEE 802.16-2004 standard.

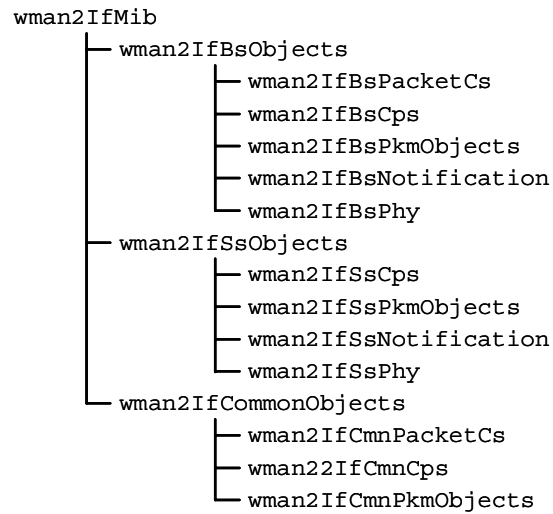


Figure 18—wman2IfMib structure

15.2.1.1 wman2IfBsObjects

15.2.1.1.1 wman2IfBsPacketCs

Figure 20 shows the structure of wman2IfBsPacketCs subtree that contains BS managed objects related to the Packet CS management entity layer.

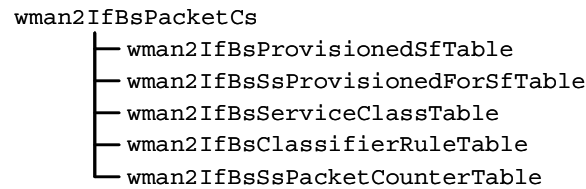


Figure 19—wman2IfBsPacketCs structure

15.2.1.1.1.1 wman2IfBsProvisionedSfTable

wman2IfBsProvisionedSfTable contains provisioned service flow profiles for SSs, and pointers to wman2IfBsServiceClassTable and wman2IfBsClassifierRuleTable for QoS parameters and classifier rules respectively.

15.2.1.1.1.2 wman2IfBsProvisionedForSfTable

wman2IfBsProvisionedForSfTable maps the MAC addresses of SSs to the service flows provisioned in wman2IfBsProvisionedSfTable. 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 wman2IfBsServiceClassTable

Each entry of the wman2IfBsServiceClassTable 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 wman2IfBsClassifierRuleTable

wman2IfBsClassifierRuleTable contains the packet classifier rules associated with service flows.

15.2.1.1.1.5 wman2IfBsSsPacketCounterTable

wman2IfBsSsPacketCounterTable 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 wman2IfBsCps

Figure 21 shows the structure of wman2IfBsCps subtree that contains BS managed objects related to the MAC CPS management entity layer.

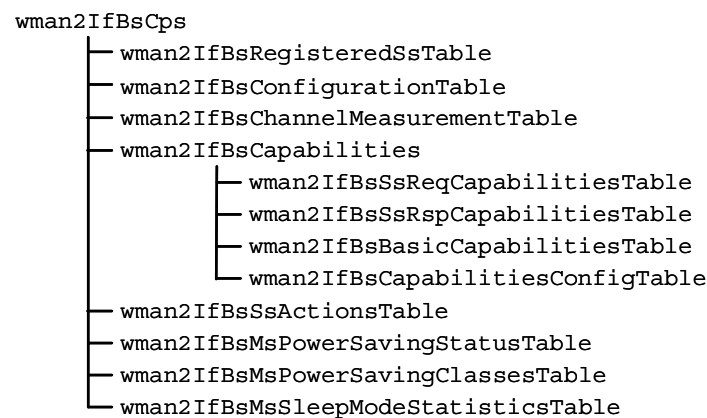


Figure 20—wman2IfBsCps structure

15.2.1.1.2.1 wman2IfBsRegisteredSsTable

Each entry in the wman2IfBsRegisteredSsTable contains the information of SS that has been registered through REG-REQ and REG-RSP messages.

15.2.1.1.2.2 wman2IfBsConfigurationTable

wman2IfBsConfigurationTable contains objects for BS system parameters and constants as defined in subclause 10.1 of IEEE 802.16-2004 standard. wman2IfBsConfigurationTable 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 wman2IfBsChannelMeasurementTable

wman2IfBsChannelMeasurementTable 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 wman2IfBsCapabilities

15.2.1.1.2.4.1 wman2IfBsSsReqCapabilitiesTable

wman2IfBsSsReqCapabilitiesTable 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 wman2IfBsSsRspCapabilitiesTable

wman2IfBsSsRspCapabilitiesTable 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 wman2IfBsBasicCapabilitiesTable

wman2IfBsBasicCapabilitiesTable contains the basic capabilities of the BS as implemented in BS hardware and software. These capabilities along with the configuration for them (wman2IfBsCapabilitiesConfigTable) 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 wman2IfBsCapabilitiesConfigTable

wman2IfBsCapabilitiesConfigTable 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 (wman2IfBsBasicCapabilitiesTable) for negotiation of basic capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP messages.

15.2.1.1.2.5 wman2IfBsSsActionsTable

wman2IfBsSsActionsTable 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 wmanIfMsBsPowerSavingStatusTable

wmanIfMsBsPowerSavingClassesTable contains the power saving status for each CID in an MS.

15.2.1.1.2.7 wmanIfMsBsPowerSavingClassesTable

wmanIfMsBsPowerSavingClassesTable 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 wmanIfBsMsSleepModeStatisticsTable

wmanIfMsBsPowerSavingClassesTable contains the sleep mode statistic for MS.

15.2.1.1.3 wman2IfBsPkmObjects

Figure 22 shows the structure of wman2IfBsPkmObjects subtree that contains BS managed objects related to the MAC privacy management entity.

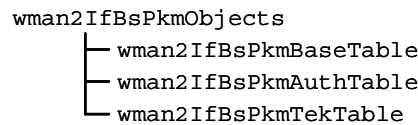


Figure 21—wman2IfBsPkmObjects structure

15.2.1.1.3.1 wman2IfBsPkmBaseTable

wman2IfBsPkmBaseTable contains base station PKM operational parameters described in subclause 10.2 of IEEE 802.16-2004 standard.

15.2.1.1.3.2 wman2IfBsSsPkmAuthTable

wman2IfBsSsPkmAuthTable contains runtime subscriber station authentication and authorization parameters for each base station.

15.2.1.1.3.3 wman2IfBsPkmTekTable

wman2IfBsPkmTekTable is double indexed by ifIndex and SAId and contains runtime Security association parameters for each base station.

15.2.1.1.4 wman2IfBsNotification

Figure 23 shows the structure of wman2IfBsNotification subtree that contains BS traps to report fault events and exceptions, such as power status, RSSI threshold crossing.

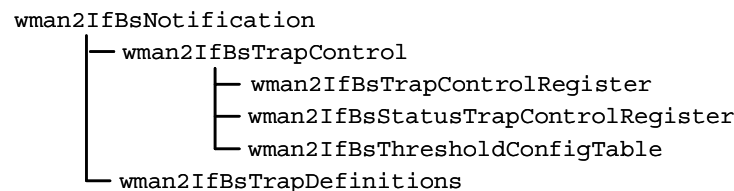


Figure 22—wman2IfBsNotification structure

15.2.1.1.4.1 wman2IfBsTrapControl

15.2.1.1.4.1.1 wman2IfBsTrapControlRegister

wman2IfBsTrapControlRegister is used to enable or disable Base traps independently.

15.2.1.1.4.1.2 wman2IfBsStatusTrapControlRegister

wman2IfBsStatusTrapControlRegister is used to enable or disable Base Station status notification traps.

15.2.1.1.4.1.3 wman2IfBsThresholdConfigTable

wman2IfBsThresholdConfigTable contains threshold objects that can be set to detect the threshold crossing events.

15.2.1.1.4.2 wman2IfBsTrapDefinitions

wman2IfBsTrapDefinitions object group defines all the traps reported by BS.

15.2.1.1.5 wman2IfBsPhy

Figure 24 shows the structure of wman2IfBsPhy subtree that contains BS managed objects related to the Physical layer.

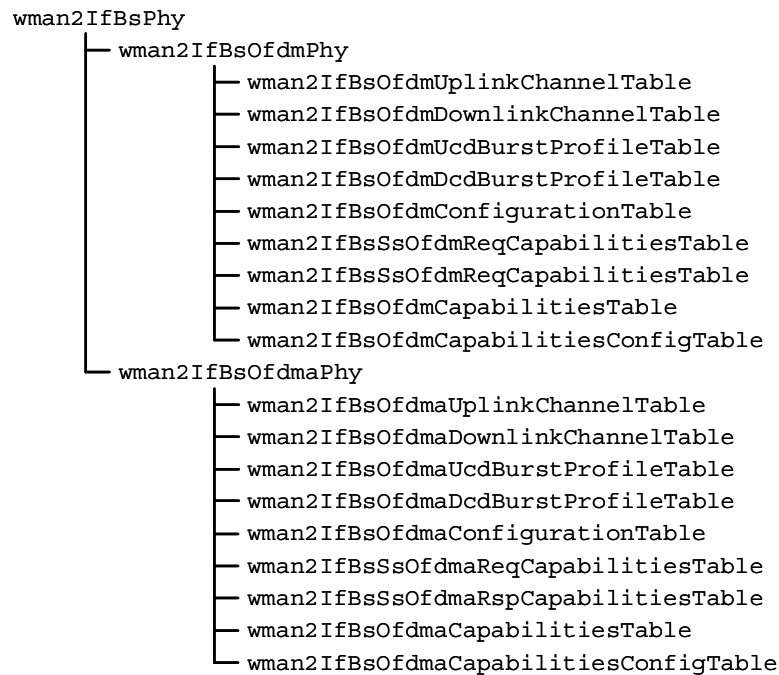


Figure 23—wman2IfBsPhy structure

15.2.1.1.5.1 wman2IfBsOfdmPhy

wman2IfBsOfdmPhy is a group containing objects specific to OFDM PHY.

15.2.1.1.5.1.1 wman2IfBsOfdmUplinkChannelTable

wman2IfBsOfdmUplinkChannelTable contains OFDM UCD (Uplink Channel Descriptor) channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.1.5.1.2 wman2IfBsOfdmDownlinkChannelTable

wman2IfBsOfdmDownlinkChannelTable contains OFDM DCD (Downlink Channel Descriptor) channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.1.5.1.3 wman2IfBsOfdmUcdBurstProfileTable

wman2IfBsOfdmUcdBurstProfileTable contains OFDM UCD burst profiles for each uplink channel.

15.2.1.1.5.1.4 wman2IfBsOfdmDcdBurstProfileTable

wman2IfBsOfdmDcdBurstProfileTable provides one row for each OFDM DCD burst profile.

15.2.1.1.5.1.5 wman2IfBsOfdmConfigurationTable

wman2IfBsOfdmConfigurationTable contains BS configuration objects, specific to OFDM PHY.

15.2.1.1.5.1.6 wman2IfBsSsOfdmReqCapabilitiesTable

wman2IfBsSsOfdmReqCapabilitiesTable 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 wman2IfBsSsOfdmRspCapabilitiesTable

wman2IfBsSsOfdmRspCapabilitiesTable 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 wman2IfBsRegisteredSsTable.

15.2.1.1.5.1.8 wman2IfBsOfdmCapabilitiesTable

wman2IfBsOfdmCapabilitiesTable 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 wman2IfBsOfdmCapabilitiesConfigTable

wman2IfBsOfdmCapabilitiesConfigTable 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.

15.2.1.1.5.2 wman2IfBsOfdmaPhy

wman2IfBsOfdmaPhy is a group containing objects specific to OFDMA PHY.

15.2.1.1.5.2.1 wman2IfBsOfdmaUplinkChannelTable

wman2IfBsOfdmaUplinkChannelTable contains OFDMA UCD channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.1.5.2.2 wman2IfBsOfdmaDownlinkChannelTable

wman2IfBsOfdmaDownlinkChannelTable contains OFDMA DCD channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.1.5.2.3 wman2IfBsOfdmaUcdBurstProfileTable

wman2IfBsOfdmaUcdBurstProfileTable contains OFDMA UCD burst profiles for each uplink channel.

15.2.1.1.5.2.4 wman2IfBsOfdmaDcdBurstProfileTable

wman2IfBsOfdmaDcdBurstProfileTable provides one row for each OFDMA DCD burst profile.

15.2.1.1.5.2.5 wman2IfBsOfdmaConfigurationTable

wman2IfBsOfdmaConfigurationTable contains BS configuration objects, specific to OFDMA PHY.

15.2.1.1.5.2.6 wman2IfBsSsOfdmaReqCapabilitiesTable

wman2IfBsSsOfdmaReqCapabilitiesTable 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 wman2IfBsSsOfdmaRspCapabilitiesTable

wman2IfBsSsOfdmaRspCapabilitiesTable 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 wman2IfBsRegisteredSsTable.

15.2.1.1.5.2.8 wman2IfBsOfdmaCapabilitiesTable

wman2IfBsOfdmaCapabilitiesTable 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 wman2IfBsOfdmaCapabilitiesConfigTable

wman2IfBsOfdmaCapabilitiesConfigTable 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 wman2IfSsObjects

15.2.1.2.1 wman2IfSsCps

Figure 25 shows the structure of wman2IfSsCps subtree that contains SS managed objects related to the MAC CPS management entity layer.

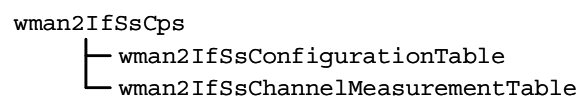


Figure 24—wman2IfSsCps structure

15.2.1.2.1.1 wman2IfSsConfigurationTable

wman2IfSsConfigurationTable 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 wman2IfSsChannelMeasurementTable

wman2IfSsChannelMeasurementTable contains downlink channel measurement information for each SS.

15.2.1.2.2 wman2IfSsPkmObjects

Figure 26 shows the structure of wman2IfSsPkmObjects subtree that contains subscriber station manageable objects related to the privacy management entity.

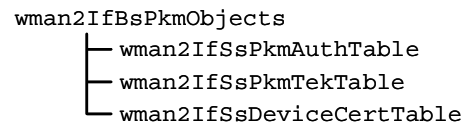


Figure 25—wman2IfSsPkmObjects structure

15.2.1.2.2.1 wman2IfSsPkmAuthTable

wman2IfSsPkmAuthTable 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 wman2IfSsPkmTekTable

wman2IfSsPkmTekTable contains subscriber station runtime parameters for each active security association.

15.2.1.2.2.3 wman2IfSsDeviceCertTable

wman2IfSsDeviceCertTable describes the PKM device certificates for each SS wireless interface.

15.2.1.2.3 wman2IfSsNotification

Figure 27 shows the structure of wman2IfSsNotification subtree that contains SS traps to report fault events and exceptions, such as RSSI threshold crossing.

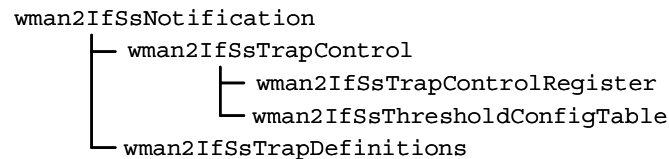


Figure 26—wman2IfSsNotification structure

15.2.1.2.3.1 wman2IfSsTrapControl

15.2.1.2.3.1.1 wman2IfSsTrapControlRegister

wman2IfSsTrapControlRegister is used to enable or disable Subscriber Station traps.

15.2.1.2.3.1.2 wman2IfSsThresholdConfigTable

wman2IfSsThresholdConfigTable contains threshold objects that can be set to detect the threshold crossing events.

15.2.1.2.3.2 wman2IfSsTrapDefinitions

wman2IfSsTrapDefinitions group defines all the traps reported by SS.

15.2.1.2.4 wman2IfSsPhy

Figure 28 shows the structure of wman2IfSsPhy subtree that contains SS managed objects related to the Physical layer.

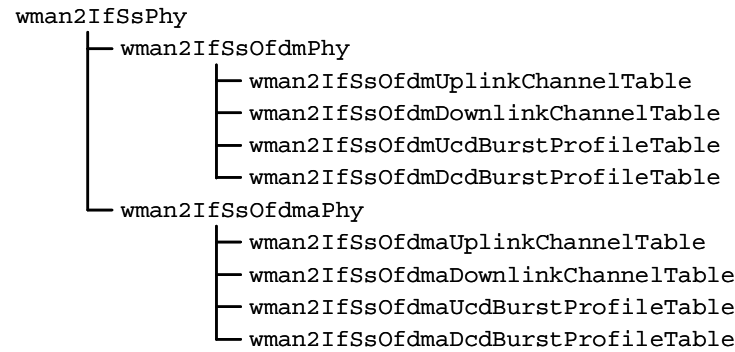


Figure 27—wman2IfSsPhy structure

15.2.1.2.4.1 wman2IfSsOfdmPhy

wman2IfSsOfdmPhy is a group containing objects specific to OFDM PHY.

15.2.1.2.4.1.1 wman2IfSsOfdmUplinkChannelTable

wman2IfSsOfdmUplinkChannelTable contains OFDM UCD channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.2.4.1.2 wman2IfSsOfdmDownlinkChannelTable

wman2IfSsOfdmUplinkChannelTable contains OFDM DCD channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.2.4.1.3 wman2IfSsOfdmUcdBurstProfileTable

wman2IfSsOfdmUcdBurstProfileTable contains OFDM UCD burst profiles for each uplink channel.

15.2.1.2.4.1.4 wman2IfSsOfdmDcdBurstProfileTable

wman2IfSsOfdmDcdBurstProfileTable provides one row for each OFDM DCD burst profile.

15.2.1.2.4.2 wman2IfSsOfdmaPhy

wman2IfSsOfdmaPhy is a group containing objects specific to OFDMA PHY.

15.2.1.2.4.2.1 wman2IfSsOfdmaUplinkChannelTable

wman2IfSsOfdmaUplinkChannelTable contains OFDMA UCD channel attributes, defining the transmission characteristics of uplink channels.

15.2.1.2.4.2.2 wman2IfSsOfdmaDownlinkChannelTable

wman2IfSsOfdmaDownlinkChannelTable contains OFDMA DCD channel attributes, defining the transmission characteristics of downlink channels.

15.2.1.2.4.2.3 wman2IfSsOfdmaUcdBurstProfileTable

wman2IfSsOfdmaUcdBurstProfileTable contains OFDMA UCD burst profiles for each uplink channel.

15.2.1.2.4.2.4 wman2IfSsOfdmaDcdBurstProfileTable

wman2IfSsOfdmaDcdBurstProfileTable provides one row for each OFDMA DCD burst profile.

15.2.1.3 wman2IfCommonObjects

15.2.1.3.1 wman2IfCmnPacketCs

Figure 29 shows the structure of wman2IfCmnPacketCs subtree that contains common managed objects related to the Packet CS management entity layer.

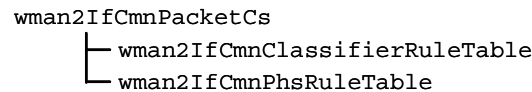


Figure 28—wman2IfCmnPacketCs structure

15.2.1.3.1.1 wman2IfCmnClassifierRuleTable

wman2IfCmnClassifierRuleTable contains runtime classifier rules screening criteria for each service flow.

15.2.1.3.1.2 wman2IfCmnPhsRuleTable

wman2IfCmnPhsRuleTable 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 wman2IfCmnCps

Figure 30 shows the structure of wman2IfCmnCps subtree that contains common managed objects related to the MAC CPS management entity.

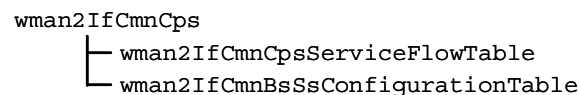


Figure 29—wman2IfCmnCps structure

15.2.1.3.2.1 wman2IfCmnCpsServiceFlowTable

wman2IfCmnCpsServiceFlowTable contains Service Flow managed objects that are common in BS and SS.

15.2.1.3.2.2 wman2IfCmnBsSsConfigurationTable

wman2IfCmnBsSsConfigurationTable 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 wman2IfCmnPkmObjects

Figure 31 shows the structure of wman2IfCmnPkmObjects subtree that contains common PKM objects.

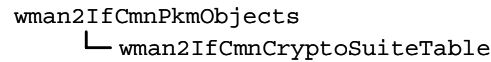


Figure 30—wman2IfCmnPkmObjects structure

15.2.1.3.3.1 wman2IfCmnCryptoSuiteTable

wman2IfCmnCryptoSuiteTable contains supported crypto suites for the particular SS and other crypto parameters such as key lifetimes.

15.2.2 ASN.1 Definitions of 802.16 MIB for SNMP

```

1  15.2.2 ASN.1 Definitions of 802.16 MIB for SNMP
2
3  WMAN2-IF-MIB DEFINITIONS ::= BEGIN
4
5      IMPORTS
6
7          MODULE-IDENTITY,
8          OBJECT-TYPE,
9          NOTIFICATION-TYPE,
10         Unsigned32, Integer32, Counter32,
11         Counter64, transmission
12         FROM SNMPv2-SMI
13         SnmpAdminString
14         FROM SNMP-FRAMEWORK-MIB
15         TEXTUAL-CONVENTION,
16         MacAddress, RowStatus, TruthValue,
17         TimeStamp, DateAndTime
18         FROM SNMPv2-TC
19         InetAddressType, InetAddress
20         FROM INET-ADDRESS-MIB
21         OBJECT-GROUP,
22         MODULE-COMPLIANCE,
23         NOTIFICATION-GROUP
24         FROM SNMPv2-CONF
25         ifIndex
26         FROM IF-MIB;
27
28  wman2IfMib MODULE-IDENTITY
29      LAST-UPDATED      "200608080000Z" -- August 08, 2006
30      ORGANIZATION      "IEEE 802.16"
31      CONTACT-INFO
32          "WG E-mail: stds-802-16@ieee.org
33           WG Chair: Roger B. Marks
34           Postal:   (U.S.) National Institute
35                   of Standards and Technology
36           E-mail:   r.b.marks@ieee.org
37
38           TGf Chair: Phillip Barber
39           Postal:   Huawei Technologies Co., Ltd
40           E-mail:   pbarber@futurewei.com
41
42           Editor:   Joey Chou
43           Postal:   Intel Corporation
44                   5000 W. Chandler Blvd,
45                   Chandler, AZ 85227, USA
46           E-mail:   joey.chou@intel.com"
47
48  DESCRIPTION
49      "This material is from IEEE Std 802.16i
50      Copyright (c) 2006 IEEE.
51      This MIB Module defines managed objects for
52      IEEE 802.16e-2005 based Subscriber Station
53      and Base Station."
54
55  REVISION      "200608080000Z"
56
57  DESCRIPTION
58      "The second revision of WMAN2-IF-MIB module."
59
60
61
62
63
64
65

```

```

1      REVISION          "200605230000Z"
2      DESCRIPTION
3          "The first revision of WMAN2-IF-MIB module that is
4              enhanced to support IEEE 802.16e-2005 standard."
5      ::= { transmission 184 }
6
7
8      wman2IfMibObjects OBJECT IDENTIFIER ::= { wman2IfMib 1 }
9      wman2IfBsObjects OBJECT IDENTIFIER ::= { wman2IfMibObjects 1 }
10     wman2IfSsObjects  OBJECT IDENTIFIER ::= { wman2IfMibObjects 2 }
11     wman2IfCommonObjects OBJECT IDENTIFIER ::= { wman2IfMibObjects 3 }
12
13
14     -- Textual Conventions
15     Wman2IfSfSchedulingType ::= TEXTUAL-CONVENTION
16         STATUS          current
17         DESCRIPTION
18             "The scheduling service provided by a SC for an
19                 upstream service flow. If the parameter is omitted
20                 from an upstream QOS Parameter Set, this object takes
21                 the value of bestEffort (2). This parameter must be
22                 reported as undefined (1) for downstream QOS Parameter
23                 Sets."
24         SYNTAX          INTEGER {undefined(1),
25                                 bestEffort(2),
26                                 nonRealTimePollingService(3),
27                                 realTimePollingService(4),
28                                 reserved(5),
29                                 unsolicitedGrantService(6)}
30
31
32     Wman2IfPhsRuleVerify ::= TEXTUAL-CONVENTION
33         STATUS          current
34         DESCRIPTION
35             "The value of this field indicates to the sending entity
36                 whether or not the packet header contents are to be
37                 verified prior to performing suppression. If PHSV is
38                 enabled, the sender shall compare the bytes in the packet
39                 header with the bytes in the PHSF that are to be
40                 suppressed as indicated by the PHSM."
41         REFERENCE
42             "Subclause 11.13.19.3.7.5 in IEEE Std 802.16-2004"
43         SYNTAX          INTEGER {phsVerifyEnable(0),
44                                 phsVerifyDisable(1)}
45
46
47     Wman2IfClassifierBitMap ::= 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                 encoded as the octet string '030000'H."
56         REFERENCE
57             "Subclause 11.13.19.3.4 in IEEE Std 802.16-2004"
58         SYNTAX          BITS {priority(0),
59

```

```

1          ipTos(1),
2          ipProtocol(2),
3          ipMaskedSrcAddr(3),
4          ipMaskedDestAddr(4),
5          srcPort(5),
6          destPort(6),
7          destMacAddr(7),
8          srcMacAddr(8),
9          ethernetProtocol(9),
10         userPriority(10),
11         vlanId(11),
12         ipv6FlowLabel(12)}
13
14
15
16 Wman2IfSfState ::= TEXTUAL-CONVENTION
17     STATUS          current
18     DESCRIPTION
19         "Wman2IfSfState defines the state of a service flow."
20     SYNTAX          INTEGER {authorized(1),
21                             admitted(2),
22                             active(3)}
23
24
25
26 Wman2IfServClassName ::= TEXTUAL-CONVENTION
27     STATUS          current
28     DESCRIPTION
29         "Wman2IfServClassName defines the type of service
30         class name."
31     SYNTAX          OCTET STRING (SIZE(2..128))
32
33
34
35 Wman2IfCsSpecification ::= TEXTUAL-CONVENTION
36     STATUS          current
37     DESCRIPTION
38         "Wman2IfCsSpecification defines the types of convergence
39         sublayer."
40     REFERENCE
41         "Subclause 11.13.19.1 in IEEE Std 802.16e-2005"
42     SYNTAX          INTEGER {reserved(0),
43                             packetIPv4(1),
44                             packetIPv6(2),
45                             packet802dot3Ethernet(3),
46                             packet802dot1QVlan(4),
47                             packetIPv4Over802dot3(5),
48                             packetIPv6Over802dot3(6),
49                             packetIPv4Over802dot1Q(7),
50                             packetIPv6Over802dot1Q(8),
51                             atm(9),
52                             packet802dot3EthernetRohcHc(10),
53                             packet802dot3EthernetEcrtphc(11),
54                             packetIp2RohcHc(12),
55                             packetIp2Ecrtphc(13)}
56
57
58
59
60
61 Wman2IfMacVersion ::= TEXTUAL-CONVENTION
62     STATUS          current
63     DESCRIPTION
64         "Version number of IEEE 802.16."
65

```

```

1          SYNTAX          INTEGER {ieee802Dot16Of2001(1),
2                                ieee802Dot16cOf2002(2),
3                                ieee802Dot16aOf2003(3),
4                                ieee802Dot16Of2004(4),
5                                ieee802Dot16e(5),
6                                tbd(6)}
7
8
9
10         Wman2IfCidType ::= TEXTUAL-CONVENTION
11             STATUS          current
12             DESCRIPTION
13                 "Type of CID."
14             SYNTAX          INTEGER (0 .. 65535)
15
16
17         Wman2IfDataEncryptAlgId ::= TEXTUAL-CONVENTION
18             STATUS          current
19             DESCRIPTION
20                 "Data encryption algorithm identifiers."
21             REFERENCE
22                 "Table 375 in IEEE Std 802.16-2004"
23             SYNTAX          INTEGER {none(0),
24                                   des56BitCbcMode(1),
25                                   aesCcmMode(2)}
26
27
28
29         Wman2IfDataAuthAlgId ::= TEXTUAL-CONVENTION
30             STATUS          current
31             DESCRIPTION
32                 "Data authentication algorithm identifiers."
33             REFERENCE
34                 "Table 376 in IEEE Std 802.16-2004"
35             SYNTAX          INTEGER {noDataAuthentication(0),
36                                   reserved(1)}
37
38
39
40         Wman2IfTekEncryptAlgId ::= TEXTUAL-CONVENTION
41             STATUS          current
42             DESCRIPTION
43                 "TEK encryption algorithm identifiers."
44             REFERENCE
45                 "Table 377 in IEEE Std 802.16-2004"
46             SYNTAX          INTEGER {tripleDes128BitKey(1),
47                                   rsa1024BitKey(2),
48                                   aes128BitKey(3)}
49
50
51
52         Wman2IfChannelNumber ::= TEXTUAL-CONVENTION
53             STATUS          current
54             DESCRIPTION
55                 "Physical channel number"
56             SYNTAX          INTEGER (0 .. 199)
57
58
59         Wman2IfOfdmFecCodeType ::= TEXTUAL-CONVENTION
60             STATUS          current
61             DESCRIPTION
62                 "FEC code type and modulation type"
63             REFERENCE
64                 "Table 356 and Table 362 in IEEE Std 802.16-2004"
65

```

```

1          SYNTAX          INTEGER {bpskCc1Over2(0),
2                                qpskRsCcCc1Over2(1),
3                                qpskRsCcCc3Over4(2),
4                                sixteenQamRsCcCc1Over2(3),
5                                sixteenQamRsCcCc3Over4(4),
6                                sixtyFourQamRsCcCc2Over3(5),
7                                sixtyFourQamRsCcCc3Over4(6),
8                                qpskBtc1Over2(7),
9                                qpskBtc3Over4(8),
10                               sixteenQamBtc3Over4(9),
11                               sixteenQamBtc4Over5(10),
12                               sixtyFourQamBtc2Over3(11),
13                               sixtyFourQamBtc5Over6(12),
14                               qpskCtc1Over2(13),
15                               qpskCtc2Over3(14),
16                               qpskCtc3Over4(15),
17                               sixteenQamCtc1Over2(16),
18                               sixteenQamCtc3Over4(17),
19                               sixtyFourQamCtc2Over3(18),
20                               sixtyFourQamCtc3Over4(19)}
21
22
23
24
25
26 Wman2IfOfdmaFecCodeType ::= TEXTUAL-CONVENTION
27     STATUS          current
28     DESCRIPTION
29         "FEC code type and modulation type"
30     REFERENCE
31         "Table 356 and Table 362 in IEEE Std 802.16-2004"
32     SYNTAX          INTEGER {qpskCc1Over2(0),
33                               qpskCc3Over4(1),
34                               sixteenQamCc1Over2(2),
35                               sixteenQamCc3Over4(3),
36                               sixtyFourQamCc2Over3(4),
37                               sixtyFourQamCc3Over4(5),
38                               qpskBtc1Over2(6),
39                               qpskBtc2Over3(7),
40                               sixteenQamBtc3Over5(8),
41                               sixteenQamBtc4Over5(9),
42                               sixtyFourQamBtc5Over8(10),
43                               sixtyFourQamBtc4Over5(11),
44                               qpskCtc1Over2(12),
45                               qpskCtc2Over3(13),
46                               qpskCtc3Over4(14),
47                               sixteenQamCtc1Over2(15),
48                               sixteenQamCtc3Over4(16),
49                               sixtyFourQamCtc2Over3(17),
50                               sixtyFourQamCtc3Over4(18),
51                               sixtyFourQamCtc5Over6(19),
52                               qpskZtCc1Over2(20),
53                               qpskZtCc3Over4(21),
54                               sixteenQamZtCc1Over2(22),
55                               sixteenQamZtCc3Over4(23),
56                               sixtyFourQamZtCc2Over3(24),
57                               sixtyFourQamZtCc3Over4(25)}
58
59
60
61
62
63
64
65

```

```

1  -- Textual convention for capabilities encodings
2  Wman2IfNumOfUplinkCid ::= TEXTUAL-CONVENTION
3      STATUS          current
4      DESCRIPTION
5          "The object of this type shows the number of Uplink CIDs
6          the SS can support."
7      REFERENCE
8          "Subclause 11.7.4 in IEEE Std 802.16-2004"
9      SYNTAX          INTEGER (2..65535)
10
11
12
13  Wman2IfArqSupportType ::= TEXTUAL-CONVENTION
14      STATUS          current
15      DESCRIPTION
16          "The object of this type indicates whether the SS support
17          ARQ."
18      REFERENCE
19          "Subclause 11.7.8.1 in IEEE Std 802.16-2004"
20      SYNTAX          INTEGER {arqNotSupported(0),
21                              arqSupported(1)}
22
23
24
25  Wman2IfMaxDsxFLOWType ::= TEXTUAL-CONVENTION
26      STATUS          current
27      DESCRIPTION
28          "The object of this type specifies the maximum number of
29          concurrent DSA, DSC, or DSD transactions that may be
30          outstanding."
31      REFERENCE
32          "Subclause 11.7.8.2 in IEEE Std 802.16-2004"
33      SYNTAX          INTEGER (0..255)
34
35
36
37  Wman2IfMacCrcSupport ::= TEXTUAL-CONVENTION
38      STATUS          current
39      DESCRIPTION
40          "The object of this type indicates whether or not the SS
41          supports MAC level CRC."
42      REFERENCE
43          "Subclause 11.7.8.3 in IEEE Std 802.16-2004"
44      SYNTAX          INTEGER {noMacCrcSupport(0),
45                              macCrcSupport(1)}
46
47
48
49  Wman2IfMaxMcaFlowType ::= TEXTUAL-CONVENTION
50      STATUS          current
51      DESCRIPTION
52          "The object of this type specifies the maximum number of
53          concurrent MCA transactions that may be outstanding."
54      REFERENCE
55          "Subclause 11.7.8.4 in IEEE Std 802.16-2004"
56      SYNTAX          INTEGER (0..255)
57
58
59
60  Wman2IfMaxMcpGroupCid ::= TEXTUAL-CONVENTION
61      STATUS          current
62      DESCRIPTION
63          "The object of this type indicates the maximum number of
64          simultaneous Multicast Polling Groups the SS is
65

```

```

1         capable of belonging to."
2     REFERENCE
3         "Subclause 11.7.8.5 in IEEE Std 802.16-2004"
4     SYNTAX         INTEGER (0..255)
5
6
7     Wman2IfMaxPkmFlowType ::= TEXTUAL-CONVENTION
8         STATUS         current
9         DESCRIPTION
10            "The object of this type specifies the maximum number of
11             concurrent PKM transactions that may be outstanding."
12        REFERENCE
13            "Subclause 11.7.8.6 in IEEE Std 802.16-2004"
14        SYNTAX         INTEGER (0..255)
15
16
17
18     Wman2IfAuthPolicyType ::= TEXTUAL-CONVENTION
19         STATUS         current
20         DESCRIPTION
21            "The object of this type specifies authorization policy
22             that both SS and BS need to negotiate and synchronize.
23             A bit value of 0 = not supported, 1 = supported. If this
24             field is omitted, then both SS and BS shall use the IEEE
25             802.16 security, constituting X.509 digital certificates
26             and the RSA public key encryption algorithm, as
27             authorization policy."
28        REFERENCE
29            "Subclause 11.7.8.7 in IEEE Std 802.16-2004"
30        SYNTAX         BITS {ieee802Dot16PrivacySupported(0),
31                             reserved1(1),
32                             reserved2(2),
33                             reserved3(3),
34                             reserved4(4),
35                             reserved5(5),
36                             reserved6(6),
37                             reserved7(7)}
38
39
40
41
42
43     Wman2IfMaxNumOfSaType ::= TEXTUAL-CONVENTION
44         STATUS         current
45         DESCRIPTION
46            "This field specifies maximum number of supported
47             security association of the SS."
48        REFERENCE
49            "Subclause 11.7.8.8 in IEEE Std 802.16-2004"
50        SYNTAX         INTEGER (0..255)
51
52
53
54     Wman2IfIpVersionType ::= TEXTUAL-CONVENTION
55         STATUS         current
56         DESCRIPTION
57            "The object of this type indicates the version of IP used
58             on the Secondary Management Connection. The value should
59             be undefined if the 2nd management CID doesn't exist."
60        REFERENCE
61            "Subclause 11.7.4 in IEEE Std 802.16-2004"
62        SYNTAX         INTEGER {undefined(0),
63                             ipv4(1),
64

```

```

1          ipv6(2) }
2
3
4 Wman2IfMacCsBitMap ::= TEXTUAL-CONVENTION
5     STATUS          current
6     DESCRIPTION
7         "The object of this type indicates the set of MAC
8         convergence sublayer support. When a bit is set, it
9         indicates the corresponding CS feature is supported."
10
11     REFERENCE
12         "Subclause 11.7.7.1 in IEEE Std 802.16e-2005"
13
14     SYNTAX          BITS {atm(0),
15                          packetIpv4(1),
16                          packetIpv6(2),
17                          packet802Dot3(3),
18                          packet802Dot1Q(4),
19                          packetIpv4Over802Dot3(5),
20                          packetIpv6Over802Dot3(6),
21                          packetIpv4Over802Dot1Q(7),
22                          packetIpv6Over802Dot1Q(8),
23                          packet802dot3EthernetRohcHc(9),
24                          packet802dot3EthernetEcrtpHc(10),
25                          packetIpv4Orv6RohcHc(11),
26                          packetIpv4Orv6EcrtpHc(12) }
27
28
29
30 Wman2IfMaxClassifiers ::= TEXTUAL-CONVENTION
31     STATUS          current
32     DESCRIPTION
33         "The object of this type indicates the maximum number of
34         admitted Classifiers that the SS is allowed to have."
35
36     REFERENCE
37         "Subclause 11.7.7.2 in IEEE Std 802.16-2004"
38
39     SYNTAX          INTEGER (0..65535)
40
41
42 Wman2IfPhsSupportType ::= TEXTUAL-CONVENTION
43     STATUS          current
44     DESCRIPTION
45         "The object of this type indicates the level
46         of PHS support."
47
48     REFERENCE
49         "Subclause 11.7.7.3 in IEEE Std 802.16-2004"
50
51     SYNTAX          INTEGER {noPhsSupport(0),
52                          atmPhsSupport(1),
53                          packetPhsSupport(2) }
54
55
56 Wman2IfBwAllocSupport ::= TEXTUAL-CONVENTION
57     STATUS          current
58     DESCRIPTION
59         "This field indicates properties of the SS that the BS
60         needs to know for bandwidth allocation purposes. When
61         a bit is set, it indicates the corresponding feature
62         is supported. All unspecified and reserved bits should
63         be set to zero."
64
65     REFERENCE
66         "Subclause 11.8.1 in IEEE Std 802.16-2004"

```



```

1          SYNTAX          BITS {reserved(0),
2                                halfDuplexFdd(1),
3                                fullDuplexFdd(2)}
4
5
6 Wman2IfPduConstruction ::= TEXTUAL-CONVENTION
7     STATUS          current
8     DESCRIPTION
9         "Specifies capabilities for construction and transmission
10        of MAC PDUs. When piggybackedRequests bit is set, it
11        indicates that the piggybacked requests are supported. The
12        fsnValuesSize bit is coded as follows:
13        0 - only 3-bit FSN values are supported
14        1 - only 11-bit FSN values are supported
15        All unspecified and reserved bits should be set to zero."
16     REFERENCE
17         "Subclause 11.8.2 in IEEE Std 802.16-2004"
18     SYNTAX          BITS {piggybackedRequests(0),
19                            fsnValuesSize(1)}
20
21
22
23
24 Wman2IfSsTransitionGap ::= TEXTUAL-CONVENTION
25     STATUS          current
26     DESCRIPTION
27         "This field indicates the transition speed SSTTG and SSRTG
28        for TDD and H-FDD SSs. Allowed values are:
29        OFDM mode: TDD and H-FDD 0..100
30        Other modes: TDD: 0..50; H-FDD: 0..100"
31     REFERENCE
32         "Subclause 11.8.3.1 in IEEE Std 802.16-2004"
33     SYNTAX          INTEGER (0..100)
34
35
36
37 Wman2IfMaxTxPowerType ::= TEXTUAL-CONVENTION
38     STATUS          current
39     DESCRIPTION
40         "This type is used to define maximum available power for
41        BPSK, QPSK, 16-QAM and 64-QAM constellations. The maximum
42        power parameters are reported in dBm and quantized in 0.5
43        dBm steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
44        (encoded 0xFF). Values outside this range shall be
45        assigned the closest extreme. SSs that do not support
46        QAM64 shall report the value of 0x00 in the maximum QAM64
47        power field."
48     REFERENCE
49         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
50     SYNTAX          INTEGER (0..255)
51
52
53
54
55 Wman2IfOfdmFftSizes ::= TEXTUAL-CONVENTION
56     STATUS          current
57     DESCRIPTION
58         "This field indicates the FFT sizes supported by the SS/MS.
59        It is used for describing OFDM or OFDMA capabilities,
60        depending on context. For each FFT size, a bit value of 0
61        indicates 'not supported' while 1 indicates 'supported'."
62     REFERENCE
63         "Subclause 11.8.3.7.1 in IEEE 802.16e-2005"
64
65

```

```

1          SYNTAX          BITS {fft256(0),
2                                fft2048(1),
3                                fft128(2),
4                                fft512(3),
5                                fft1024(4)}
6
7
8  Wman2IfOfdmSsDeModType ::= TEXTUAL-CONVENTION
9      STATUS              current
10     DESCRIPTION
11         "This field indicates the different demodulator options
12         supported by a WirelessMAN-OFDM PHY SS for downlink. This
13         field is not used for other PHY specifications. A bit
14         value of 0 indicates 'not supported' while 1 indicates
15         'supported'."
16     REFERENCE
17         "Subclause 11.8.3.6.2 in IEEE Std 802.16-2004"
18     SYNTAX          BITS {qam64(0),
19                           btc(1),
20                           ctc(2),
21                           stc(3),
22                           aac(4)}
23
24
25  Wman2IfOfdmSsModType ::= TEXTUAL-CONVENTION
26     STATUS              current
27     DESCRIPTION
28         "This field indicates the different modulator options
29         supported by a WirelessMAN-OFDM PHY SS for uplink. This
30         field is not used for other PHY specifications. A bit
31         value of 0 indicates 'not supported' while 1 indicates
32         'supported'."
33     REFERENCE
34         "Subclause 11.8.3.6.3 in IEEE Std 802.16-2004"
35     SYNTAX          BITS {qam64(0),
36                           btc(1),
37                           ctc(2),
38                           subchannellization(3),
39                           focusedCtBwReq(4)}
40
41
42  Wman2IfOfdmFocusedCt ::= TEXTUAL-CONVENTION
43     STATUS              current
44     DESCRIPTION
45         "This field indicates whether the SS supports Focused
46         Contention (see 8.3.7.3.3). A bit value of 0 indicates
47         'not supported' while 1 indicates 'supported'."
48     REFERENCE
49         "Subclause 11.8.3.6.4 in IEEE Std 802.16-2004"
50     SYNTAX          BITS {focusedCtSupport(0)}
51
52
53  Wman2IfOfdmTcSublayer ::= TEXTUAL-CONVENTION
54     STATUS              current
55     DESCRIPTION
56         "This field indicates whether or not the SS supports the
57         TC sublayer (see 8.3.4). A bit value of 0 indicates
58         'not supported' while 1 indicates 'supported'."
59

```

```

1      REFERENCE
2          "Subclause 11.8.3.6.5 in IEEE Std 802.16-2004"
3      SYNTAX      BITS {tcSublayerSupport(0)}
4
5
6      Wman2IfBsIdType ::= TEXTUAL-CONVENTION
7          STATUS      current
8          DESCRIPTION
9              "Defines the encoding of BSID. The BSID is a 6 byte number
10             and follows the encoding rules of MacAddress textual
11             convention, i.e. as if it were transmitted
12             least-significant bit first. The value should be displayed
13             with 2 parts clearly separated by a colon e.g:
14             001DFF:00003A. The most significant part is representing
15             the Operator ID. "
16      SYNTAX      OCTET STRING (SIZE(6))
17
18
19
20      Wman2IfIpv6FlowLabel ::= TEXTUAL-CONVENTION
21          STATUS      current
22          DESCRIPTION
23              "The value of this field specifies the matching values for
24             the IPv6 Flow label field. As the flow label field has a
25             length of 20 bits, the first 4 bits of the most
26             significant byte shall be set to 0x0 and disregarded."
27      SYNTAX      OCTET STRING (SIZE(3))
28
29
30
31      Wman2IfOfdmaMsDeModType ::= TEXTUAL-CONVENTION
32          STATUS      current
33          DESCRIPTION
34              "This field indicates the different demodulator options
35             supported by a WirelessMAN-OFDMA PHY SS for downlink.
36             A bit value of 0 indicates 'not supported' while 1
37             indicates 'supported'."
38      REFERENCE
39          "Subclause 11.8.3.7.2 in IEEE 802.16e"
40      SYNTAX      BITS {qam64(0),
41                       btc(1),
42                       ctc(2),
43                       stc(3),
44                       aasDiversityMapScan(4),
45                       harqChase(5),
46                       harqCtcIr(6),
47                       reserved(7),
48                       harqCcIr(8),
49                       ldpc(9)}
50
51
52
53
54
55      Wman2IfOfdmaMsModType ::= TEXTUAL-CONVENTION
56          STATUS      current
57          DESCRIPTION
58              "This field indicates the different modulator options
59             supported by a WirelessMAN-OFDMA PHY SS for uplink. A bit
60             value of 0 indicates 'not supported' while 1 indicates
61             'supported'."
62      REFERENCE
63          "Subclause 11.8.3.7.3 in IEEE 802.16e"
64
65

```

```

1          SYNTAX          BITS {qam64(0),
2                                btc(1),
3                                ctc(2),
4                                stc(3),
5                                harqChase(4),
6                                ctcIr(5),
7                                ccIr(6),
8                                ldpc(7)}
9
10
11
12 Wman2IfOfdmaPermutation ::= TEXTUAL-CONVENTION
13     STATUS          current
14     DESCRIPTION
15         "This field indicates the OFDMA SS Permutation support
16         A bit value of 0 indicates 'not supported' while 1
17         indicates 'supported'."
18     REFERENCE
19         "Subclause 11.8.3.7.5 in IEEE 802.16e"
20     SYNTAX          BITS {optionalPuscSupport(0),
21                            optionalFuscSupport(1),
22                            amcOneBySixSupport(2),
23                            amcTwoByThreeSupport(3),
24                            amcThreeByTwoSupport(4),
25                            amcSupportWithHarqMap(5),
26                            tusclSupport(6),
27                            tusc2(7)}
28
29
30
31
32 Wman2IfOfdmaMobility ::= TEXTUAL-CONVENTION
33     STATUS          current
34     DESCRIPTION
35         "This field indicates whether or not the MS supports
36         mobility hand-over, Sleepmode, and Idle-mode. A bit
37         value of 0 indicates 'not supported' while 1 indicates
38         it is supported."
39     REFERENCE
40         "Subclause 11.8.3.7.5 in IEEE 802.16e"
41     SYNTAX          BITS {handoverSupport(0),
42                            sleepModeSupport(1),
43                            idleModeSupport(2)}
44
45
46
47
48 Wman2IfPsClassId ::= TEXTUAL-CONVENTION
49     STATUS          current
50     DESCRIPTION
51         "Wman2IfPsClassId indicates the index to Power Saving
52         Classes. The ID shall be unique within the group of Power
53         Saving Classes associated with the MS. This ID may be
54         used in further MOB_SLP-REQ/RSP messages for activation /
55         deactivation of Power Saving Class."
56     REFERENCE
57         "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
58     SYNTAX          INTEGER (0..63)
59
60
61
62 Wman2PsClassType ::= TEXTUAL-CONVENTION
63     STATUS          current
64     DESCRIPTION
65

```

```

1         "The types of power saving classes."
2     REFERENCE
3         "Table 374a in IEEE Std 802.16e-2005"
4     SYNTAX      INTEGER {powerSavingClassTypeI(1),
5                  powerSavingClassTypeII(2),
6                  powerSavingClassTypeIII(3)}
7
8
9
10    Wman2PsClassCidDirection ::= TEXTUAL-CONVENTION
11        STATUS      current
12        DESCRIPTION
13            "The direction of power saving class's CIDs.
14             0b00 = Unspecified. Each CID has its own direction
15             assign in its connection creation. Can be
16             DL, UL, or both (in the case of management
17             connections).
18             0b01 = Downlink direction only.
19             0b10 = Uplink direction only."
20        REFERENCE
21            "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
22        SYNTAX      INTEGER {unspecified(0),
23                             downlink(1),
24                             uplink(2)}
25
26
27
28
29    Wman2IfPowerSavingMode ::= TEXTUAL-CONVENTION
30        STATUS      current
31        DESCRIPTION
32            "Power saving class mode active or not active."
33        REFERENCE
34            "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
35        SYNTAX      INTEGER {psNotActive(0),
36                             psActive(1)}
37
38
39
40    --
41    -- BS object group - containing tables and objects to be implemented in
42    -- the Base station
43    --
44    -- wman2IfBsPacketCs contain the Base Station Packet Convergence
45    -- Sublayer objects
46    --
47
48    wman2IfBsPacketCs OBJECT IDENTIFIER ::= { wman2IfBsObjects 1 }
49
50    wman2IfBsProvisionedSfTable OBJECT-TYPE
51        SYNTAX      SEQUENCE OF Wman2IfBsProvisionedSfEntry
52        MAX-ACCESS  not-accessible
53        STATUS      current
54        DESCRIPTION
55            "This table contains service flow profiles provisioned by
56            NMS. The service flow should be created with SS(s)
57            following instruction given by wman2IfBsSfState object.
58            1. The QoS parameters of the service flow are provisioned
59            in wman2IfBsServiceClassTable and referenced by
60            wman2IfBsServiceClassIndex.
61            2. The classifier rules of the service flow are provisioned
62            in wman2IfBsClassifierRuleTable, where they refer to SF
63
64
65

```

```

1           via wman2IfBsSfId.
2
3           The MAC addresses of SSs the service flow is created with
4           are provisioned in wman2IfBsSsProvisionedForSfTable, where
5           they refer to SF via wman2IfBsSfId."
6
7       REFERENCE
8           "Subclause 6.3.13 and 6.3.14 in IEEE Std 802.16-2004"
9       ::= { wman2IfBsPacketCs 1 }
10
11
12   wman2IfBsProvisionedSfEntry OBJECT-TYPE
13       SYNTAX      Wman2IfBsProvisionedSfEntry
14       MAX-ACCESS  not-accessible
15       STATUS      current
16       DESCRIPTION
17           "This table provides one row for each service flow
18           provisioned by NMS. The table is indexed by ifIndex and
19           wman2IfBsSfId. ifIndex is associated with the BS sector."
20       INDEX { ifIndex, wman2IfBsSfId }
21       ::= { wman2IfBsProvisionedSfTable 1 }
22
23
24
25   Wman2IfBsProvisionedSfEntry ::= SEQUENCE {
26       wman2IfBsSfId                Unsigned32,
27       wman2IfBsSfDirection          INTEGER,
28       wman2IfBsServiceClassIndex    INTEGER,
29       wman2IfBsSfState              Wman2IfSfState,
30       wman2IfBsSfProvisionedTime    TimeStamp,
31       wman2IfBsSfCsSpecification    Wman2IfCsSpecification,
32       wman2IfBsProvisionedSfRowStatus RowStatus}
33
34
35
36   wman2IfBsSfId OBJECT-TYPE
37       SYNTAX      Unsigned32 (1 .. 4294967295)
38       MAX-ACCESS  not-accessible
39       STATUS      current
40       DESCRIPTION
41           "A 32 bit quantity that uniquely identifies a service flow
42           to both the subscriber station and base station (BS)."
43       ::= { wman2IfBsProvisionedSfEntry 1 }
44
45
46
47   wman2IfBsSfDirection OBJECT-TYPE
48       SYNTAX      INTEGER {downstream(1),
49                           upstream(2)}
50       MAX-ACCESS  read-create
51       STATUS      current
52       DESCRIPTION
53           "An attribute indicating the service flow is downstream or
54           upstream."
55       ::= { wman2IfBsProvisionedSfEntry 2 }
56
57
58
59   wman2IfBsServiceClassIndex OBJECT-TYPE
60       SYNTAX      INTEGER (1..65535)
61       MAX-ACCESS  read-create
62       STATUS      current
63       DESCRIPTION
64           "The index in wman2IfBsServiceClassTable describing the
65

```

```

1      service class or QoS parameters for such service flow.
2      If no associated entry in wman2IfBsServiceClassTable
3      exists, this object returns a value of zero."
4      ::= { wman2IfBsProvisionedSfEntry 3 }

```

wman2IfBsSfState OBJECT-TYPE

```

8      SYNTAX      Wman2IfSfState
9
10     MAX-ACCESS   read-create
11     STATUS      current
12     DESCRIPTION
13         "wman2IfBsSfState determines the requested state of a service
14         flow.
15         - authorized state: A service flow is provisioned but
16         not resource is reserved yet
17         - admitted state: service flow has resources reserved.
18         - active state: has resources committed by the BS (e.g., is
19         actively sending maps containing unsolicited grants for a
20         UGS-based service flow),"
21
22     REFERENCE
23         "Subclause 6.3.14.6, in IEEE Std 802.16-2004"
24     ::= { wman2IfBsProvisionedSfEntry 4 }

```

wman2IfBsSfProvisionedTime OBJECT-TYPE

```

28     SYNTAX      TimeStamp
29
30     MAX-ACCESS   read-create
31     STATUS      current
32     DESCRIPTION
33         "Indicates the date and time when the service flow is
34         provisioned."
35     ::= { wman2IfBsProvisionedSfEntry 5 }

```

wman2IfBsSfCsSpecification OBJECT-TYPE

```

39     SYNTAX      Wman2IfCsSpecification
40
41     MAX-ACCESS   read-create
42     STATUS      current
43     DESCRIPTION
44         "This parameter specifies the convergence sublayer
45         encapsulation mode."
46
47     REFERENCE
48         "Subclause 11.13.19.1 in IEEE Std 802.16-2004"
49     ::= { wman2IfBsProvisionedSfEntry 6 }

```

wman2IfBsProvisionedSfRowStatus OBJECT-TYPE

```

52     SYNTAX      RowStatus
53
54     MAX-ACCESS   read-create
55     STATUS      current
56     DESCRIPTION
57         "This object is used to create a new row or modify or
58         delete an existing row in this table.
59
60
61         If the implementator of this MIB has choosen not
62         to implement 'dynamic assignment' of profiles, this
63         object is not useful and should return noSuchName
64         upon SNMP request."
65

```

```

1      ::= { wman2IfBsProvisionedSfEntry 7 }
2
3
4  wman2IfBsSsProvisionedForSfTable OBJECT-TYPE
5      SYNTAX      SEQUENCE OF Wman2IfBsSsProvisionedForSfEntry
6      MAX-ACCESS  not-accessible
7      STATUS      current
8      DESCRIPTION
9          "This table maps the MAC addresses of SSs to the service
10         flows provisioned in wman2IfBsProvisionedSfTable."
11
12     REFERENCE
13         "Subclause 6.3.14 in IEEE Std 802.16-2004"
14     ::= { wman2IfBsPacketCs 2 }
15
16
17  wman2IfBsSsProvisionedForSfEntry OBJECT-TYPE
18      SYNTAX      Wman2IfBsSsProvisionedForSfEntry
19      MAX-ACCESS  not-accessible
20      STATUS      current
21      DESCRIPTION
22          "This table is indexed by wman2IfBsSsProvMacAddress and
23         wman2IfBsProvSfId."
24
25     INDEX { wman2IfBsSsProvMacAddress, wman2IfBsProvSfId }
26     ::= { wman2IfBsSsProvisionedForSfTable 1 }
27
28
29  Wman2IfBsSsProvisionedForSfEntry ::= SEQUENCE {
30      wman2IfBsSsProvMacAddress      MacAddress,
31      wman2IfBsProvSfId              Unsigned32,
32      wman2IfBsSsProvisionedForSfRowStatus  RowStatus}
33
34
35  wman2IfBsSsProvMacAddress OBJECT-TYPE
36      SYNTAX      MacAddress
37      MAX-ACCESS  not-accessible
38      STATUS      current
39      DESCRIPTION
40          "The MAC address of the SS, the service flow is created
41         with."
42
43     ::= { wman2IfBsSsProvisionedForSfEntry 1 }
44
45
46  wman2IfBsProvSfId OBJECT-TYPE
47      SYNTAX      Unsigned32 (1 .. 4294967295)
48      MAX-ACCESS  not-accessible
49      STATUS      current
50      DESCRIPTION
51          "A 32 bit quantity that uniquely identifies a service flow.
52         The value of this object can be used by BS to index the
53         wman2BsProvisionedSfTable."
54
55     ::= { wman2IfBsSsProvisionedForSfEntry 2 }
56
57
58  wman2IfBsSsProvisionedForSfRowStatus OBJECT-TYPE
59      SYNTAX      RowStatus
60      MAX-ACCESS  read-create
61      STATUS      current
62      DESCRIPTION
63          "This object is used to ensure that the write, create,
64         delete operation to multiple columns is guaranteed to
65

```



```

1         be treated as atomic operation by agent."
2     ::= { wman2IfBsSsProvisionedForSfEntry 3 }
3
4
5 wman2IfBsServiceClassTable OBJECT-TYPE
6     SYNTAX      SEQUENCE OF Wman2IfBsServiceClassEntry
7     MAX-ACCESS  not-accessible
8     STATUS      current
9     DESCRIPTION
10
11         "This table is provisioned and is indexed by
12         wman2IfBsQoSProfileIndex. Each entry of the table contains
13         corresponding service flow characteristic attributes
14         (e.g. QoS parameter set). The value of
15         wman2IfBsQoSProfileIndex is obtained from
16         wman2IfBsServiceClassIndex in wman2IfBsProvisionedSfTable"
17
18     REFERENCE
19         "Subclause 6.3.14.4 in IEEE Std 802.16-2004"
20     ::= { wman2IfBsPacketCs 3 }
21
22
23 wman2IfBsServiceClassEntry OBJECT-TYPE
24     SYNTAX      Wman2IfBsServiceClassEntry
25     MAX-ACCESS  not-accessible
26     STATUS      current
27     DESCRIPTION
28
29         "This table provides one row for each service class"
30     INDEX { ifIndex, wman2IfBsQoSProfileIndex }
31     ::= { wman2IfBsServiceClassTable 1 }
32
33
34 Wman2IfBsServiceClassEntry ::= SEQUENCE {
35     wman2IfBsQoSProfileIndex          INTEGER,
36     wman2IfBsQoSServiceClassName     Wman2IfServClassName,
37     wman2IfBsQoSSTrafficPriority      INTEGER,
38     wman2IfBsQoSMaxSustainedRate     Unsigned32,
39     wman2IfBsQoSMaxTrafficBurst      Unsigned32,
40     wman2IfBsQoSMinReservedRate      Unsigned32,
41     wman2IfBsQOSToleratedJitter      Unsigned32,
42     wman2IfBsQoSMaxLatency           Unsigned32,
43     wman2IfBsQoSFixedVsVariableSduInd INTEGER,
44     wman2IfBsQOSSduSize              Unsigned32,
45     wman2IfBsQoSScSchedulingType     Wman2IfSfSchedulingType,
46     wman2IfBsQoSScArqEnable          TruthValue,
47     wman2IfBsQoSScArqWindowSize      INTEGER,
48     wman2IfBsQoSScArqBlockLifetime   INTEGER,
49     wman2IfBsQoSScArqSyncLossTimeout INTEGER,
50     wman2IfBsQoSScArqDeliverInOrder  TruthValue,
51     wman2IfBsQoSScArqRxPurgeTimeout  INTEGER,
52     wman2IfBsQoSScArqBlockSize       INTEGER,
53     wman2IfBsQoSMinRsvdTolerableRate Unsigned32,
54     wman2IfBsQoSReqTxPolicy          BITS,
55     wman2IfBsQOSServiceClassRowStatus RowStatus }
56
57
58
59
60
61 wman2IfBsQoSProfileIndex OBJECT-TYPE
62     SYNTAX      INTEGER (1 .. 65535)
63     MAX-ACCESS  not-accessible
64     STATUS      current
65

```

```

1      DESCRIPTION
2          "The index value which uniquely identifies an entry
3          in the wman2IfBsServiceClassTable"
4      ::= { wman2IfBsServiceClassEntry 1 }
5
6
7      wman2IfBsQoSServiceClassName OBJECT-TYPE
8          SYNTAX      Wman2IfServClassName
9          MAX-ACCESS   read-create
10         STATUS      current
11         DESCRIPTION
12             "Refers to the Service Class Name"
13         REFERENCE
14             "Subclause 11.13.3 in IEEE Std 802.16-2004"
15         ::= { wman2IfBsServiceClassEntry 2 }
16
17
18
19      wman2IfBsQoSSTrafficPriority OBJECT-TYPE
20          SYNTAX      INTEGER (0..7)
21          MAX-ACCESS   read-create
22          STATUS      current
23          DESCRIPTION
24              "The value of this parameter specifies the priority
25              assigned to a service flow. For uplink service flows,
26              the BS should use this parameter when determining
27              precedence in request service and grant generation,
28              and the SS shall preferentially select contention
29              Request opportunities for Priority Request CIDs
30              based on this priority. Higher numbers indicate higher
31              priority"
32          REFERENCE
33              "Subclause 11.13.5 in IEEE Std 802.16-2004"
34          ::= { wman2IfBsServiceClassEntry 3 }
35
36
37
38
39      wman2IfBsQoSMaxSustainedRate OBJECT-TYPE
40          SYNTAX      Unsigned32
41          UNITS       "b/s"
42          MAX-ACCESS   read-create
43          STATUS      current
44          DESCRIPTION
45              "This parameter defines the peak information rate
46              of the service. The rate is expressed in bits per
47              second and pertains to the SDUs at the input to
48              the system."
49          REFERENCE
50              "Subclause 11.13.6 in IEEE Std 802.16-2004"
51          ::= { wman2IfBsServiceClassEntry 4 }
52
53
54
55
56      wman2IfBsQoSMaxTrafficBurst OBJECT-TYPE
57          SYNTAX      Unsigned32
58          UNITS       "byte"
59          MAX-ACCESS   read-create
60          STATUS      current
61          DESCRIPTION
62              "This parameter defines the maximum burst size that
63              must be accommodated for the service."
64
65

```

```

1      REFERENCE
2          "Subclause 11.13.7 in IEEE Std 802.16-2004"
3      ::= { wman2IfBsServiceClassEntry 5 }
4
5
6  wman2IfBsQoSMinReservedRate OBJECT-TYPE
7      SYNTAX      Unsigned32
8      UNITS       "b/s"
9      MAX-ACCESS  read-create
10     STATUS      current
11     DESCRIPTION
12         "This parameter specifies the minimum rate reserved
13         for this service flow."
14     REFERENCE
15         "Subclause 11.13.8 in IEEE Std 802.16-2004"
16     ::= { wman2IfBsServiceClassEntry 6 }
17
18
19
20  wman2IfBsQoSToleratedJitter OBJECT-TYPE
21     SYNTAX      Unsigned32
22     UNITS       "millisecond"
23     MAX-ACCESS  read-create
24     STATUS      current
25     DESCRIPTION
26         "This parameter defines the Maximum delay
27         variation (jitter) for the connection."
28     REFERENCE
29         "Subclause 11.13.13 in IEEE Std 802.16-2004"
30     ::= { wman2IfBsServiceClassEntry 7 }
31
32
33
34
35  wman2IfBsQoSMaxLatency OBJECT-TYPE
36     SYNTAX      Unsigned32
37     UNITS       "millisecond"
38     MAX-ACCESS  read-create
39     STATUS      current
40     DESCRIPTION
41         "The value of this parameter specifies the maximum
42         latency between the reception of a packet by the BS
43         or SS on its network interface and the forwarding
44         of the packet to its RF Interface."
45     REFERENCE
46         "Subclause 11.13.14 in IEEE Std 802.16-2004"
47     ::= { wman2IfBsServiceClassEntry 8 }
48
49
50
51  wman2IfBsQoSFixedVsVariableSduInd OBJECT-TYPE
52     SYNTAX      INTEGER {variableLength(0),
53                        fixedLength(1)}
54     MAX-ACCESS  read-create
55     STATUS      current
56     DESCRIPTION
57         "The value of this parameter specifies whether the SDUs
58         on the service flow are variable-length (0) or
59         fixed-length (1). The parameter is used only if
60         packing is on for the service flow. The default value
61         is 0, i.e., variable-length SDUs."
62     REFERENCE
63
64
65

```

```

1         "Subclause 11.13.15 in IEEE Std 802.16-2004"
2     DEFVAL      { variableLength }
3     ::= { wman2IfBsServiceClassEntry 9 }
4
5
6     wman2IfBsQoSsduSize OBJECT-TYPE
7         SYNTAX      Unsigned32
8         UNITS       "byte"
9         MAX-ACCESS  read-create
10        STATUS      current
11        DESCRIPTION
12            "The value of this parameter specifies the length of the
13             SDU for a fixed-length SDU service flow. This parameter
14             is used only if packing is on and the service flow is
15             indicated as carrying fixed-length SDUs. The default
16             value is 49 bytes, i.e., VC-switched ATM cells with PHS.
17             The parameter is relevant for both ATM and Packet
18             Convergence Sublayers."
19
20        REFERENCE
21            "Subclause 11.13.16 in IEEE Std 802.16-2004"
22
23        DEFVAL      { 49 }
24        ::= { wman2IfBsServiceClassEntry 10 }
25
26
27     wman2IfBsQoSschedulingType OBJECT-TYPE
28         SYNTAX      Wman2IfSfSchedulingType
29         MAX-ACCESS  read-create
30         STATUS      current
31         DESCRIPTION
32             "Specifies the upstream scheduling service used for
33             upstream service flow. If the referenced parameter
34             is not present in the corresponding 802.16 QOS
35             Parameter Set of an upstream service flow, the
36             default value of this object is bestEffort(2)."

```

```

1      REFERENCE
2          "Subclause 11.13.18 in IEEE Std 802.16-2004"
3      ::= { wman2IfBsServiceClassEntry 13 }
4
5
6  wman2IfBsQosScArqBlockLifetime OBJECT-TYPE
7      SYNTAX      INTEGER (0 .. 65535)
8      UNITS       "10 us"
9      MAX-ACCESS  read-create
10     STATUS      current
11     DESCRIPTION
12         "The maximum time interval an ARQ fragment will be
13         managed by the transmitter ARQ machine, once
14         initial transmission of the fragment has occurred.
15         If transmission or retransmission of the fragment
16         is not acknowledged by the receiver before the
17         time limit is reached, the fragment is discarded.
18         A value of 0 means Infinite."
19     REFERENCE
20         "Subclause 11.13.18 in IEEE Std 802.16-2004"
21     DEFVAL      {0}
22     ::= { wman2IfBsServiceClassEntry 14 }
23
24
25  wman2IfBsQosScArqSyncLossTimeout OBJECT-TYPE
26     SYNTAX      INTEGER (0 .. 65535 )
27     UNITS       "10 us"
28     MAX-ACCESS  read-create
29     STATUS      current
30     DESCRIPTION
31         "The maximum interval before declaring a loss
32         of synchronization of the sender and receiver
33         state machines. A value of 0 means Infinite."
34     REFERENCE
35         "Subclause 11.13.18 in IEEE Std 802.16-2004"
36     DEFVAL      {0}
37     ::= { wman2IfBsServiceClassEntry 15 }
38
39
40  wman2IfBsQosScArqDeliverInOrder OBJECT-TYPE
41     SYNTAX      TruthValue
42     MAX-ACCESS  read-create
43     STATUS      current
44     DESCRIPTION
45         "Indicates whether or not data is to be delivered
46         by the receiving MAC to its client application
47         in the order in which data was handed off to the
48         originating MAC."
49     REFERENCE
50         "Subclause 11.13.18 in IEEE Std 802.16-2004"
51     ::= { wman2IfBsServiceClassEntry 16 }
52
53
54  wman2IfBsQosScArqRxPurgeTimeout OBJECT-TYPE
55     SYNTAX      INTEGER (0 .. 65535)
56     UNITS       "10 us"
57     MAX-ACCESS  read-create
58     STATUS      current
59

```

```

1      DESCRIPTION
2          "Indicates the time interval the ARQ window is advanced
3            after a fragment is received. A value of 0 means
4            Infinite."
5
6      REFERENCE
7          "Subclause 11.13.18 in IEEE Std 802.16-2004"
8
9      DEFVAL    {0}
10     ::= { wman2IfBsServiceClassEntry 17 }
11
12 wman2IfBsQoSScArqBlockSize OBJECT-TYPE
13     SYNTAX      INTEGER (1..2040)
14     UNITS        "byte"
15     MAX-ACCESS   read-create
16     STATUS       current
17     DESCRIPTION
18         "The value of this parameter specifies the size of an
19         ARQ block. This parameter shall be established by
20         negotiation during the connection creation dialog."
21
22     REFERENCE
23         "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
24
25     ::= { wman2IfBsServiceClassEntry 18 }
26
27 wman2IfBsQoSSCMinRsvdTolerableRate OBJECT-TYPE
28     SYNTAX      Unsigned32
29     UNITS        "b/s"
30     MAX-ACCESS   read-create
31     STATUS       current
32     DESCRIPTION
33         "Minimum Tolerable Traffic Rate = R (bits/sec) with
34         time base T(sec) means the following. Let S denote
35         additional demand accumulated at the MAC SAP of the
36         transmitter during an arbitrary time interval of the
37         length T. Then the amount of data forwarded at the
38         receiver to CS (in bits) during this interval should
39         be not less than min {S, R * T}."
40
41     REFERENCE
42         "Subclause 11.13.9 in IEEE Std 802.16-2004"
43
44     ::= { wman2IfBsServiceClassEntry 19 }
45
46
47 wman2IfBsQoSReqTxPolicy OBJECT-TYPE
48     SYNTAX      BITS {noBroadcastBwReq(0),
49                        reserved1(1),
50                        noPiggybackReq(2),
51                        noFragmentData(3),
52                        noPHS(4),
53                        noSduPacking(5),
54                        noCrc(6),
55                        reserved2(7)}
56
57     MAX-ACCESS   read-create
58     STATUS       current
59     DESCRIPTION
60         "The value of this parameter provides the capability to
61         specify certain attributes for the associated service
62         flow. An attribute is enabled by setting the
63
64
65

```

```

1           corresponding bit position to 1."
2   REFERENCE   "Subclause 11.13.12 in IEEE Std 802.16-2004"
3   ::= { wman2IfBsServiceClassEntry 20 }
4
5
6   wman2IfBsQoSServiceClassRowStatus OBJECT-TYPE
7       SYNTAX      RowStatus
8       MAX-ACCESS   read-create
9       STATUS       current
10      DESCRIPTION
11          "This object is used to create a new row or modify or
12          delete an existing row in this table.
13
14          If the implementator of this MIB has choosen not
15          to implement 'dynamic assignment' of profiles, this
16          object is not useful and should return noSuchName
17          upon SNMP request."
18      ::= { wman2IfBsServiceClassEntry 21 }
19
20
21   wman2IfBsClassifierRuleTable OBJECT-TYPE
22       SYNTAX      SEQUENCE OF Wman2IfBsClassifierRuleEntry
23       MAX-ACCESS   not-accessible
24       STATUS       current
25       DESCRIPTION
26          "This table contains packet classifier rules associated
27          with service flows."
28       REFERENCE
29          "Subclause 11.13.19.3.4 in IEEE Std 802.16-2004"
30       ::= { wman2IfBsPacketCs 4 }
31
32
33   wman2IfBsClassifierRuleEntry OBJECT-TYPE
34       SYNTAX      Wman2IfBsClassifierRuleEntry
35       MAX-ACCESS   not-accessible
36       STATUS       current
37       DESCRIPTION
38          "This table provides one row for each packet classifier
39          rule, and is indexed by ifIndex, wman2IfBsSfId, and
40          wman2IfBsClassifierRuleIndex. IfIdex is associated with
41          the BS sector. wman2IfBsSfId identifies the service flow,
42          while wman2IfBsClassifierRuleIndex identifies the packet
43          classifier rule."
44       INDEX { ifIndex, wman2IfBsSfId, wman2IfBsClassifierRuleIndex }
45       ::= { wman2IfBsClassifierRuleTable 1 }
46
47
48   Wman2IfBsClassifierRuleEntry ::= SEQUENCE {
49       wman2IfBsClassifierRuleIndex      Unsigned32,
50       wman2IfBsClassifierRulePriority    INTEGER,
51       wman2IfBsClassifierRuleIpTosLow    INTEGER,
52       wman2IfBsClassifierRuleIpTosHigh   INTEGER,
53       wman2IfBsClassifierRuleIpTosMask   INTEGER,
54       wman2IfBsClassifierRuleIpProtocol  Integer32,
55       wman2IfBsClassifierRuleIpSourceAddr InetAddress,
56       wman2IfBsClassifierRuleIpSourceMask InetAddress,
57       wman2IfBsClassifierRuleIpDestAddr  InetAddress,
58       wman2IfBsClassifierRuleIpDestMask  InetAddress,
59
60
61
62
63
64
65

```

```

1      wman2IfBsClassifierRuleSourcePortStart Integer32,
2      wman2IfBsClassifierRuleSourcePortEnd   Integer32,
3      wman2IfBsClassifierRuleDestPortStart   Integer32,
4      wman2IfBsClassifierRuleDestPortEnd     Integer32,
5      wman2IfBsClassifierRuleDestMacAddr     MacAddress,
6      wman2IfBsClassifierRuleDestMacMask     MacAddress,
7      wman2IfBsClassifierRuleSourceMacAddr   MacAddress,
8      wman2IfBsClassifierRuleSourceMacMask   MacAddress,
9      wman2IfBsClassifierRuleEnetProtocolType INTEGER,
10     wman2IfBsClassifierRuleEnetProtocol    Integer32,
11     wman2IfBsClassifierRuleUserPriLow      Integer32,
12     wman2IfBsClassifierRuleUserPriHigh     Integer32,
13     wman2IfBsClassifierRuleVlanId          Integer32,
14     wman2IfBsClassifierRuleState           INTEGER,
15     wman2IfBsClassifierRulePhsSize          Integer32,
16     wman2IfBsClassifierRulePhsMask         OCTET STRING,
17     wman2IfBsClassifierRulePhsVerify       Wman2IfPhsRuleVerify,
18     wman2IfBsClassifierRuleIpv6FlowLabel   Wman2IfIpv6FlowLabel,
19     wman2IfBsClassifierRuleBitMap          Wman2IfClassifierBitMap,
20     wman2IfBsClassifierRuleRowStatus       RowStatus}
21
22
23
24
25
26 wman2IfBsClassifierRuleIndex OBJECT-TYPE
27     SYNTAX      Unsigned32 (1..4294967295)
28     MAX-ACCESS  not-accessible
29     STATUS      current
30     DESCRIPTION
31         "An index is assigned to a classifier in BS classifiers
32         table"
33     ::= { wman2IfBsClassifierRuleEntry 1 }
34
35
36
37 wman2IfBsClassifierRulePriority OBJECT-TYPE
38     SYNTAX      INTEGER (0..255)
39     MAX-ACCESS  read-create
40     STATUS      current
41     DESCRIPTION
42         "The value specifies the priority for the Classifier, which
43         is used for determining the order of the Classifier. A
44         higher value indicates higher priority. Classifiers may
45         have priorities in the range 0..255."
46     REFERENCE
47         "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
48     DEFVAL      { 0 }
49     ::= { wman2IfBsClassifierRuleEntry 2 }
50
51
52
53
54 wman2IfBsClassifierRuleIpTosLow OBJECT-TYPE
55     SYNTAX      INTEGER (0..255)
56     MAX-ACCESS  read-create
57     STATUS      current
58     DESCRIPTION
59         "The low value of a range of TOS byte values. If the
60         referenced parameter is not present in a classifier, this
61         object reports the value of 0."
62     REFERENCE
63         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
64
65

```



```

1      ::= { wman2IfBsClassifierRuleEntry 3 }
2
3
4  wman2IfBsClassifierRuleIpTosHigh OBJECT-TYPE
5      SYNTAX      INTEGER (0..255)
6      MAX-ACCESS  read-create
7      STATUS      current
8      DESCRIPTION
9          "The 8-bit high value of a range of TOS byte values.
10         If the referenced parameter is not present in a classifier,
11         this object reports the value of 0."
12
13     REFERENCE
14         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
15     ::= { wman2IfBsClassifierRuleEntry 4 }
16
17
18  wman2IfBsClassifierRuleIpTosMask OBJECT-TYPE
19      SYNTAX      INTEGER (0..255)
20      MAX-ACCESS  read-create
21      STATUS      current
22      DESCRIPTION
23          "The value of this object specifies the matching parameter
24          for the IP type of service/DSCP [IETF RFC 2474] byte mask.
25          An IP packet with IP type of service (ToS) byte value
26          ip-tos matches this parameter if tos-low less than or
27          equal (ip-tos AND tos-mask) less than or equal tos-high."
28
29     REFERENCE
30         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
31     ::= { wman2IfBsClassifierRuleEntry 5 }
32
33
34
35  wman2IfBsClassifierRuleIpProtocol OBJECT-TYPE
36      SYNTAX      Integer32 (0..255)
37      MAX-ACCESS  read-create
38      STATUS      current
39      DESCRIPTION
40          "This object indicates the value of the IP Protocol field
41          required for IP packets to match this rule. If the
42          referenced parameter is not present in a classifier, this
43          object reports the value of 0."
44
45     REFERENCE
46         "Subclause 11.13.19.3.4.3 in IEEE Std 802.16-2004"
47     ::= { wman2IfBsClassifierRuleEntry 6 }
48
49
50
51  wman2IfBsClassifierRuleIpSourceAddr OBJECT-TYPE
52      SYNTAX      InetAddress
53      MAX-ACCESS  read-create
54      STATUS      current
55      DESCRIPTION
56          "This object specifies the value of the IP Source Address
57          required for packets to match this rule. An IP packet
58          matches the rule when the packet ip source address bitwise
59          ANDed with the wman2IfBsClassifierRuleIpSourceMask value
60          equals the wman2IfBsClassifierRuleIpSourceAddr value.
61          If the referenced parameter is not present in a classifier,
62          this object reports the value of 0.0.0.0."
63
64     REFERENCE
65

```

```

1         "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
2         ::= { wman2IfBsClassifierRuleEntry 7 }
3
4
5 wman2IfBsClassifierRuleIpSourceMask OBJECT-TYPE
6     SYNTAX      InetAddress
7     MAX-ACCESS  read-create
8     STATUS      current
9     DESCRIPTION
10        "This object specifies which bits of a packet's IP Source
11        Address that are compared to match this rule. An IP packet
12        matches the rule when the packet source address bitwise
13        ANDed with the
14        wman2IfBsClassifierRuleIpSourceMask value equals the
15        wman2IfBsClassifierRuleIpSourceAddr value.
16        If the referenced parameter is not present in a classifier,
17        this object reports the value of 0.0.0.0."
18    REFERENCE
19        "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
20    ::= { wman2IfBsClassifierRuleEntry 8 }
21
22
23 wman2IfBsClassifierRuleIpDestAddr OBJECT-TYPE
24     SYNTAX      InetAddress
25     MAX-ACCESS  read-create
26     STATUS      current
27     DESCRIPTION
28        "This object specifies the value of the IP Destination
29        Address required for packets to match this rule. An IP
30        packet matches the rule when the packet IP destination
31        address bitwise ANDed with the
32        wman2IfBsClassifierRuleIpDestMask value equals the
33        wman2IfBsClassifierRuleIpDestAddr value.
34        If the referenced parameter is not present in a
35        classifier, this object reports the value of 0.0.0.0."
36    REFERENCE
37        "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
38    ::= { wman2IfBsClassifierRuleEntry 9 }
39
40
41 wman2IfBsClassifierRuleIpDestMask OBJECT-TYPE
42     SYNTAX      InetAddress
43     MAX-ACCESS  read-create
44     STATUS      current
45     DESCRIPTION
46        "This object specifies which bits of a packet's IP
47        Destination Address that are compared to match this rule.
48        An IP packet matches the rule when the packet destination
49        address bitwise ANDed with the
50        wman2IfBsClassifierRuleIpDestMask value equals the
51        wman2IfBsClassifierRuleIpDestAddr value.
52        If the referenced parameter is not present in a classifier
53        , this object reports the value of 0.0.0.0."
54    REFERENCE
55        "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
56    ::= { wman2IfBsClassifierRuleEntry 10 }
57
58
59
60
61
62
63
64
65

```

```

1  wman2IfBsClassifierRuleSourcePortStart OBJECT-TYPE
2      SYNTAX      Integer32 (0..65535)
3      MAX-ACCESS  read-create
4      STATUS      current
5      DESCRIPTION
6          "This object specifies the low end inclusive range of
7          TCP/UDP source port numbers to which a packet is compared.
8          This object is irrelevant for non-TCP/UDP IP packets.
9          If the referenced parameter is not present in a
10         classifier, this object reports the value of 0."
11     REFERENCE
12         "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
13     ::= { wman2IfBsClassifierRuleEntry 11 }
14
15  wman2IfBsClassifierRuleSourcePortEnd OBJECT-TYPE
16      SYNTAX      Integer32 (0..65535)
17      MAX-ACCESS  read-create
18      STATUS      current
19      DESCRIPTION
20          "This object specifies the high end inclusive range of
21          TCP/UDP source port numbers to which a packet is compared.
22          This object is irrelevant for non-TCP/UDP IP packets.
23          If the referenced parameter is not present in a classifier,
24          this object reports the value of 65535."
25     REFERENCE
26         "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
27     ::= { wman2IfBsClassifierRuleEntry 12 }
28
29  wman2IfBsClassifierRuleDestPortStart OBJECT-TYPE
30      SYNTAX      Integer32 (0..65535)
31      MAX-ACCESS  read-create
32      STATUS      current
33      DESCRIPTION
34          "This object specifies the low end inclusive range of
35          TCP/UDP destination port numbers to which a packet is
36          compared. If the referenced parameter is not present
37          in a classifier, this object reports the value of 0."
38     REFERENCE
39         "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
40     ::= { wman2IfBsClassifierRuleEntry 13 }
41
42  wman2IfBsClassifierRuleDestPortEnd OBJECT-TYPE
43      SYNTAX      Integer32 (0..65535)
44      MAX-ACCESS  read-create
45      STATUS      current
46      DESCRIPTION
47          "This object specifies the high end inclusive range of
48          TCP/UDP destination port numbers to which a packet is
49          compared. If the referenced parameter is not present
50          in a classifier, this object reports the value of
51          65535."
52     REFERENCE
53         "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
54     ::= { wman2IfBsClassifierRuleEntry 14 }

```

```

1
2 wman2IfBsClassifierRuleDestMacAddr 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         wman2IfBsClassifierRuleDestMacMask equals the value of
10        wman2IfBsClassifierRuleDestMacAddr. 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     ::= { wman2IfBsClassifierRuleEntry 15 }
17
18 wman2IfBsClassifierRuleDestMacMask OBJECT-TYPE
19     SYNTAX      MacAddress
20     MAX-ACCESS  read-create
21     STATUS      current
22     DESCRIPTION
23         "An Ethernet packet matches an entry when its destination
24         MAC address bitwise ANDed with
25         wman2IfBsClassifierRuleDestMacMask equals the value of
26         wman2IfBsClassifierRuleDestMacAddr. If the referenced
27         parameter is not present in a classifier, this object
28         reports the value of '000000000000'H."
29
30     REFERENCE
31         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
32     ::= { wman2IfBsClassifierRuleEntry 16 }
33
34 wman2IfBsClassifierRuleSourceMacAddr OBJECT-TYPE
35     SYNTAX      MacAddress
36     MAX-ACCESS  read-create
37     STATUS      current
38     DESCRIPTION
39         "An Ethernet packet matches this entry when its source
40         MAC address bitwise ANDed with
41         wman2IfBsClassifierRuleSourceMacMask equals the value
42         of wman2IfBsClassifierRuleSourceMacAddr. If the
43         referenced parameter is not present in a classifier,
44         this object reports the value of '000000000000'H."
45
46     REFERENCE
47         "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
48     ::= { wman2IfBsClassifierRuleEntry 17 }
49
50 wman2IfBsClassifierRuleSourceMacMask OBJECT-TYPE
51     SYNTAX      MacAddress
52     MAX-ACCESS  read-create
53     STATUS      current
54     DESCRIPTION
55         "An Ethernet packet matches an entry when its source
56         MAC address bitwise ANDed with
57         wman2IfBsClassifierRuleSourceMacMask equals the value of
58

```

```

1          wman2IfBsClassifierRuleSourceMacAddr. If the referenced
2          parameter is not present in a classifier, this object
3          reports the value of '000000000000'H."
4
5      REFERENCE
6          "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
7      ::= { wman2IfBsClassifierRuleEntry 18 }
8
9
10     wman2IfBsClassifierRuleEnetProtocolType OBJECT-TYPE
11         SYNTAX      INTEGER {none(0),
12                        ethertype(1),
13                        dsap(2)}
14     MAX-ACCESS      read-create
15     STATUS           current
16     DESCRIPTION
17         "This object indicates the format of the layer 3 protocol
18         id in the Ethernet packet. A value of none(0) means that
19         the rule does not use the layer 3 protocol type as a
20         matching criteria. A value of ethertype(1) means that the
21         rule applies only to frames which contains an EtherType
22         value. Ethertype values are contained in packets using
23         the Dec-Intel-Xerox (DIX) encapsulation or the RFC1042
24         Sub-Network Access Protocol (SNAP) encapsulation formats.
25         A value of dsap(2) means that the rule applies only to
26         frames using the IEEE802.3 encapsulation format with a
27         Destination Service Access Point (DSAP) other than 0xAA
28         (which is reserved for SNAP). If the Ethernet frame
29         contains an 802.1P/Q Tag header (i.e. EtherType 0x8100),
30         this object applies to the embedded EtherType field within
31         the 802.1P/Q header. If the referenced parameter is not
32         present in a classifier, this object reports the value of
33         0."
34     REFERENCE
35         "Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
36     ::= { wman2IfBsClassifierRuleEntry 19 }
37
38
39     wman2IfBsClassifierRuleEnetProtocol OBJECT-TYPE
40         SYNTAX      Integer32 (0..65535)
41     MAX-ACCESS      read-create
42     STATUS           current
43     DESCRIPTION
44         "If wman2IfBsClassifierRuleEnetProtocolType is none(0),
45         this object is ignored when considering whether a packet
46         matches the current rule.
47         If wman2IfBsClassifierRuleEnetProtocolType is ethertype(1),
48         this object gives the 16-bit value of the EtherType that
49         the packet must match in order to match the rule.
50         If wman2IfBsClassifierRuleEnetProtocolType is dsap(2), the
51         lower 8 bits of this object's value must match the DSAP
52         byte of the packet in order to match the rule.
53         If the Ethernet frame contains an 802.1P/Q Tag header
54         (i.e. EtherType 0x8100), this object applies to the
55         embedded EtherType field within the 802.1P/Q header.
56         If the referenced parameter is not present in the
57         classifier, the value of this object is reported as 0."
58
59
60
61
62
63
64
65

```

```

1      REFERENCE
2          "Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
3      ::= { wman2IfBsClassifierRuleEntry 20 }
4
5
6  wman2IfBsClassifierRuleUserPriLow OBJECT-TYPE
7      SYNTAX      Integer32 (0..7)
8      MAX-ACCESS  read-create
9      STATUS      current
10     DESCRIPTION
11         "This object applies only to Ethernet frames using the
12          802.1P/Q tag header (indicated with EtherType 0x8100).
13          Such frames include a 16-bit Tag that contains a 3 bit
14          Priority field and a 12 bit VLAN number.
15          Tagged Ethernet packets must have a 3-bit Priority field
16          within the range of wman2IfBsClassifierRuleUserPriLow and
17          wman2IfBsClassifierRuleUserPriHigh in order to match this
18          rule.
19          If the referenced parameter is not present in the
20          classifier, the value of this object is reported as 0."
21     REFERENCE
22         "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
23     ::= { wman2IfBsClassifierRuleEntry 21 }
24
25
26  wman2IfBsClassifierRuleUserPriHigh OBJECT-TYPE
27      SYNTAX      Integer32 (0..7)
28      MAX-ACCESS  read-create
29      STATUS      current
30      DESCRIPTION
31         "This object applies only to Ethernet frames using the
32          802.1P/Q tag header (indicated with EtherType 0x8100).
33          Such frames include a 16-bit Tag that contains a 3 bit
34          Priority field and a 12 bit VLAN number.
35          Tagged Ethernet packets must have a 3-bit Priority
36          field within the range of wman2IfBsClassifierRuleUserPriLow
37          and wman2IfBsClassifierRuleUserPriHigh in order to match
38          this rule.
39          If the referenced parameter is not present in the
40          classifier, the value of this object is reported as 7."
41     REFERENCE
42         "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
43     ::= { wman2IfBsClassifierRuleEntry 22 }
44
45
46  wman2IfBsClassifierRuleVlanId OBJECT-TYPE
47      SYNTAX      Integer32 (0..4095)
48      MAX-ACCESS  read-create
49      STATUS      current
50      DESCRIPTION
51         "This object applies only to Ethernet frames using the
52          802.1P/Q tag header.
53          If this object's value is nonzero, tagged packets must
54          have a VLAN Identifier that matches the value in order
55          to match the rule.
56          Only the least significant 12 bits of this object's
57          value are valid.
58
59
60
61
62
63
64
65

```

```

1         If the referenced parameter is not present in the
2         classifier, the value of this object is reported as 0."
3     REFERENCE
4         "Subclause 11.13.19.3.4.12 in IEEE Std 802.16-2004"
5     ::= { wman2IfBsClassifierRuleEntry 23 }
6
7
8     wman2IfBsClassifierRuleState OBJECT-TYPE
9         SYNTAX      INTEGER {active(1),
10                        inactive(2)}
11         MAX-ACCESS  read-create
12         STATUS      deprecated
13         DESCRIPTION
14             "This object indicates whether or not the classifier is
15             enabled to classify packets to a Service Flow.
16             If the referenced parameter is not present in the
17             classifier, the value of this object is reported
18             as active(1)."

```

```

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             (wman2IfBsClassifierRulePhsSize/8). BS should use this value
6             to create a new PHS rule index (PHSI) and field (PHSF) as
7             defined in the standard. If flag phsMask in
8             wman2IfBsClassifierRuleBitMap is set to 0 and flag phsSize
9             in wman2IfBsClassifierRuleBitMap is set to 0, then BS can
10            still create PHS rules using its own custom mask (i.e. the
11            rule is not configured by NMS)."
```

REFERENCE

```

15          "Subclause 11.13.19.3.7.3 in IEEE Std 802.16-2004"
16          ::= { wman2IfBsClassifierRuleEntry 26 }
```

wman2IfBsClassifierRulePhsVerify OBJECT-TYPE

```

19          SYNTAX      Wman2IfPhsRuleVerify
20          MAX-ACCESS   read-create
21          STATUS       current
22          DESCRIPTION
23              "The value of this field indicates to the sending entity
24              whether or not the packet header contents are to be
25              verified prior to performing suppression."
26          DEFVAL      { phsVerifyEnable }
27          ::= { wman2IfBsClassifierRuleEntry 27 }
```

wman2IfBsClassifierRuleIpv6FlowLabel OBJECT-TYPE

```

32          SYNTAX      Wman2IfIpv6FlowLabel
33          MAX-ACCESS   read-create
34          STATUS       current
35          DESCRIPTION
36              "The value of this field specifies the matching values for
37              the IPv6 Flow label field."
38          ::= { wman2IfBsClassifierRuleEntry 28 }
```

wman2IfBsClassifierRuleBitMap OBJECT-TYPE

```

43          SYNTAX      Wman2IfClassifierBitMap
44          MAX-ACCESS   read-create
45          STATUS       current
46          DESCRIPTION
47              "This object indicates which parameter encodings were
48              actually present in the entry. A bit set to '1' indicates
49              the corresponding classifier encoding is present, and '0'
50              means otherwise"
51          ::= { wman2IfBsClassifierRuleEntry 29 }
```

wman2IfBsClassifierRuleRowStatus OBJECT-TYPE

```

56          SYNTAX      RowStatus
57          MAX-ACCESS   read-create
58          STATUS       current
59          DESCRIPTION
60              "This object is used to create a new row or modify or
61              delete an existing row in this table.
```



```

1         If the implementator of this MIB has choosen not
2         to implement 'dynamic assignment' of profiles, this
3         object is not useful and should return noSuchName
4         upon SNMP request."
5         ::= { wman2IfBsClassifierRuleEntry 30 }
6
7
8 wman2IfBsSsPacketCounterTable OBJECT-TYPE
9     SYNTAX      SEQUENCE OF Wman2IfBsSsPacketCounterEntry
10    MAX-ACCESS   not-accessible
11    STATUS       current
12    DESCRIPTION
13        "This table contains counters to keep track of the number
14        of packets and octets that have been received or
15        transmitted on the per service flow basis."
16    ::= { wman2IfBsPacketCs 5 }
17
18
19
20 wman2IfBsSsPacketCounterEntry OBJECT-TYPE
21     SYNTAX      Wman2IfBsSsPacketCounterEntry
22     MAX-ACCESS   not-accessible
23     STATUS       current
24     DESCRIPTION
25         "This table provides one row for each service flow, and
26         is indexed by ifIndex, wman2IfCmnCpsSfMacAddress, and
27         wman2IfCmnCpsSfId."
28     INDEX { ifIndex, wman2IfCmnCpsSfMacAddress,
29             wman2IfCmnCpsSfId }
30     ::= { wman2IfBsSsPacketCounterTable 1 }
31
32
33
34 Wman2IfBsSsPacketCounterEntry ::= SEQUENCE {
35     wman2IfBsSsMacSduCount          Counter64,
36     wman2IfBsSsOctetCount           Counter64,
37     wman2IfBsSsResetCounter         INTEGER,
38     wman2IfBsSsResetCounterTime     TimeStamp}
39
40
41
42 wman2IfBsSsMacSduCount OBJECT-TYPE
43     SYNTAX      Counter64
44     MAX-ACCESS   read-only
45     STATUS       current
46     DESCRIPTION
47         "This object counts the number of MAC SDUs that have
48         been transmitted or received."
49     ::= { wman2IfBsSsPacketCounterEntry 1 }
50
51
52
53 wman2IfBsSsOctetCount OBJECT-TYPE
54     SYNTAX      Counter64
55     MAX-ACCESS   read-only
56     STATUS       current
57     DESCRIPTION
58         "This object counts the number of octets of MAC SDUs
59         that have been transmitted or received."
60     ::= { wman2IfBsSsPacketCounterEntry 2 }
61
62
63
64 wman2IfBsSsResetCounter OBJECT-TYPE
65     SYNTAX      INTEGER {null(0),

```

```

1           resetCounter(1) }
2   MAX-ACCESS  read-write
3   STATUS      current
4   DESCRIPTION
5       "When this attribute is SET to resetCounter(1), the
6       corresponding entry of packet counters will be reset.
7       A GET operation performed on this object will always
8       return null(0). The counter is normally reset after
9       the packet count information is retrieved. "
10      ::= { wman2IfBsSsPacketCounterEntry 3 }
11
12
13
14   wman2IfBsSsResetCounterTime OBJECT-TYPE
15       SYNTAX      TimeStamp
16       MAX-ACCESS  read-only
17       STATUS      current
18       DESCRIPTION
19           "Indicates the date and time when the counter is
20           reset."
21       ::= { wman2IfBsSsPacketCounterEntry 4 }
22
23
24
25   --
26   -- wman2IfBsCps contain the Base Station Common Part Sublayer objects
27   --
28   wman2IfBsCps OBJECT IDENTIFIER ::= { wman2IfBsObjects 2 }
29
30
31   wman2IfBsRegisteredSsTable OBJECT-TYPE
32       SYNTAX      SEQUENCE OF Wman2IfBsRegisteredSsEntry
33       MAX-ACCESS  not-accessible
34       STATUS      current
35       DESCRIPTION
36           "This table contains the basic capability information
37           of SSs that have been negotiated and agreed between
38           BS and SS via REG-REQ and REG-RSP messages. An entry
39           in this table indicates the SS has entered and registered
40           into the BS."
41       REFERENCE
42           "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
43       ::= { wman2IfBsCps 1 }
44
45
46
47   wman2IfBsRegisteredSsEntry OBJECT-TYPE
48       SYNTAX      Wman2IfBsRegisteredSsEntry
49       MAX-ACCESS  not-accessible
50       STATUS      current
51       DESCRIPTION
52           "This table provides one row for each SS that has been
53           registered in the BS, and is indexed by
54           wman2IfBsSsMacAddress. The primary index is the ifIndex
55           with an ifType of propBWAmp2Mp, indicating the BS sector
56           with which the SS is associated. wman2IfBsSsMacAddress
57           identifies the SS being registered."
58       INDEX { ifIndex, wman2IfBsSsMacAddress }
59       ::= { wman2IfBsRegisteredSsTable 1 }
60
61
62
63   Wman2IfBsRegisteredSsEntry ::= SEQUENCE {
64
65

```

1	wman2IfBsSsMacAddress	MacAddress,
2	wman2IfBsSsBasicCid	Wman2IfCidType,
3	wman2IfBsSsPrimaryCid	Wman2IfCidType,
4	wman2IfBsSsSecondaryCid	Wman2IfCidType,
5	wman2IfBsSsManagementSupport	INTEGER,
6	wman2IfBsSsIpManagementMode	INTEGER,
7	wman2IfBsSs2ndMgmtArqEnable	TruthValue,
8	wman2IfBsSs2ndMgmtArqWindowSize	INTEGER,
9	wman2IfBsSs2ndMgmtArqDnLinkTxDelay	INTEGER,
10	wman2IfBsSs2ndMgmtArqUpLinkTxDelay	INTEGER,
11	wman2IfBsSs2ndMgmtArqDnLinkRxDelay	INTEGER,
12	wman2IfBsSs2ndMgmtArqUpLinkRxDelay	INTEGER,
13	wman2IfBsSs2ndMgmtArqBlockLifetime	INTEGER,
14	wman2IfBsSs2ndMgmtArqSyncLossTimeout	INTEGER,
15	wman2IfBsSs2ndMgmtArqDeliverInOrder	TruthValue,
16	wman2IfBsSs2ndMgmtArqRxPurgeTimeout	INTEGER,
17	wman2IfBsSs2ndMgmtArqBlockSize	INTEGER,
18	wman2IfBsSsVendorIdEncoding	OCTET STRING,
19	wman2IfBsSsAasBroadcastPermission	INTEGER,
20	wman2IfBsSsMaxTxPowerBpsk	Wman2IfMaxTxPowerType,
21	wman2IfBsSsMaxTxPowerQpsk	Wman2IfMaxTxPowerType,
22	wman2IfBsSsMaxTxPower16Qam	Wman2IfMaxTxPowerType,
23	wman2IfBsSsMaxTxPower64Qam	Wman2IfMaxTxPowerType,
24	wman2IfBsSsMacVersion	Wman2IfMacVersion}
25		
26	wman2IfBsSsMacAddress OBJECT-TYPE	
27	SYNTAX MacAddress	
28	MAX-ACCESS not-accessible	
29	STATUS current	
30	DESCRIPTION	
31	"The MAC address of SS is received from the RNG-REQ	
32	message. When SS registers, this MAC address is entered	
33	into the table, and used as the identifier to the SS."	
34	REFERENCE	
35	"Subclause 6.3.2.3.5 in IEEE Std 802.16-2004"	
36	::= { wman2IfBsRegisteredSsEntry 1 }	
37		
38	wman2IfBsSsBasicCid OBJECT-TYPE	
39	SYNTAX Wman2IfCidType	
40	MAX-ACCESS read-only	
41	STATUS current	
42	DESCRIPTION	
43	"The value of this object indicates the SS's basic CID	
44	that was sent in the RNG-RSP message."	
45	REFERENCE	
46	"Subclause 6.3.2.3.6 in IEEE Std 802.16-2004"	
47	::= { wman2IfBsRegisteredSsEntry 2 }	
48		
49	wman2IfBsSsPrimaryCid OBJECT-TYPE	
50	SYNTAX Wman2IfCidType	
51	MAX-ACCESS read-only	
52	STATUS current	
53	DESCRIPTION	
54	"The value of this object indicates the primary CID of the	

```

1         SS received from the RNG-RSP message."
2     REFERENCE
3         "Subclause 6.3.2.3.6 in IEEE Std 802.16-2004"
4         ::= { wman2IfBsRegisteredSsEntry 3 }
5
6
7     wman2IfBsSsSecondaryCid OBJECT-TYPE
8         SYNTAX      Wman2IfCidType
9         MAX-ACCESS   read-only
10        STATUS      current
11        DESCRIPTION
12            "The value of this object indicates the secondary
13             management CID present in the REG-RSP message. The value
14             should be null if the 2nd management connection is not
15             available."
16        REFERENCE
17            "Subclause 6.4.2.3.8 in IEEE Std 802.16-2004"
18            ::= { wman2IfBsRegisteredSsEntry 4 }
19
20
21
22
23     wman2IfBsSsManagementSupport OBJECT-TYPE
24         SYNTAX      INTEGER {unmanagedSs(0),
25                           managedSs(1)}
26         MAX-ACCESS   read-only
27         STATUS      current
28         DESCRIPTION
29             "This object indicates whether or not the SS is managed."
30         REFERENCE
31             "Subclause 11.7.2 in IEEE Std 802.16-2004"
32             ::= { wman2IfBsRegisteredSsEntry 5 }
33
34
35
36
37     wman2IfBsSsIpManagementMode OBJECT-TYPE
38         SYNTAX      INTEGER {unmanaged(0),
39                           ipManaged(1)}
40         MAX-ACCESS   read-only
41         STATUS      current
42         DESCRIPTION
43             "The IP management mode parameter dictates whether
44             the provider intends to manage the SS on an ongoing
45             basis via IP-based mechanisms."
46         REFERENCE
47             "Subclause 11.7.3 in IEEE Std 802.16-2004"
48             ::= { wman2IfBsRegisteredSsEntry 6 }
49
50
51
52
53     wman2IfBsSs2ndMgmtArqEnable OBJECT-TYPE
54         SYNTAX      TruthValue
55         MAX-ACCESS   read-only
56         STATUS      current
57         DESCRIPTION
58             "True(1) ARQ enabling is requested for the 2nd
59             management channel."
60         REFERENCE
61             "Subclause 11.13.18.1 in IEEE Std 802.16-2004"
62             ::= { wman2IfBsRegisteredSsEntry 7 }
63
64
65     wman2IfBsSs2ndMgmtArqWindowSize OBJECT-TYPE

```

```

1      SYNTAX      INTEGER (1 .. 1024)
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5
6          "Indicates the maximum number of unacknowledged
7          fragments at any time for 2nd management connection."
8      REFERENCE
9
10         "Subclause 11.13.18.2 in IEEE Std 802.16-2004"
11     ::= { wman2IfBsRegisteredSsEntry 8 }
12
13 wman2IfBsSs2ndMgmtArqDnLinkTxDelay OBJECT-TYPE
14     SYNTAX      INTEGER (0 .. 65535)
15     UNITS        "us"
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19
20         "The object defines the ARQ transmitter delay for
21         downlink transmission."
22     REFERENCE
23
24         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
25     ::= { wman2IfBsRegisteredSsEntry 9 }
26
27 wman2IfBsSs2ndMgmtArqUpLinkTxDelay OBJECT-TYPE
28     SYNTAX      INTEGER (0 .. 65535)
29     UNITS        "us"
30     MAX-ACCESS  read-only
31     STATUS      current
32     DESCRIPTION
33
34         "The object defines the ARQ transmitter delay for
35         uplink transmission."
36     REFERENCE
37
38         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
39     ::= { wman2IfBsRegisteredSsEntry 10 }
40
41 wman2IfBsSs2ndMgmtArqDnLinkRxDelay OBJECT-TYPE
42     SYNTAX      INTEGER (0 .. 65535)
43     UNITS        "us"
44     MAX-ACCESS  read-only
45     STATUS      current
46     DESCRIPTION
47
48         "The object defines the ARQ receiver delay for
49         downlink transmission."
50     REFERENCE
51
52         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
53     ::= { wman2IfBsRegisteredSsEntry 11 }
54
55 wman2IfBsSs2ndMgmtArqUpLinkRxDelay OBJECT-TYPE
56     SYNTAX      INTEGER (0 .. 65535)
57     UNITS        "us"
58     MAX-ACCESS  read-only
59     STATUS      current
60     DESCRIPTION
61
62         "The object defines the ARQ receiver delay for
63         uplink transmission."
64
65

```

```

1      REFERENCE
2          "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
3      ::= { wman2IfBsRegisteredSsEntry 12 }
4
5
6  wman2IfBsSs2ndMgmtArqBlockLifetime OBJECT-TYPE
7      SYNTAX      INTEGER (0 .. 65535)
8      UNITS       "10 us"
9      MAX-ACCESS  read-only
10     STATUS      current
11     DESCRIPTION
12         "The maximum time interval an ARQ fragment will be
13         managed by the transmitter ARQ machine, once
14         initial transmission of the fragment has occurred.
15         If transmission or retransmission of the fragment
16         is not acknowledged by the receiver before the
17         time limit is reached, the fragment is discarded.
18         A value of 0 means Infinite."
19     REFERENCE
20         "Subclause 11.13.18.4 in IEEE Std 802.16-2004"
21     DEFVAL      {0}
22     ::= { wman2IfBsRegisteredSsEntry 13 }
23
24
25  wman2IfBsSs2ndMgmtArqSyncLossTimeout OBJECT-TYPE
26     SYNTAX      INTEGER (0 .. 65535)
27     UNITS       "10 us"
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31         "The maximum interval before declaring a loss
32         of synchronization of the sender and receiver
33         state machines. A value of 0 means Infinite."
34     REFERENCE
35         "Subclause 11.13.18.5 in IEEE Std 802.16-2004"
36     DEFVAL      {0}
37     ::= { wman2IfBsRegisteredSsEntry 14 }
38
39
40  wman2IfBsSs2ndMgmtArqDeliverInOrder OBJECT-TYPE
41     SYNTAX      TruthValue
42     MAX-ACCESS  read-only
43     STATUS      current
44     DESCRIPTION
45         "Indicates whether or not data is to be delivered
46         by the receiving MAC to its client application
47         in the order in which data was handed off to the
48         originating MAC."
49     REFERENCE
50         "Subclause 11.13.18.6 in IEEE Std 802.16-2004"
51     ::= { wman2IfBsRegisteredSsEntry 15 }
52
53
54  wman2IfBsSs2ndMgmtArqRxPurgeTimeout OBJECT-TYPE
55     SYNTAX      INTEGER (0 .. 65535)
56     UNITS       "10 us"
57     MAX-ACCESS  read-only
58     STATUS      current
59

```

```

1      DESCRIPTION
2          "Indicates the time interval the ARQ window is advanced
3          after a fragment is received. A value of 0 means Infinite."
4      REFERENCE
5          "Subclause 11.13.18.7 in IEEE Std 802.16-2004"
6      DEFVAL      {0}
7      ::= { wman2IfBsRegisteredSsEntry 16 }
8
9
10
11  wman2IfBsSs2ndMgmtArqBlockSize OBJECT-TYPE
12      SYNTAX      INTEGER (1 .. 2040)
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "This parameter specifies the size of a ARQ block. This
17          parameter shall be established by negotiation during the
18          connection setup. The requester includes its desired
19          setting in the REQ message. The receiver of the REQ
20          message shall take the smaller of the value it prefers and
21          value in the REQ message. The minimum value is included in
22          the RSP message."
23      REFERENCE
24          "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
25      ::= { wman2IfBsRegisteredSsEntry 17 }
26
27
28
29
30  wman2IfBsSsVendorIdEncoding OBJECT-TYPE
31      SYNTAX      OCTET STRING (SIZE(3))
32      MAX-ACCESS  read-only
33      STATUS      current
34      DESCRIPTION
35          "The value field contains the vendor identification
36          specified by the 3 byte vendor-specific organizationally
37          unique identifier of the SS or BS MAC address. A vendor ID
38          used in a REG-REQ shall be the Vendor ID of the SS sending
39          the request. A vendor ID used in a REG-RSP shall be the
40          Vendor ID of the BS sending the response."
41      REFERENCE
42          "Subclause 11.1.5 in IEEE Std 802.16-2004"
43      ::= { wman2IfBsRegisteredSsEntry 18 }
44
45
46
47
48  wman2IfBsSsAasBroadcastPermission OBJECT-TYPE
49      SYNTAX      INTEGER {contBasedBwReqPermitted(0),
50                      contBasedBwReqNotPermitted(1)}
51      MAX-ACCESS  read-only
52      STATUS      current
53      DESCRIPTION
54          "This parameter specifies if SS can issue contention-based
55          bandwidth request or not."
56      REFERENCE
57          "Subclause 11.6 in IEEE Std 802.16-2004"
58      ::= { wman2IfBsRegisteredSsEntry 19 }
59
60
61
62  wman2IfBsSsMaxTxPowerBpsk OBJECT-TYPE
63      SYNTAX      Wman2IfMaxTxPowerType
64      MAX-ACCESS  read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "The maximum available power for BPSK. 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          the 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     ::= { wman2IfBsRegisteredSsEntry 20 }
13
14
15 wman2IfBsSsMaxTxPowerQpsk OBJECT-TYPE
16     SYNTAX      Wman2IfMaxTxPowerType
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20         "The maximum available power for QPSK. The maximum power
21         parameters are reported in dBm and quantized in 0.5 dBm
22         steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
23         (encoded 0xFF). Values outside this range shall be assigned
24         to closest extreme. This parameter is only applicable to
25         systems supporting the SCa, OFDM or OFDMA PHY."
26     REFERENCE
27         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
28     ::= { wman2IfBsRegisteredSsEntry 21 }
29
30
31 wman2IfBsSsMaxTxPower16Qam OBJECT-TYPE
32     SYNTAX      Wman2IfMaxTxPowerType
33     MAX-ACCESS  read-only
34     STATUS      current
35     DESCRIPTION
36         "The maximum available power for 16-QAM constellations.
37         The maximum power parameters are reported in dBm and
38         quantized in 0.5 dBm steps ranging from -64 dBm (encoded
39         0x00) to 63.5 dBm (encoded 0xFF). Values outside this
40         range shall be assigned the closest extreme. This parameter
41         is only applicable to systems supporting the SCa, OFDM or
42         OFDMA PHY."
43     REFERENCE
44         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
45     ::= { wman2IfBsRegisteredSsEntry 22 }
46
47
48 wman2IfBsSsMaxTxPower64Qam OBJECT-TYPE
49     SYNTAX      Wman2IfMaxTxPowerType
50     MAX-ACCESS  read-only
51     STATUS      current
52     DESCRIPTION
53         "The maximum available power for 64-QAM constellations.
54         The maximum power parameters are reported in dBm and
55         quantized in 0.5 dBm steps ranging from -64 dBm (encoded
56         0x00) to 63.5 dBm (encoded 0xFF). Values outside this
57         range shall be assigned the closest extreme. Ss that do
58         not support QAM64 shall report the value of 0x00. This
59
60
61
62
63
64
65

```



```

1         parameter is only applicable to systems supporting the SCA,
2         OFDM or OFDMA PHY."
3
4     REFERENCE
5         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
6     ::= { wman2IfBsRegisteredSsEntry 23 }
7
8 wman2IfBsSsMacVersion OBJECT-TYPE
9     SYNTAX      Wman2IfMacVersion
10    MAX-ACCESS   read-only
11    STATUS       current
12    DESCRIPTION
13        "This parameter specifies the version of 802.16 to which the
14        message originator conforms."
15    REFERENCE
16        "Subclause 11.1.3 in IEEE Std 802.16-2004"
17    ::= { wman2IfBsRegisteredSsEntry 24 }
18
19 --
20 -- wman2IfBsConfigurationTable contains global parameters common in BS
21 --
22 wman2IfBsConfigurationTable OBJECT-TYPE
23     SYNTAX      SEQUENCE OF Wman2IfBsConfigurationEntry
24     MAX-ACCESS   not-accessible
25     STATUS       current
26     DESCRIPTION
27         "This table provides one row for each BS sector that
28         contains the BS system parameters as defined in Subclause
29         10.1 of [3]. The objects in this table define the default
30         behaviour of the BS for 2nd Management connection
31         scheduling and SFID allocation as well as configuration
32         parameters of the CPS scheduler and AAS system."
33     REFERENCE
34         "Subclause 10.1 in IEEE Std 802.16-2004"
35     ::= { wman2IfBsCps 2 }
36
37 wman2IfBsConfigurationEntry OBJECT-TYPE
38     SYNTAX      Wman2IfBsConfigurationEntry
39     MAX-ACCESS   not-accessible
40     STATUS       current
41     DESCRIPTION
42         "This table is indexed by ifIndex with an ifType of
43         propBWAmp2Mp."
44     INDEX { ifIndex }
45     ::= { wman2IfBsConfigurationTable 1 }
46
47 Wman2IfBsConfigurationEntry ::= SEQUENCE {
48     wman2IfBsDcdInterval          INTEGER,
49     wman2IfBsUcdInterval          INTEGER,
50     wman2IfBsUcdTransition        INTEGER,
51     wman2IfBsDcdTransition        INTEGER,
52     wman2IfBsInitialRangingInterval INTEGER,
53     wman2IfBsSsULMapProcTime      Unsigned32,
54     wman2IfBsSsRangRespProcTime   Unsigned32,
55     wman2IfBsT5Timeout            INTEGER,

```

```

1      wman2IfBsT9Timeout      INTEGER,
2      wman2IfBsT13Timeout     INTEGER,
3      wman2IfBsT15Timeout     INTEGER,
4      wman2IfBsT17Timeout     INTEGER,
5      wman2IfBsT27IdleTimer   Unsigned32,
6      wman2IfBsT27ActiveTimer Unsigned32,
7      wman2IfBs2ndMgmtDlQoSProfileIndex INTEGER,
8      wman2IfBs2ndMgmtUlQoSProfileIndex INTEGER,
9      wman2IfBsAutoSfidEnabled INTEGER,
10     wman2IfBsAutoSfidRangeMin Unsigned32,
11     wman2IfBsAutoSfidRangeMax Unsigned32,
12     wman2IfBsAasChanFbckReqFreq INTEGER,
13     wman2IfBsAasBeamSelectFreq INTEGER,
14     wman2IfBsAasChanFbckReqResolution INTEGER,
15     wman2IfBsAasBeamReqResolution INTEGER,
16     wman2IfBsAasNumOptDiversityZones INTEGER,
17     wman2IfBsResetSector     INTEGER}
18
19 wman2IfBsDcdInterval OBJECT-TYPE
20     SYNTAX      INTEGER (0..10000)
21     UNITS       "milliseconds"
22     MAX-ACCESS  read-write
23     STATUS      current
24     DESCRIPTION
25         "Time between transmission of DCD messages in ms."
26         ::= { wman2IfBsConfigurationEntry 1 }
27
28 wman2IfBsUcdInterval OBJECT-TYPE
29     SYNTAX      INTEGER (0..10000)
30     UNITS       "milliseconds"
31     MAX-ACCESS  read-write
32     STATUS      current
33     DESCRIPTION
34         "Time between transmission of UCD messages in ms."
35         ::= { wman2IfBsConfigurationEntry 2 }
36
37 wman2IfBsUcdTransition OBJECT-TYPE
38     SYNTAX      INTEGER (2..65535)
39     UNITS       "Number of MAC Frames"
40     MAX-ACCESS  read-write
41     STATUS      current
42     DESCRIPTION
43         "The time the BS shall wait after transmitting a UCD message
44         with an incremented Configuration Change Count before
45         issuing a UL-MAP message referring to
46         Uplink_Burst_Profiles defined in that UCD message."
47         ::= { wman2IfBsConfigurationEntry 3 }
48
49 wman2IfBsDcdTransition OBJECT-TYPE
50     SYNTAX      INTEGER (2..65535)
51     UNITS       "Number of MAC Frames"
52     MAX-ACCESS  read-write
53     STATUS      current
54     DESCRIPTION
55

```

```

1         "The time the BS shall wait after transmitting a DCD message
2         with an incremented Configuration Change Count before
3         issuing a DL-MAP message referring to
4         Downlink_Burst_Profiles defined in that DCD message."
5         ::= { wman2IfBsConfigurationEntry 4 }
6
7
8 wman2IfBsInitialRangingInterval OBJECT-TYPE
9     SYNTAX      INTEGER(0..2000)
10    UNITS       "milliseconds"
11    MAX-ACCESS   read-write
12    STATUS       current
13    DESCRIPTION
14        "Time between Initial Ranging regions assigned by the BS
15        in ms."
16    ::= { wman2IfBsConfigurationEntry 5 }
17
18
19
20 wman2IfBsSsULMapProcTime OBJECT-TYPE
21     SYNTAX      Unsigned32 (200 .. 4294967295)
22     UNITS       "micro seconds"
23     MAX-ACCESS   read-write
24     STATUS       current
25     DESCRIPTION
26         "Time provided between arrival of the last bit of a UL-MAP
27         at an SS and effectiveness of that map in us."
28     ::= { wman2IfBsConfigurationEntry 6 }
29
30
31
32 wman2IfBsSsRangRespProcTime OBJECT-TYPE
33     SYNTAX      Unsigned32 (10000 .. 4294967295)
34     UNITS       "micro seconds"
35     MAX-ACCESS   read-write
36     STATUS       current
37     DESCRIPTION
38         "Time allowed for an SS following receipt of a ranging
39         response before it is expected to reply to an invited
40         ranging request in us."
41     ::= { wman2IfBsConfigurationEntry 7 }
42
43
44
45 wman2IfBsT5Timeout OBJECT-TYPE
46     SYNTAX      INTEGER (0 .. 2000)
47     UNITS       "milliseconds"
48     MAX-ACCESS   read-write
49     STATUS       current
50     DESCRIPTION
51         "Wait for Uplink Channel Change Response in ms."
52     ::= { wman2IfBsConfigurationEntry 8 }
53
54
55
56 wman2IfBsT9Timeout OBJECT-TYPE
57     SYNTAX      INTEGER (300 .. 65535)
58     UNITS       "milliseconds"
59     MAX-ACCESS   read-write
60     STATUS       current
61     DESCRIPTION
62         "Registration Timeout, the time allowed between the BS
63         sending a RNG-RSP (success) to an SS, and receiving a
64
65

```

```

1          SBC-REQ from that same SS in ms."
2      ::= { wman2IfBsConfigurationEntry 9 }
3
4
5      wman2IfBsT13Timeout OBJECT-TYPE
6          SYNTAX      INTEGER (15 .. 65535)
7          UNITS        "minutes"
8          MAX-ACCESS   read-write
9          STATUS       current
10         DESCRIPTION
11             "The time allowed for an SS, following receipt of a
12              REG-RSP message to send a TFTP-CPLT message to the BS
13              in min."
14         ::= { wman2IfBsConfigurationEntry 10 }
15
16
17
18      wman2IfBsT15Timeout OBJECT-TYPE
19          SYNTAX      INTEGER (20 .. 65535)
20          UNITS        "milliseconds"
21          MAX-ACCESS   read-write
22          STATUS       current
23          DESCRIPTION
24              "Wait for MCA-RSP in ms."
25          ::= { wman2IfBsConfigurationEntry 11 }
26
27
28
29      wman2IfBsT17Timeout OBJECT-TYPE
30          SYNTAX      INTEGER (5 .. 65535)
31          UNITS        "minutes"
32          MAX-ACCESS   read-write
33          STATUS       current
34          DESCRIPTION
35              "Time allowed for SS to complete SS Authorization and
36              Key Exchange in minutes."
37          ::= { wman2IfBsConfigurationEntry 12 }
38
39
40
41      wman2IfBsT27IdleTimer OBJECT-TYPE
42          SYNTAX      Unsigned32 (10000 .. 4294967295)
43          UNITS        "us"
44          MAX-ACCESS   read-write
45          STATUS       current
46          DESCRIPTION
47              "Maximum time between unicast grants to SS when BS believes
48              SS uplink transmission quality is good enough."
49          ::= { wman2IfBsConfigurationEntry 13 }
50
51
52
53      wman2IfBsT27ActiveTimer OBJECT-TYPE
54          SYNTAX      Unsigned32 (10000 .. 4294967295)
55          UNITS        "us"
56          MAX-ACCESS   read-write
57          STATUS       current
58          DESCRIPTION
59              "Maximum time between unicast grants to SS when BS believes
60              SS uplink transmission quality is not good enough."
61          ::= { wman2IfBsConfigurationEntry 14 }
62
63
64
65

```

```

1  wman2IfBs2ndMgmtDlQoSProfileIndex OBJECT-TYPE
2      SYNTAX      INTEGER (1..65535)
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "This object defines the index of a row in
7          wman2IfBsServiceClassTable which is used to obtain all QoS
8          parameters required for the BS downlink 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         ::= { wman2IfBsConfigurationEntry 15 }
14
15  wman2IfBs2ndMgmtUlQoSProfileIndex OBJECT-TYPE
16      SYNTAX      INTEGER (1..65535)
17      MAX-ACCESS  read-write
18      STATUS      current
19      DESCRIPTION
20          "This object defines the index of a row in
21          wman2IfBsServiceClassTable which is used to obtain all QoS
22         parameters required for the BS uplink scheduler to
23         properly allocate and manage the bandwidth and schedule
24         the 2nd Management Connection traffic. The 2nd Management
25         Connection traffic doesn't differ from Traffic Connection
26         traffic in the area of QoS management."
27         ::= { wman2IfBsConfigurationEntry 16 }
28
29  wman2IfBsAutoSfidEnabled OBJECT-TYPE
30      SYNTAX      INTEGER {autoSfidDisabled(0),
31                        autoSfidEnabled(1)}
32      MAX-ACCESS  read-write
33      STATUS      current
34      DESCRIPTION
35          "This object defines whether the BS is allowed to
36          autonomously allocate SFIDs. When the object is set to
37          autoSfidEnabled, the BS is allowed to autonomously allocate
38          SFIDs from the range of allowed values defined by
39          wman2IfBsConfigExtAutoSfidRangeMin and
40          wman2IfBsConfigExtAutoSfidRangeMax. A SF is created
41          autonomously when it has not been provisioned in the
42          wman2IfBsProvisionedSfTable and may be initiated by either
43          the SS or BS. The BS should always initiate SF creation
44          based on the provisioned Service flows configured in
45          wman2IfBsProvisionedSfTable."
46          REFERENCE
47              "Subclause 11.13.1 in IEEE Std 802.16-2004"
48              ::= { wman2IfBsConfigurationEntry 17 }
49
50  wman2IfBsAutoSfidRangeMin OBJECT-TYPE
51      SYNTAX      Unsigned32 ( 1 .. 4294967295)
52      MAX-ACCESS  read-write
53      STATUS      current
54      DESCRIPTION

```

```

1         "This object defines the minimum value of the range of SFID
2         values allocated for the BS sector for the purpose of
3         autonomous creation of service flows. This value is used
4         when the object wman2IfBsAutoSfidEnabled allows
5         autonomous creation of SFIDs."
6
7     REFERENCE
8         "Subclause 11.13.1 in IEEE Std 802.16-2004"
9     ::= { wman2IfBsConfigurationEntry 18 }
10
11
12 wman2IfBsAutoSfidRangeMax OBJECT-TYPE
13     SYNTAX      Unsigned32 ( 1 .. 4294967295)
14     MAX-ACCESS  read-write
15     STATUS      current
16     DESCRIPTION
17         "This object defines the maximum value of the range of SFID
18         values allocated for the BS sector for the purpose of
19         autonomous creation of the service flows. This value is
20         used when the object wman2IfBsAutoSfidEnabled allows
21         autonomous creation of SFIDs."
22
23     REFERENCE
24         "Subclause 11.13.1 in IEEE Std 802.16-2004"
25     ::= { wman2IfBsConfigurationEntry 19 }
26
27
28
29 wman2IfBsAasChanFbckReqFreq OBJECT-TYPE
30     SYNTAX      INTEGER (5..10000)
31     UNITS        "ms"
32     MAX-ACCESS  read-write
33     STATUS      current
34     DESCRIPTION
35         "This object defines AAS channel feedback request frequency.
36         It controls the frequency of downlink beam measurements.
37         The relevant MAC messages are AAS-FBCK-REQ/RSP"
38
39     REFERENCE
40         "Subclause 6.3.2.3.40 in IEEE Std 802.16-2004"
41     ::= { wman2IfBsConfigurationEntry 20 }
42
43
44
45 wman2IfBsAasBeamSelectFreq OBJECT-TYPE
46     SYNTAX      INTEGER (5..10000)
47     UNITS        "ms"
48     MAX-ACCESS  read-write
49     STATUS      current
50     DESCRIPTION
51         "This object defines AAS beam select frequency.
52         It controls how often SS issues beam select messages.
53         The relevant MAC message is AAS_Beam_Select"
54
55     REFERENCE
56         "Subclause 6.3.2.3.41 in IEEE Std 802.16-2004"
57     ::= { wman2IfBsConfigurationEntry 21 }
58
59
60
61 wman2IfBsAasChanFbckReqResolution OBJECT-TYPE
62     SYNTAX      INTEGER { aasChanFbckRes00(0),
63                           aasChanFbckRes01(1),
64                           aasChanFbckRes10(2),
65                           aasChanFbckRes11(3) }

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "This object defines AAS feedback request frequency
5          measurements resolution. It is coded as follows:
6          aasChanFbckRes00 - every 4th carrier
7                          (-100, -96, -92, .., 100)
8          aasChanFbckRes01 - every 8th carrier
9                          (-100, -92, -84, .., 100)
10         aasChanFbckRes10 - every 16th carrier
11                          (-100, -84, -68, .., 100)
12         aasChanFbckRes11 - every 32th carrier
13                          (-100, -68, -36, .., 100)"
14
15      REFERENCE
16          "Subclause 8.3.6.4 in IEEE Std 802.16-2004"
17      ::= { wman2IfBsConfigurationEntry 22 }
18
19  wman2IfBsAasBeamReqResolution OBJECT-TYPE
20      SYNTAX      INTEGER {aasBeamReqRes000(0),
21                          aasBeamReqRes001(1),
22                          aasBeamReqRes010(2),
23                          aasBeamReqRes011(3),
24                          aasBeamReqRes100(4)}
25
26      MAX-ACCESS  read-write
27      STATUS      current
28      DESCRIPTION
29          "This object defines AAS beam select request resolution
30          parameter. It is coded as follows:
31          aasBeamReqRes000 - every 4th carrier
32          aasBeamReqRes001 - every 8th carrier
33          aasBeamReqRes010 - every 16th carrier
34          aasBeamReqRes011 - every 32th carrier
35          aasBeamReqRes100 - every 64th carrier"
36
37      REFERENCE
38          "Subclause 8.3.6.5 in IEEE Std 802.16-2004"
39      ::= { wman2IfBsConfigurationEntry 23 }
40
41  wman2IfBsAasNumOptDiversityZones OBJECT-TYPE
42      SYNTAX      INTEGER (0..65535)
43
44      MAX-ACCESS  read-write
45      STATUS      current
46      DESCRIPTION
47          "This object defines the number of optional diversity zones
48          transmitted in downlink."
49
50      REFERENCE
51          "Figure 209 in IEEE Std 802.16-2004"
52      ::= { wman2IfBsConfigurationEntry 24 }
53
54  wman2IfBsResetSector OBJECT-TYPE
55      SYNTAX      INTEGER {actionResetSectorNoAction(0),
56                          actionResetSector(1)}
57
58      MAX-ACCESS  read-write
59      STATUS      current
60      DESCRIPTION
61

```

```

1      "This object should be implemented as follows:
2      - When set to actionsResetSector value, instructs BS to
3        reset the sector identified by ifIndex. As a result of
4        this action the Phy and Mac of this sector should be
5        reinitialised.
6      - When set to value different than actionsResetSector it
7        should be ignored
8      - When read it should return actionsResetSectorNoAction"
9      ::= { wman2IfBsConfigurationEntry 25 }
10
11
12
13  --
14  -- Base Station Channel Measurement Table
15  --
16  wman2IfBsChannelMeasurementTable OBJECT-TYPE
17      SYNTAX      SEQUENCE OF Wman2IfBsChannelMeasurementEntry
18      MAX-ACCESS  not-accessible
19      STATUS      current
20      DESCRIPTION
21          "This table contains channel measurement information as
22          derived from BS measurement of uplink signal from SS,
23          and the downlink signal as reported from SS using
24          REP-REQ/RSP messages. The table shall be maintained as
25          FIFO to store measurement samples that can be used to
26          create RSSI and CINR histogram report. When the
27          measurement entry for a SS reaches the limit, the oldest
28          entry shall be deleted as the new entry is added to the
29          table."
30      REFERENCE
31          "6.3.2.3.33 in IEEE Std 802.16-2004"
32      ::= { wman2IfBsCps 3 }
33
34  wman2IfBsChannelMeasurementEntry OBJECT-TYPE
35      SYNTAX      Wman2IfBsChannelMeasurementEntry
36      MAX-ACCESS  not-accessible
37      STATUS      current
38      DESCRIPTION
39          "Each entry in the table contains RSSI and CINR
40          signal quality measurement on signal received from the SS.
41          The primary index is the ifIndex with ifType of propBWAmp2Mp
42          identifying the BS sector. wman2IfBsSsMacAddress identifies
43          the SS from which the signal was received.
44          wman2IfBsChannelDirection is the index to the direction of
45          the channel. wman2IfBsHistogramIndex is the index to
46          histogram samples. Since there is no time stamp in the
47          table, wman2IfBsHistogramIndex should be increased
48          monotonically, and wraps around when it reaches the
49          implementation specific limit."
50      INDEX      { ifIndex,
51                  wman2IfBsSsMacAddress,
52                  wman2IfBsChannelDirection,
53                  wman2IfBsHistogramIndex }
54      ::= { wman2IfBsChannelMeasurementTable 1 }
55
56  Wman2IfBsChannelMeasurementEntry ::= SEQUENCE {
57

```



```

1      wman2IfBsChannelDirection          INTEGER,
2      wman2IfBsHistogramIndex            Unsigned32,
3      wman2IfBsChannelNumber             Wman2IfChannelNumber,
4      wman2IfBsStartFrame                 INTEGER,
5      wman2IfBsDuration                   INTEGER,
6      wman2IfBsBasicReport                BITS,
7      wman2IfBsMeanCinrReport             INTEGER,
8      wman2IfBsMeanRssiReport             INTEGER,
9      wman2IfBsStdDeviationCinrReport     INTEGER,
10     wman2IfBsStdDeviationRssiReport     INTEGER}
11
12
13
14 wman2IfBsChannelDirection OBJECT-TYPE
15     SYNTAX      INTEGER {downstream(1),
16                      upstream(2)}
17     MAX-ACCESS  not-accessible
18     STATUS      current
19     DESCRIPTION
20         "wman2IfBsChannelDirection identifies the direction of a
21         a channel where the measurement takes place."
22     ::= { wman2IfBsChannelMeasurementEntry 1 }
23
24
25
26 wman2IfBsHistogramIndex OBJECT-TYPE
27     SYNTAX      Unsigned32 (1 .. 4294967295)
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31         "wman2IfBsHistogramIndex identifies the histogram samples
32         in the table for each subscriber station."
33     ::= { wman2IfBsChannelMeasurementEntry 2 }
34
35
36
37 wman2IfBsChannelNumber OBJECT-TYPE
38     SYNTAX      Wman2IfChannelNumber
39     MAX-ACCESS  read-only
40     STATUS      current
41     DESCRIPTION
42         "Physical channel number to be reported on is only
43         applicable to license exempt band. For licensed band,
44         this parameter should be null."
45     REFERENCE
46         "Subclause 11.12 in IEEE Std 802.16-2004"
47     ::= { wman2IfBsChannelMeasurementEntry 3 }
48
49
50
51 wman2IfBsStartFrame OBJECT-TYPE
52     SYNTAX      INTEGER (0..65535)
53     MAX-ACCESS  read-only
54     STATUS      current
55     DESCRIPTION
56         "Frame number in which measurement for this channel
57         started."
58     REFERENCE
59         "Subclause 11.12 in IEEE Std 802.16-2004"
60     ::= { wman2IfBsChannelMeasurementEntry 4 }
61
62
63
64 wman2IfBsDuration OBJECT-TYPE
65

```

```

1      SYNTAX      INTEGER (0 .. 16777215)
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "Cumulative measurement duration on the channel in
6          multiples of Ts. For any value exceeding 0xFFFFFFFF,
7          report 0xFFFFFFFF."
8
9      REFERENCE
10         "Subclause 11.12 in IEEE Std 802.16-2004"
11
12     ::= { wman2IfBsChannelMeasurementEntry 5 }
13
14 wman2IfBsBasicReport OBJECT-TYPE
15     SYNTAX      BITS {wirelessHuman(0),
16                     unknownTransmission(1),
17                     primaryUser(2),
18                     channelNotMeasured(3)}
19     MAX-ACCESS  read-only
20     STATUS      current
21     DESCRIPTION
22         "Bit #0: WirelessHUMAN detected on the channel
23         Bit #1: Unknown transmissions detected on the channel
24         Bit #2: Primary User detected on the channel
25         Bit #3: Unmeasured. Channel not measured"
26
27     REFERENCE
28         "Subclause 11.12 in IEEE Std 802.16-2004"
29
30     ::= { wman2IfBsChannelMeasurementEntry 6 }
31
32
33 wman2IfBsMeanCinrReport OBJECT-TYPE
34     SYNTAX      INTEGER (0 .. 41)
35     UNITS        "dB"
36     MAX-ACCESS  read-only
37     STATUS      current
38     DESCRIPTION
39         "Mean CINR report."
40
41     REFERENCE
42         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
43
44     ::= { wman2IfBsChannelMeasurementEntry 7 }
45
46
47 wman2IfBsMeanRssiReport OBJECT-TYPE
48     SYNTAX      INTEGER (0 .. 83)
49     UNITS        "dBm"
50     MAX-ACCESS  read-only
51     STATUS      current
52     DESCRIPTION
53         "Mean RSSI report."
54
55     REFERENCE
56         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
57
58     ::= { wman2IfBsChannelMeasurementEntry 8 }
59
60 wman2IfBsStdDeviationCinrReport OBJECT-TYPE
61     SYNTAX      INTEGER (0 .. 41)
62     UNITS        "dB"
63     MAX-ACCESS  read-only
64     STATUS      current
65

```

```

1      DESCRIPTION
2          "Standard deviation CINR report."
3      REFERENCE
4          "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
5      ::= { wman2IfBsChannelMeasurementEntry 9 }
6
7
8      wman2IfBsStdDeviationRssiReport OBJECT-TYPE
9          SYNTAX      INTEGER (0 .. 83)
10         UNITS        "dB"
11         MAX-ACCESS   read-only
12         STATUS       current
13         DESCRIPTION
14             "Standard deviation RSSI report."
15         REFERENCE
16             "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
17         ::= { wman2IfBsChannelMeasurementEntry 10 }
18
19
20
21
22
23
24
25      wman2IfBsCapabilities OBJECT IDENTIFIER ::= { wman2IfBsCps 4 }
26
27
28      wman2IfBsSsReqCapabilitiesTable OBJECT-TYPE
29          SYNTAX      SEQUENCE OF Wman2IfBsSsReqCapabilitiesEntry
30          MAX-ACCESS   not-accessible
31          STATUS       current
32          DESCRIPTION
33              "This table contains the basic capability information of SSs
34              that have been reported by SSs to BS using RNG-REQ, SBC-REQ
35              and REG-REQ messages. Entries in this table should be
36              created when an SS registers with a BS."
37          ::= { wman2IfBsCapabilities 1 }
38
39
40
41      wman2IfBsSsReqCapabilitiesEntry OBJECT-TYPE
42          SYNTAX      Wman2IfBsSsReqCapabilitiesEntry
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 table
48              wman2IfBsRegisteredSsTable."
49          AUGMENTS { wman2IfBsRegisteredSsEntry }
50          ::= { wman2IfBsSsReqCapabilitiesTable 1 }
51
52
53
54      Wman2IfBsSsReqCapabilitiesEntry ::= SEQUENCE {
55          wman2IfBsSsReqCapUplinkCidSupport      Wman2IfNumOfUplinkCid,
56          wman2IfBsSsReqCapArqSupport            Wman2IfArqSupportType,
57          wman2IfBsSsReqCapDsxFlowControl        Wman2IfMaxDsxFlowType,
58          wman2IfBsSsReqCapMacCrcSupport         Wman2IfMacCrcSupport,
59          wman2IfBsSsReqCapMcaFlowControl        Wman2IfMaxMcaFlowType,
60          wman2IfBsSsReqCapMcpGroupCidSupport    Wman2IfMaxMcpGroupCid,
61          wman2IfBsSsReqCapPkmFlowControl       Wman2IfMaxPkmFlowType,
62          wman2IfBsSsReqCapAuthPolicyControl    Wman2IfAuthPolicyType,
63          wman2IfBsSsReqCapMaxNumOfSupportedSA  Wman2IfMaxNumOfSaType,
64

```

```

1      wman2IfBsSsReqCapIpVersion      Wman2IfIpVersionType,
2      wman2IfBsSsReqCapMacCsSupportBitMap  Wman2IfMacCsBitMap,
3      wman2IfBsSsReqCapMaxNumOfClassifier  Wman2IfMaxClassifiers,
4      wman2IfBsSsReqCapPhsSupport      Wman2IfPhsSupportType,
5      wman2IfBsSsReqCapBandwidthAllocSupport  Wman2IfBwAllocSupport,
6      wman2IfBsSsReqCapPduConstruction  Wman2IfPduConstruction,
7      wman2IfBsSsReqCapTtgTransitionGap  Wman2IfSsTransitionGap,
8      wman2IfBsSsReqCapRtgTransitionGap  Wman2IfSsTransitionGap}
9
10
11
12  wman2IfBsSsReqCapUplinkCidSupport OBJECT-TYPE
13      SYNTAX      Wman2IfNumOfUplinkCid
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "This object shows the number of Uplink CIDs the SS can
18          support."
19      ::= { wman2IfBsSsReqCapabilitiesEntry 1 }
20
21
22
23  wman2IfBsSsReqCapArqSupport OBJECT-TYPE
24      SYNTAX      Wman2IfArqSupportType
25      MAX-ACCESS  read-only
26      STATUS      current
27      DESCRIPTION
28          "This object indicates whether the SS supports ARQ."
29      ::= { wman2IfBsSsReqCapabilitiesEntry 2 }
30
31
32
33  wman2IfBsSsReqCapDsxFowControl OBJECT-TYPE
34      SYNTAX      Wman2IfMaxDsxFowType
35      MAX-ACCESS  read-only
36      STATUS      current
37      DESCRIPTION
38          "This object specifies the maximum number of concurrent
39          DSA, DSC, or DSD transactions that SS is capable of having
40          outstanding."
41      DEFVAL      { 0 }
42      ::= { wman2IfBsSsReqCapabilitiesEntry 3 }
43
44
45
46  wman2IfBsSsReqCapMacCrcSupport OBJECT-TYPE
47      SYNTAX      Wman2IfMacCrcSupport
48      MAX-ACCESS  read-only
49      STATUS      current
50      DESCRIPTION
51          "This object indicates whether or not the SS supports MAC
52          level CRC."
53      DEFVAL      { macCrcSupport }
54      ::= { wman2IfBsSsReqCapabilitiesEntry 4 }
55
56
57
58  wman2IfBsSsReqCapMcaFlowControl OBJECT-TYPE
59      SYNTAX      Wman2IfMaxMcaFlowType
60      MAX-ACCESS  read-only
61      STATUS      current
62      DESCRIPTION
63          "This object specifies the maximum number of concurrent MCA
64          transactions that SS is capable of having outstanding."
65

```

```

1      DEFVAL      { 0 }
2      ::= { wman2IfBsSsReqCapabilitiesEntry 5 }
3
4
5  wman2IfBsSsReqCapMcpGroupCidSupport OBJECT-TYPE
6      SYNTAX      Wman2IfMaxMcpGroupCid
7      MAX-ACCESS  read-only
8      STATUS      current
9      DESCRIPTION
10         "This object indicates the maximum number of
11         simultaneous Multicast Polling Groups the SS is
12         capable of belonging to."
13
14      DEFVAL      { 0 }
15      ::= { wman2IfBsSsReqCapabilitiesEntry 6 }
16
17
18  wman2IfBsSsReqCapPkmFlowControl OBJECT-TYPE
19      SYNTAX      Wman2IfMaxPkmFlowType
20      MAX-ACCESS  read-only
21      STATUS      current
22      DESCRIPTION
23         "This object specifies the maximum number of concurrent PKM
24         transactions that SS is capable of having outstanding."
25
26      DEFVAL      { 0 }
27      ::= { wman2IfBsSsReqCapabilitiesEntry 7 }
28
29
30  wman2IfBsSsReqCapAuthPolicyControl OBJECT-TYPE
31      SYNTAX      Wman2IfAuthPolicyType
32      MAX-ACCESS  read-only
33      STATUS      current
34      DESCRIPTION
35         "This object specifies authorization policy that SS is
36         capable of. A bit value of 0 = not supported,
37         1 = supported. If this field is omitted, then both SS and
38         BS shall use the IEEE 802.16 security, constituting X.509
39         digital certificates and the RSA public key encryption
40         algorithm, as authorization policy."
41
42      ::= { wman2IfBsSsReqCapabilitiesEntry 8 }
43
44
45  wman2IfBsSsReqCapMaxNumOfSupportedSA OBJECT-TYPE
46      SYNTAX      Wman2IfMaxNumOfSaType
47      MAX-ACCESS  read-only
48      STATUS      current
49      DESCRIPTION
50         "This field specifies the maximum number of supported
51         security associations of the SS."
52
53      DEFVAL      { 1 }
54      ::= { wman2IfBsSsReqCapabilitiesEntry 9 }
55
56
57  wman2IfBsSsReqCapIpVersion OBJECT-TYPE
58      SYNTAX      Wman2IfIpVersionType
59      MAX-ACCESS  read-only
60      STATUS      current
61      DESCRIPTION
62         "This object indicates the version of IP used on the 2nd
63         Management Connection. The value should be undefined
64
65

```

```

1         if the 2nd management CID doesn't exist."
2     ::= { wman2IfBsSsReqCapabilitiesEntry 10 }
3
4
5 wman2IfBsSsReqCapMacCsSupportBitMap OBJECT-TYPE
6     SYNTAX      Wman2IfMacCsBitMap
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "This object indicates SS reported set of MAC convergence
11        sublayer support. When a bit is set, it indicates
12        the corresponding CS feature is supported."
13    ::= { wman2IfBsSsReqCapabilitiesEntry 11 }
14
15
16 wman2IfBsSsReqCapMaxNumOfClassifier OBJECT-TYPE
17     SYNTAX      Wman2IfMaxClassifiers
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21        "This object indicates the maximum number of admitted
22        Classifiers that the SS can support."
23    DEFVAL      { 0 }
24    ::= { wman2IfBsSsReqCapabilitiesEntry 12 }
25
26
27
28
29 wman2IfBsSsReqCapPhsSupport OBJECT-TYPE
30     SYNTAX      Wman2IfPhsSupportType
31     MAX-ACCESS  read-only
32     STATUS      current
33     DESCRIPTION
34        "This object indicates indicates the level of SS support
35        for PHS."
36    DEFVAL      { noPhsSupport }
37    ::= { wman2IfBsSsReqCapabilitiesEntry 13 }
38
39
40
41 wman2IfBsSsReqCapBandwidthAllocSupport OBJECT-TYPE
42     SYNTAX      Wman2IfBwAllocSupport
43     MAX-ACCESS  read-only
44     STATUS      current
45     DESCRIPTION
46        "This field indicates the bandwidth allocation
47        capabilities of the SS. The usage is defined by
48        Wman2IfBwAllocSupport."
49    ::= { wman2IfBsSsReqCapabilitiesEntry 14 }
50
51
52
53 wman2IfBsSsReqCapPduConstruction OBJECT-TYPE
54     SYNTAX      Wman2IfPduConstruction
55     MAX-ACCESS  read-only
56     STATUS      current
57     DESCRIPTION
58        "This field indicates the SS's capabilities for
59        construction and transmission of MAC PDUs. The usage
60        is defined by Wman2IfPduConstruction."
61    ::= { wman2IfBsSsReqCapabilitiesEntry 15 }
62
63
64
65 wman2IfBsSsReqCapTtgTransitionGap OBJECT-TYPE

```

```

1      SYNTAX      Wman2IfSsTransitionGap
2      UNITS       "us"
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This field indicates the SS's transition speed SSTTG
7          for TDD and H-FDD SSs. The usage is defined by
8          Wman2IfSsTransitionGap."
9      ::= { wman2IfBsSsReqCapabilitiesEntry 16 }
10
11
12
13  wman2IfBsSsReqCapRtgTransitionGap OBJECT-TYPE
14      SYNTAX      Wman2IfSsTransitionGap
15      UNITS       "us"
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "This field indicates the SS's transition speed SSRTG
20          for TDD and H-FDD SSs. The usage is defined by
21          Wman2IfSsTransitionGap."
22      ::= { wman2IfBsSsReqCapabilitiesEntry 17 }
23
24
25
26  wman2IfBsSsRspCapabilitiesTable OBJECT-TYPE
27      SYNTAX      SEQUENCE OF Wman2IfBsSsRspCapabilitiesEntry
28      MAX-ACCESS  not-accessible
29      STATUS      current
30      DESCRIPTION
31          "This table contains the basic capability information of SSs
32          that have been negotiated and agreed between BS and SS via
33          RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages.
34          This table augments the wman2IfBsRegisteredSsTable."
35      REFERENCE
36          "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
37      ::= { wman2IfBsCapabilities 2 }
38
39
40
41
42  wman2IfBsSsRspCapabilitiesEntry OBJECT-TYPE
43      SYNTAX      Wman2IfBsSsRspCapabilitiesEntry
44      MAX-ACCESS  not-accessible
45      STATUS      current
46      DESCRIPTION
47          "This table provides one row for each SS that has been
48          registered in the BS. This table augments the
49          wman2IfBsRegisteredSsTable. "
50      AUGMENTS { wman2IfBsRegisteredSsEntry }
51      ::= { wman2IfBsSsRspCapabilitiesTable 1 }
52
53
54
55  Wman2IfBsSsRspCapabilitiesEntry ::= SEQUENCE {
56      wman2IfBsSsRspCapUplinkCidSupport      Wman2IfNumOfUplinkCid,
57      wman2IfBsSsRspCapArqSupport            Wman2IfArqSupportType,
58      wman2IfBsSsRspCapDsxFlowControl        Wman2IfMaxDsxFlowType,
59      wman2IfBsSsRspCapMacCrcSupport          Wman2IfMacCrcSupport,
60      wman2IfBsSsRspCapMcaFlowControl         Wman2IfMaxMcaFlowType,
61      wman2IfBsSsRspCapMcpGroupCidSupport     Wman2IfMaxMcpGroupCid,
62      wman2IfBsSsRspCapPkmFlowControl         Wman2IfMaxPkmFlowType,
63      wman2IfBsSsRspCapAuthPolicyControl     Wman2IfAuthPolicyType,
64

```

```

1      wman2IfBsSsRspCapMaxNumOfSupportedSA      Wman2IfMaxNumOfSaType,
2      wman2IfBsSsRspCapIpVersion                Wman2IfIpVersionType,
3      wman2IfBsSsRspCapMacCsSupportBitMap        Wman2IfMacCsBitMap,
4      wman2IfBsSsRspCapMaxNumOfClassifier        Wman2IfMaxClassifiers,
5      wman2IfBsSsRspCapPhsSupport                Wman2IfPhsSupportType,
6      wman2IfBsSsRspCapBandwidthAllocSupport    Wman2IfBwAllocSupport,
7      wman2IfBsSsRspCapPduConstruction          Wman2IfPduConstruction,
8      wman2IfBsSsRspCapTtgTransitionGap          Wman2IfSsTransitionGap,
9      wman2IfBsSsRspCapRtgTransitionGap          Wman2IfSsTransitionGap}
10
11
12
13  wman2IfBsSsRspCapUplinkCidSupport OBJECT-TYPE
14      SYNTAX      Wman2IfNumOfUplinkCid
15      MAX-ACCESS  read-only
16      STATUS      current
17      DESCRIPTION
18          "Negotiated number of Uplink CIDs the SS can support."
19      ::= { wman2IfBsSsRspCapabilitiesEntry 1 }
20
21
22
23  wman2IfBsSsRspCapArqSupport OBJECT-TYPE
24      SYNTAX      Wman2IfArqSupportType
25      MAX-ACCESS  read-only
26      STATUS      current
27      DESCRIPTION
28          "This object indicates whether the SS is allowed to use ARQ
29          as a result of the capabilities negotiation."
30      ::= { wman2IfBsSsRspCapabilitiesEntry 2 }
31
32
33
34  wman2IfBsSsRspCapDsxFowControl OBJECT-TYPE
35      SYNTAX      Wman2IfMaxDsxFowType
36      MAX-ACCESS  read-only
37      STATUS      current
38      DESCRIPTION
39          "Negotiated maximum number of concurrent DSA, DSC, or DSD
40          transactions that may be outstanding."
41      ::= { wman2IfBsSsRspCapabilitiesEntry 3 }
42
43
44
45  wman2IfBsSsRspCapMacCrcSupport OBJECT-TYPE
46      SYNTAX      Wman2IfMacCrcSupport
47      MAX-ACCESS  read-only
48      STATUS      current
49      DESCRIPTION
50          "This object indicates whether or not the SS is allowed to
51          use MAC level CRC as a result of the capabilities
52          negotiation."
53      DEFVAL      { macCrcSupport }
54      ::= { wman2IfBsSsRspCapabilitiesEntry 4 }
55
56
57
58  wman2IfBsSsRspCapMcaFlowControl OBJECT-TYPE
59      SYNTAX      Wman2IfMaxMcaFlowType
60      MAX-ACCESS  read-only
61      STATUS      current
62      DESCRIPTION
63          "Negotiated maximum number of concurrent
64          MCA transactions that may be outstanding."
65

```



```

1      DEFVAL      { 0 }
2      ::= { wman2IfBsSsRspCapabilitiesEntry 5 }
3
4
5      wman2IfBsSsRspCapMcpGroupCidSupport OBJECT-TYPE
6          SYNTAX      Wman2IfMaxMcpGroupCid
7          MAX-ACCESS  read-only
8          STATUS      current
9          DESCRIPTION
10             "Negotiated maximum number of simultaneous Multicast
11              Polling Groups the SS is capable of belonging to."
12
13      DEFVAL      { 0 }
14      ::= { wman2IfBsSsRspCapabilitiesEntry 6 }
15
16
17      wman2IfBsSsRspCapPkmFlowControl OBJECT-TYPE
18          SYNTAX      Wman2IfMaxPkmFlowType
19          MAX-ACCESS  read-only
20          STATUS      current
21          DESCRIPTION
22             "Negotiated maximum number of concurrent PKM
23              transactions that may be outstanding."
24
25      DEFVAL      { 0 }
26      ::= { wman2IfBsSsRspCapabilitiesEntry 7 }
27
28
29      wman2IfBsSsRspCapAuthPolicyControl OBJECT-TYPE
30          SYNTAX      Wman2IfAuthPolicyType
31          MAX-ACCESS  read-only
32          STATUS      current
33          DESCRIPTION
34             "This object specifies negotiated authorization policy.
35              A bit value of 0 = not supported, 1 = supported. If this
36              field is omitted, then both SS and BS shall use the IEEE
37              802.16 security, constituting X.509 digital certificates
38              and the RSA public key encryption algorithm, as
39              authorization policy."
40
41      ::= { wman2IfBsSsRspCapabilitiesEntry 8 }
42
43
44      wman2IfBsSsRspCapMaxNumOfSupportedSA OBJECT-TYPE
45          SYNTAX      Wman2IfMaxNumOfSaType
46          MAX-ACCESS  read-only
47          STATUS      current
48          DESCRIPTION
49             "Negotiated maximum number of supported security
50              association of the SS."
51
52      DEFVAL      { 1 }
53      ::= { wman2IfBsSsRspCapabilitiesEntry 9 }
54
55
56      wman2IfBsSsRspCapIpVersion OBJECT-TYPE
57          SYNTAX      Wman2IfIpVersionType
58          MAX-ACCESS  read-only
59          STATUS      current
60          DESCRIPTION
61             "Negotiated version of IP used on the 2nd Management
62              Connection. The value should be undefined if the 2nd
63              management CID doesn't exist."
64
65

```

```

1      ::= { wman2IfBsSsRspCapabilitiesEntry 10 }
2
3
4  wman2IfBsSsRspCapMacCsSupportBitMap OBJECT-TYPE
5      SYNTAX      Wman2IfMacCsBitMap
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "Negotiated set of MAC convergence sublayer support.
10         When a bit is set, it indicates the corresponding CS
11         feature is supported."
12
13     ::= { wman2IfBsSsRspCapabilitiesEntry 11 }
14
15  wman2IfBsSsRspCapMaxNumOfClassifier OBJECT-TYPE
16      SYNTAX      Wman2IfMaxClassifiers
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20          "Negotiated maximum number of admitted Classifiers
21          that the SS is allowed to have."
22
23      DEFVAL      { 0 }
24
25     ::= { wman2IfBsSsRspCapabilitiesEntry 12 }
26
27  wman2IfBsSsRspCapPhsSupport OBJECT-TYPE
28      SYNTAX      Wman2IfPhsSupportType
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32          "This object indicates the negotiated level of PHS
33          support."
34
35      DEFVAL      { noPhsSupport }
36
37     ::= { wman2IfBsSsRspCapabilitiesEntry 13 }
38
39  wman2IfBsSsRspCapBandwidthAllocSupport OBJECT-TYPE
40      SYNTAX      Wman2IfBwAllocSupport
41      MAX-ACCESS  read-only
42      STATUS      current
43      DESCRIPTION
44          "This field indicates negotiated properties of the SS
45          for bandwidth allocation purposes. The usage is defined
46          by Wman2IfBwAllocSupport."
47
48     ::= { wman2IfBsSsRspCapabilitiesEntry 14 }
49
50
51  wman2IfBsSsRspCapPduConstruction OBJECT-TYPE
52      SYNTAX      Wman2IfPduConstruction
53      MAX-ACCESS  read-only
54      STATUS      current
55      DESCRIPTION
56          "Specifies negotiated capabilities for construction and
57          transmission of MAC PDUs. The usage is defined by
58          Wman2IfPduConstruction."
59
60     ::= { wman2IfBsSsRspCapabilitiesEntry 15 }
61
62
63  wman2IfBsSsRspCapTtgTransitionGap OBJECT-TYPE
64      SYNTAX      Wman2IfSsTransitionGap
65

```

```

1      UNITS          "us"
2      MAX-ACCESS    read-only
3      STATUS        current
4      DESCRIPTION
5          "This field indicates the negotiated transition speed
6          SSTTG for TDD and H-FDD SSs. The usage is defined by
7          Wman2IfSsTransitionGap."
8      ::= { wman2IfBsSsRspCapabilitiesEntry 16 }
9
10
11
12  wman2IfBsSsRspCapRtgTransitionGap OBJECT-TYPE
13      SYNTAX          Wman2IfSsTransitionGap
14      UNITS            "us"
15      MAX-ACCESS      read-only
16      STATUS          current
17      DESCRIPTION
18          "This field indicates the negotiated transition speed
19          SSRTG for TDD and H-FDD SSs. The usage is defined by
20          Wman2IfSsTransitionGap."
21      ::= { wman2IfBsSsRspCapabilitiesEntry 17 }
22
23
24
25  wman2IfBsBasicCapabilitiesTable OBJECT-TYPE
26      SYNTAX          SEQUENCE OF Wman2IfBsBasicCapabilitiesEntry
27      MAX-ACCESS      not-accessible
28      STATUS          current
29      DESCRIPTION
30          "This table contains the basic capabilities of the BS as
31          implemented in BS hardware and software. These capabilities
32          along with the configuration for them
33          (wman2IfBsCapabilitiesConfigTable) are used for negotiation
34          of basic capabilities with SS using RNG-RSP, SBC-RSP and
35          REG-RSP messages. The negotiated capabilities are obtained
36          by interSubclause of SS raw reported capabilities, BS raw
37          capabilities and BS configured capabilities. The objects in
38          the table have read-only access. The table is maintained
39          by BS."
40      ::= { wman2IfBsCapabilities 3 }
41
42
43
44
45  wman2IfBsBasicCapabilitiesEntry OBJECT-TYPE
46      SYNTAX          Wman2IfBsBasicCapabilitiesEntry
47      MAX-ACCESS      not-accessible
48      STATUS          current
49      DESCRIPTION
50          "This table provides one row for each BS sector and is
51          indexed by ifIndex."
52      INDEX { ifIndex }
53      ::= { wman2IfBsBasicCapabilitiesTable 1 }
54
55
56
57  Wman2IfBsBasicCapabilitiesEntry ::= SEQUENCE {
58      wman2IfBsCapUplinkCidSupport      Wman2IfNumOfUplinkCid,
59      wman2IfBsCapArqSupport            Wman2IfArqSupportType,
60      wman2IfBsCapDsxFLOWControl        Wman2IfMaxDsxFLOWType,
61      wman2IfBsCapMacCrcSupport         Wman2IfMacCrcSupport,
62      wman2IfBsCapMcaFLOWControl        Wman2IfMaxMcaFLOWType,
63      wman2IfBsCapMcpGroupCidSupport    Wman2IfMaxMcpGroupCid,
64

```

```

1      wman2IfBsCapPkmFlowControl      Wman2IfMaxPkmFlowType,
2      wman2IfBsCapAuthPolicyControl   Wman2IfAuthPolicyType,
3      wman2IfBsCapMaxNumOfSupportedSA Wman2IfMaxNumOfSaType,
4      wman2IfBsCapIpVersion            Wman2IfIpVersionType,
5      wman2IfBsCapMacCsSupportBitMap   Wman2IfMacCsBitMap,
6      wman2IfBsCapMaxNumOfClassifier   Wman2IfMaxClassifiers,
7      wman2IfBsCapPhsSupport           Wman2IfPhsSupportType,
8      wman2IfBsCapBandwidthAllocSupport Wman2IfBwAllocSupport,
9      wman2IfBsCapPduConstruction      Wman2IfPduConstruction,
10     wman2IfBsCapTtgTransitionGap      Wman2IfSsTransitionGap,
11     wman2IfBsCapRtgTransitionGap      Wman2IfSsTransitionGap}
12
13
14
15     wman2IfBsCapUplinkCidSupport OBJECT-TYPE
16         SYNTAX      Wman2IfNumOfUplinkCid
17         MAX-ACCESS  read-only
18         STATUS      current
19         DESCRIPTION
20             "This object shows the number of Uplink CIDs the BS can
21             support per SS."
22         ::= { wman2IfBsBasicCapabilitiesEntry 1 }
23
24
25
26     wman2IfBsCapArqSupport OBJECT-TYPE
27         SYNTAX      Wman2IfArqSupportType
28         MAX-ACCESS  read-only
29         STATUS      current
30         DESCRIPTION
31             "This object indicates whether the BS supports ARQ."
32         ::= { wman2IfBsBasicCapabilitiesEntry 2 }
33
34
35
36     wman2IfBsCapDsxFowControl OBJECT-TYPE
37         SYNTAX      Wman2IfMaxDsxFowType
38         MAX-ACCESS  read-only
39         STATUS      current
40         DESCRIPTION
41             "This object specifies the maximum number of concurrent
42             DSA, DSC, or DSD transactions that BS allows each SS to
43             have outstanding."
44         DEFVAL      { 0 }
45         ::= { wman2IfBsBasicCapabilitiesEntry 3 }
46
47
48
49     wman2IfBsCapMacCrcSupport OBJECT-TYPE
50         SYNTAX      Wman2IfMacCrcSupport
51         MAX-ACCESS  read-only
52         STATUS      current
53         DESCRIPTION
54             "This object indicates whether or not the BS supports MAC
55             level CRC."
56         DEFVAL      { macCrcSupport }
57         ::= { wman2IfBsBasicCapabilitiesEntry 4 }
58
59
60
61     wman2IfBsCapMcaFlowControl OBJECT-TYPE
62         SYNTAX      Wman2IfMaxMcaFlowType
63         MAX-ACCESS  read-only
64         STATUS      current
65

```

```

1      DESCRIPTION
2          "This object specifies the maximum number of concurrent
3          MCA transactions that BS allows each SS to have."
4      DEFVAL      { 0 }
5      ::= { wman2IfBsBasicCapabilitiesEntry 5 }
6
7
8      wman2IfBsCapMcpGroupCidSupport OBJECT-TYPE
9          SYNTAX      Wman2IfMaxMcpGroupCid
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "This object indicates the maximum number of simultaneous
14             Multicast Polling Groups the BS allows each SS to belong
15             to."
16         DEFVAL      { 0 }
17         ::= { wman2IfBsBasicCapabilitiesEntry 6 }
18
19
20
21      wman2IfBsCapPkmFlowControl OBJECT-TYPE
22          SYNTAX      Wman2IfMaxPkmFlowType
23          MAX-ACCESS  read-only
24          STATUS      current
25          DESCRIPTION
26              "This object specifies the maximum number of concurrent
27              PKM transactions that BS allows each SS to have."
28          DEFVAL      { 0 }
29          ::= { wman2IfBsBasicCapabilitiesEntry 7 }
30
31
32
33      wman2IfBsCapAuthPolicyControl OBJECT-TYPE
34          SYNTAX      Wman2IfAuthPolicyType
35          MAX-ACCESS  read-only
36          STATUS      current
37          DESCRIPTION
38              "This object specifies authorization policy that BS is
39              capable of. A bit value of 0 = not supported,
40              1 = supported. If this field is omitted, then both SS and
41              BS shall use the IEEE 802.16 security, constituting X.509
42              digital certificates and the RSA public key encryption
43              algorithm, as authorization policy."
44          ::= { wman2IfBsBasicCapabilitiesEntry 8 }
45
46
47
48
49      wman2IfBsCapMaxNumOfSupportedSA OBJECT-TYPE
50          SYNTAX      Wman2IfMaxNumOfSaType
51          MAX-ACCESS  read-only
52          STATUS      current
53          DESCRIPTION
54              "This field specifies maximum number of supported security
55              associations per SS that the BS allows."
56          DEFVAL      { 1 }
57          ::= { wman2IfBsBasicCapabilitiesEntry 9 }
58
59
60
61      wman2IfBsCapIpVersion OBJECT-TYPE
62          SYNTAX      Wman2IfIpVersionType
63          MAX-ACCESS  read-only
64          STATUS      current
65

```

```

1      DESCRIPTION
2          "This object indicates the version of IP BS allows each SS
3          to use on the 2nd Management Connection. The value
4          'undefined' should not be used for this field."
5
6      REFERENCE
7          "Subclause 11.7.4 in IEEE Std 802.16-2004"
8      ::= { wman2IfBsBasicCapabilitiesEntry 10 }
9
10
11 wman2IfBsCapMacCsSupportBitMap OBJECT-TYPE
12     SYNTAX      Wman2IfMacCsBitMap
13     MAX-ACCESS  read-only
14     STATUS      current
15     DESCRIPTION
16         "This object indicates BS set of MAC convergence
17         sublayer support. When a bit is set, it indicates
18         the corresponding CS feature is supported."
19     ::= { wman2IfBsBasicCapabilitiesEntry 11 }
20
21
22
23 wman2IfBsCapMaxNumOfClassifier OBJECT-TYPE
24     SYNTAX      Wman2IfMaxClassifiers
25     MAX-ACCESS  read-only
26     STATUS      current
27     DESCRIPTION
28         "This object indicates the maximum number of admitted
29         Classifiers per SS that the BS allows."
30     DEFVAL      { 0 }
31     ::= { wman2IfBsBasicCapabilitiesEntry 12 }
32
33
34
35 wman2IfBsCapPhsSupport OBJECT-TYPE
36     SYNTAX      Wman2IfPhsSupportType
37     MAX-ACCESS  read-only
38     STATUS      current
39     DESCRIPTION
40         "This object indicates the level of BS support for PHS.
41         The usage is defined by Wman2IfPhsSupportType."
42     DEFVAL      { noPhsSupport }
43     ::= { wman2IfBsBasicCapabilitiesEntry 13 }
44
45
46
47 wman2IfBsCapBandwidthAllocSupport OBJECT-TYPE
48     SYNTAX      Wman2IfBwAllocSupport
49     MAX-ACCESS  read-only
50     STATUS      current
51     DESCRIPTION
52         "This field indicates the bandwidth allocation properties
53         that the BS permits SSs to use. The usage is defined by
54         Wman2IfBwAllocSupport."
55     ::= { wman2IfBsBasicCapabilitiesEntry 14 }
56
57
58
59 wman2IfBsCapPduConstruction OBJECT-TYPE
60     SYNTAX      Wman2IfPduConstruction
61     MAX-ACCESS  read-only
62     STATUS      current
63     DESCRIPTION
64         "Specifies the capabilities for construction and
65

```

```

1         transmission of MAC PDUs allowed by the BS. The usage is
2         defined by Wman2IfPduConstruction."
3         ::= { wman2IfBsBasicCapabilitiesEntry 15 }
4
5
6 wman2IfBsCapTtgTransitionGap OBJECT-TYPE
7     SYNTAX      Wman2IfSsTransitionGap
8     UNITS       "us"
9     MAX-ACCESS  read-only
10    STATUS      current
11    DESCRIPTION
12        "This field indicates the transition speed SSTTG for TDD
13        and H-FDD SSs allowed by the BS. The usage is defined by
14        Wman2IfSsTransitionGap."
15        ::= { wman2IfBsBasicCapabilitiesEntry 16 }
16
17
18
19 wman2IfBsCapRtgTransitionGap OBJECT-TYPE
20     SYNTAX      Wman2IfSsTransitionGap
21     UNITS       "us"
22     MAX-ACCESS  read-only
23     STATUS      current
24     DESCRIPTION
25        "This field indicates the transition speed SSRTG for TDD
26        and H-FDD SSs allowed by the BS. The usage is defined
27        by Wman2IfSsTransitionGap."
28        ::= { wman2IfBsBasicCapabilitiesEntry 17 }
29
30
31
32 wman2IfBsCapabilitiesConfigTable OBJECT-TYPE
33     SYNTAX      SEQUENCE OF Wman2IfBsCapabilitiesConfigEntry
34     MAX-ACCESS  not-accessible
35     STATUS      current
36     DESCRIPTION
37        "This table contains the configuration for basic
38        capabilities of BS. The table is intended to be used to
39        restrict the Capabilities implemented by BS, for example in
40        order to comply with local regulatory requirements. The BS
41        should use the configuration along with the implemented
42        Capabilities (wman2IfBsBasicCapabilitiesTable) for
43        negotiation of basic capabilities with SS using RNG-RSP,
44        SBC-RSP and REG-RSP messages. The negotiated capabilities
45        are obtained by interSubclause of SS reported capabilities,
46        BS raw capabilities and BS configured capabilities. The
47        objects in the table have read-write access. The rows are
48        created by BS as a copy of wman2IfBsBasicCapabilitiesTable
49        and can be modified by NMS."
50        ::= { wman2IfBsCapabilities 4 }
51
52
53
54
55
56 wman2IfBsCapabilitiesConfigEntry OBJECT-TYPE
57     SYNTAX      Wman2IfBsCapabilitiesConfigEntry
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      ::= { wman2IfBsCapabilitiesConfigTable 1 }
2
3
4  Wman2IfBsCapabilitiesConfigEntry ::= SEQUENCE {
5      wman2IfBsCapCfgUplinkCidSupport      Wman2IfNumOfUplinkCid,
6      wman2IfBsCapCfgArqSupport            Wman2IfArqSupportType,
7      wman2IfBsCapCfgDsxFlowControl        Wman2IfMaxDsxFlowType,
8      wman2IfBsCapCfgMacCrcSupport         Wman2IfMacCrcSupport,
9      wman2IfBsCapCfgMcaFlowControl        Wman2IfMaxMcaFlowType,
10     wman2IfBsCapCfgMcpGroupCidSupport     Wman2IfMaxMcpGroupCid,
11     wman2IfBsCapCfgPkmFlowControl         Wman2IfMaxPkmFlowType,
12     wman2IfBsCapCfgAuthPolicyControl      Wman2IfAuthPolicyType,
13     wman2IfBsCapCfgMaxNumOfSupportedSA    Wman2IfMaxNumOfSaType,
14     wman2IfBsCapCfgIpVersion              Wman2IfIpVersionType,
15     wman2IfBsCapCfgMacCsSupportBitMap     Wman2IfMacCsBitMap,
16     wman2IfBsCapCfgMaxNumOfClassifier     Wman2IfMaxClassifiers,
17     wman2IfBsCapCfgPhsSupport             Wman2IfPhsSupportType,
18     wman2IfBsCapCfgBandwidthAllocSupport  Wman2IfBwAllocSupport,
19     wman2IfBsCapCfgPduConstruction        Wman2IfPduConstruction,
20     wman2IfBsCapCfgTtgTransitionGap       Wman2IfSsTransitionGap,
21     wman2IfBsCapCfgRtgTransitionGap       Wman2IfSsTransitionGap}
22
23
24
25
26  wman2IfBsCapCfgUplinkCidSupport OBJECT-TYPE
27      SYNTAX      Wman2IfNumOfUplinkCid
28      MAX-ACCESS  read-write
29      STATUS      current
30      DESCRIPTION
31          "This object shows the configured number of Uplink CIDs the
32          BS can support per SS."
33      ::= { wman2IfBsCapabilitiesConfigEntry 1 }
34
35
36
37  wman2IfBsCapCfgArqSupport OBJECT-TYPE
38      SYNTAX      Wman2IfArqSupportType
39      MAX-ACCESS  read-write
40      STATUS      current
41      DESCRIPTION
42          "This object indicates whether the BS is configured to
43          support ARQ."
44      ::= { wman2IfBsCapabilitiesConfigEntry 2 }
45
46
47
48  wman2IfBsCapCfgDsxFlowControl OBJECT-TYPE
49      SYNTAX      Wman2IfMaxDsxFlowType
50      MAX-ACCESS  read-write
51      STATUS      current
52      DESCRIPTION
53          "This object specifies the configured maximum number of
54          concurrent DSA, DSC, or DSD transactions that BS allows
55          each SS to have outstanding."
56      DEFVAL      { 0 }
57      ::= { wman2IfBsCapabilitiesConfigEntry 3 }
58
59
60
61  wman2IfBsCapCfgMacCrcSupport OBJECT-TYPE
62      SYNTAX      Wman2IfMacCrcSupport
63      MAX-ACCESS  read-write
64      STATUS      current
65

```



```

1      DESCRIPTION
2          "This object indicates whether BS is configured to support
3          MAC level CRC."
4      DEFVAL      { macCrcSupport }
5      ::= { wman2IfBsCapabilitiesConfigEntry 4 }
6
7
8      wman2IfBsCapCfgMcaFlowControl OBJECT-TYPE
9          SYNTAX      Wman2IfMaxMcaFlowType
10         MAX-ACCESS  read-write
11         STATUS      current
12         DESCRIPTION
13             "This object specifies the maximum number of concurrent
14             MCA transactions that BS is configured to allow each SS to
15             have."
16         DEFVAL      { 0 }
17         ::= { wman2IfBsCapabilitiesConfigEntry 5 }
18
19
20
21      wman2IfBsCapCfgMcpGroupCidSupport OBJECT-TYPE
22          SYNTAX      Wman2IfMaxMcpGroupCid
23          MAX-ACCESS  read-write
24          STATUS      current
25          DESCRIPTION
26              "This object indicates the maximum number of simultaneous
27              Multicast Polling Groups the BS is configured to allow
28              each SS to belong to."
29          DEFVAL      { 0 }
30          ::= { wman2IfBsCapabilitiesConfigEntry 6 }
31
32
33
34      wman2IfBsCapCfgPkmFlowControl OBJECT-TYPE
35          SYNTAX      Wman2IfMaxPkmFlowType
36          MAX-ACCESS  read-write
37          STATUS      current
38          DESCRIPTION
39              "This object specifies the maximum number of concurrent
40              PKM transactions that BS is configured to allow each SS
41              to have."
42          DEFVAL      { 0 }
43          ::= { wman2IfBsCapabilitiesConfigEntry 7 }
44
45
46
47      wman2IfBsCapCfgAuthPolicyControl OBJECT-TYPE
48          SYNTAX      Wman2IfAuthPolicyType
49          MAX-ACCESS  read-write
50          STATUS      current
51          DESCRIPTION
52              "This object specifies authorization policy that BS is
53              configured to be capable of. A bit value of 0 = not
54              supported, 1 = supported. If this field is omitted, then
55              both SS and BS shall use the IEEE 802.16 security,
56              constituting X.509 digital certificates and the RSA
57              public key encryption algorithm, as authorization policy."
58          ::= { wman2IfBsCapabilitiesConfigEntry 8 }
59
60
61
62      wman2IfBsCapCfgMaxNumOfSupportedSA OBJECT-TYPE
63          SYNTAX      Wman2IfMaxNumOfSaType
64
65

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "This field specifies configured maximum number of supported
5          security association per SS."
6      DEFVAL      { 1 }
7      ::= { wman2IfBsCapabilitiesConfigEntry 9 }
8
9
10
11  wman2IfBsCapCfgIpVersion OBJECT-TYPE
12      SYNTAX      Wman2IfIpVersionType
13      MAX-ACCESS  read-write
14      STATUS      current
15      DESCRIPTION
16          "This object indicates the configured version of IP that the
17          BS allows each SS to use on the 2nd Management Connection.
18          The value 'undefined' should not be used in this field."
19      ::= { wman2IfBsCapabilitiesConfigEntry 10 }
20
21
22
23  wman2IfBsCapCfgMacCsSupportBitMap OBJECT-TYPE
24      SYNTAX      Wman2IfMacCsBitMap
25      MAX-ACCESS  read-write
26      STATUS      current
27      DESCRIPTION
28          "This object indicates BS configured set of MAC convergence
29          sublayer support. When a bit is set, it indicates
30          the corresponding CS feature is supported."
31      ::= { wman2IfBsCapabilitiesConfigEntry 11 }
32
33
34
35  wman2IfBsCapCfgMaxNumOfClassifier OBJECT-TYPE
36      SYNTAX      Wman2IfMaxClassifiers
37      MAX-ACCESS  read-write
38      STATUS      current
39      DESCRIPTION
40          "This object indicates the configured maximum number of
41          admitted Classifiers per SS that the BS can support."
42      DEFVAL      { 0 }
43      ::= { wman2IfBsCapabilitiesConfigEntry 12 }
44
45
46
47  wman2IfBsCapCfgPhsSupport OBJECT-TYPE
48      SYNTAX      Wman2IfPhsSupportType
49      MAX-ACCESS  read-write
50      STATUS      current
51      DESCRIPTION
52          "This object indicates the configured level of BS support
53          for PHS."
54      DEFVAL      { noPhsSupport }
55      ::= { wman2IfBsCapabilitiesConfigEntry 13 }
56
57
58
59  wman2IfBsCapCfgBandwidthAllocSupport OBJECT-TYPE
60      SYNTAX      Wman2IfBwAllocSupport
61      MAX-ACCESS  read-write
62      STATUS      current
63      DESCRIPTION
64          "This field indicates configured properties of the BS for
65

```

```

1           bandwidth allocation purposes. The usage is defined by
2           Wman2IfCapBwAllocSupport."
3       ::= { wman2IfBsCapabilitiesConfigEntry 14 }
4
5
6   wman2IfBsCapCfgPduConstruction OBJECT-TYPE
7       SYNTAX      Wman2IfPduConstruction
8       MAX-ACCESS  read-write
9       STATUS      current
10      DESCRIPTION
11          "Specifies configured capabilities for construction and
12           transmission of MAC PDUs. The usage is defined by
13           Wman2IfPduConstruction."
14      ::= { wman2IfBsCapabilitiesConfigEntry 15 }
15
16
17
18   wman2IfBsCapCfgTtgTransitionGap OBJECT-TYPE
19       SYNTAX      Wman2IfSsTransitionGap
20       UNITS       "us"
21       MAX-ACCESS  read-write
22       STATUS      current
23       DESCRIPTION
24          "This field indicates the configured transition speed
25           SSTTG for TDD and H-FDD SSs. The usage is defined by
26           Wman2IfSsTransitionGap."
27      ::= { wman2IfBsCapabilitiesConfigEntry 16 }
28
29
30
31   wman2IfBsCapCfgRtgTransitionGap OBJECT-TYPE
32       SYNTAX      Wman2IfSsTransitionGap
33       UNITS       "us"
34       MAX-ACCESS  read-write
35       STATUS      current
36       DESCRIPTION
37          "This field indicates the configured transition speed
38           SSRTG for TDD and H-FDD SSs. The usage is defined by
39           Wman2IfSsTransitionGap."
40      ::= { wman2IfBsCapabilitiesConfigEntry 17 }
41
42
43
44   wman2IfBsSsActionsTable OBJECT-TYPE
45       SYNTAX      SEQUENCE OF Wman2IfBsSsActionsEntry
46       MAX-ACCESS  not-accessible
47       STATUS      current
48       DESCRIPTION
49          "This table contains all the actions specified for SSs in
50           the standard. The actions are routed down to SS using
51           unsolicited MAC messages: REG-RSP, DREG-REQ and RES-CMD.
52           The table also contains the parameters of the actions in
53           cases where they are specified by the standard."
54      ::= { wman2IfBsCps 5 }
55
56
57
58   wman2IfBsSsActionsEntry OBJECT-TYPE
59       SYNTAX      Wman2IfBsSsActionsEntry
60       MAX-ACCESS  not-accessible
61       STATUS      current
62       DESCRIPTION
63          "This table is indexed by wman2IfBsSsActionsMacAddress. The
64

```

```

1         action can be requested for SS in any state not only those
2         registered. However BS will decide whether the action is
3         applicable to the SS based on its current state and execute
4         it or skip it as defined in each action definition."
5
6     INDEX { wman2IfBsSsActionsMacAddress }
7     ::= { wman2IfBsSsActionsTable 1 }
8
9
10    Wman2IfBsSsActionsEntry ::= SEQUENCE {
11        wman2IfBsSsActionsMacAddress      MacAddress,
12        wman2IfBsSsActionsResetSs         INTEGER,
13        wman2IfBsSsActionsAbortSs         INTEGER,
14        wman2IfBsSsActionsOverrideDnFreq  Unsigned32,
15        wman2IfBsSsActionsOverrideChannelId INTEGER,
16        wman2IfBsSsActionsDeReRegSs       INTEGER,
17        wman2IfBsSsActionsDeReRegSsCode   INTEGER,
18        wman2IfBsSsActionsRowStatus       RowStatus}
19
20
21    wman2IfBsSsActionsMacAddress OBJECT-TYPE
22        SYNTAX      MacAddress
23        MAX-ACCESS   not-accessible
24        STATUS       current
25        DESCRIPTION
26            "This object uniquely identifies the SS as an action
27             target."
28        ::= { wman2IfBsSsActionsEntry 1 }
29
30
31
32    wman2IfBsSsActionsResetSs OBJECT-TYPE
33        SYNTAX      INTEGER {actionsResetSsNoAction(0),
34                             actionsResetSs(1)}
35        MAX-ACCESS   read-create
36        STATUS       current
37        DESCRIPTION
38            "This object should be implemented as follows:
39            - When set to actionsResetSs value, instructs BS to send
40              RES-CMD to SS
41            - When set to value different than actionsResetSs it
42              should be ignored
43            - When read it should return actionsResetSsNoAction
44            The RES-CMD message shall be transmitted by the BS on an
45            SS Basic CID to force the SS to reset itself,
46            reinitialize its MAC, and repeat initial system access."
47        REFERENCE
48            "Subclause 6.3.2.3.22 in IEEE Std 802.16-2004"
49        ::= { wman2IfBsSsActionsEntry 2 }
50
51
52
53
54
55    wman2IfBsSsActionsAbortSs OBJECT-TYPE
56        SYNTAX      INTEGER {actionsAbortSsNoAction(0),
57                             actionsAbortSs(1),
58                             actionAbortSsParams(2)}
59
60
61        MAX-ACCESS   read-create
62        STATUS       current
63        DESCRIPTION
64            "This object should be implemented as follows:
65

```

- When set to actionsAbortSs value, it instructs BS to send unsolicited RNG-RSP with Ranging Status equal to 'abort' without override parameters
- When set to actionAbortSsParams value, it instructs BS to send unsolicited RNG-RSP with Ranging Status equal to 'abort' and with 'Downlink Frequency Override' and 'Uplink Channel ID Override' parameters.
- When set to any other value it should be ignored
- When read it should returned actionsAbortSsNoAction"

REFERENCE

"Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
 ::= { wman2IfBsSsActionsEntry 3 }

wman2IfBsSsActionsOverrideDnFreq OBJECT-TYPE

SYNTAX Unsigned32
 UNITS "kHz"
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION

"This object is used as a parameter of the AbortSs action with the code actionAbortSsParams. It is used for licensed bands only. It defines the Center frequency, in kHz, of new downlink channel where the SS should redo initial ranging."

REFERENCE

"Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
 ::= { wman2IfBsSsActionsEntry 4 }

wman2IfBsSsActionsOverrideChannelId OBJECT-TYPE

SYNTAX INTEGER (0..199)
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION

"This object is used as a parameter of the AbortSs action with the code actionAbortSsParams. It is coded as follows:
 - Licensed bands: The identifier of the uplink channel with which the SS is to redo initial ranging (not used with PHYs without channelized uplinks).
 - License-exempt bands: The Channel Nr (see 8.5.1) where the SS should redo initial ranging."

REFERENCE

"Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
 ::= { wman2IfBsSsActionsEntry 5 }

wman2IfBsSsActionsDeReRegSs OBJECT-TYPE

SYNTAX INTEGER {actionsDeReRegSsNoAction(0),
 actionsDeReRegSs(1)}
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION

"This object should be implemented as follows:
 - When set to actionsDeReRegSs value, instructs BS to send DREG-CMD to SS with specified action code
 - When set to value different than actionsDeReRegSs it

```

1         should be ignored
2         - When read it should return actionsDeReRegSsNoAction
3         The DREG-CMD message shall be transmitted by the BS on an
4         SS Basic CID to force the SS to change its access state.
5         Upon receiving a DREG-CMD, the SS shall take the action
6         indicated by the action code defined by
7         wman2IfBsSsActionsDeReRegSsCode."
8
9     REFERENCE
10        "Subclause 6.3.2.3.26 in IEEE Std 802.16-2004"
11        ::= { wman2IfBsSsActionsEntry 6 }
12
13
14 wman2IfBsSsActionsDeReRegSsCode OBJECT-TYPE
15     SYNTAX      INTEGER {actionsDeReRegSsCodeChangeChan(0),
16                      actionsDeReRegSsCodeNoTransmit(1),
17                      actionsDeReRegSsCodeLtdTransmit(2),
18                      actionsDeReRegSsCodeResume(3)}
19
20     MAX-ACCESS   read-create
21     STATUS       current
22     DESCRIPTION
23         "This object defines the action code for
24         wman2IfBsSsActionsDeReRegSs action. The codes are defined
25         as follows:
26         actionsDeReRegSsCodeChangeChan - SS shall leave the
27         current channel and attempt to access another channel.
28         actionsDeReRegSsCodeNoTransmit - SS shall listen to the
29         current channel but shall not transmit until an
30         RES-CMD message or DREG_CMD with an Action Code that
31         allows transmission is received.
32         actionsDeReRegSsCodeLtdTransmit - SS shall listen to the
33         current channel but only transmit on the Basic,
34         Primary Management and 2nd Management Connections.
35         actionsDeReRegSsCodeResume - SS shall return to normal
36         operation and may transmit on any of its active
37         connections."
38
39     REFERENCE
40        "Subclause 6.3.2.3.26, Table 55 in IEEE Std 802.16-2004"
41        ::= { wman2IfBsSsActionsEntry 7 }
42
43
44 wman2IfBsSsActionsRowStatus OBJECT-TYPE
45     SYNTAX      RowStatus
46     MAX-ACCESS   read-create
47     STATUS       current
48     DESCRIPTION
49         "This object is used to ensure that the write operation to
50         multiple columns is guaranteed to be treated as atomic
51         operation by agent."
52         ::= { wman2IfBsSsActionsEntry 8 }
53
54
55 --
56 -- wman2IfBsMsPowerSavingStatusTable contains the power saving status
57 --
58
59 wman2IfBsMsPowerSavingStatusTable OBJECT-TYPE
60     SYNTAX      SEQUENCE OF Wman2IfBsMsPowerSavingStatusEntry
61     MAX-ACCESS   not-accessible
62
63
64
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table contains the power saving status for each CID
4          in an MS. When the BS roams to a different BS, all entries
5          associated with such MS will be deleted."
6      ::= { wman2IfBsCps 6 }
7
8
9
10     wman2IfBsMsPowerSavingStatusEntry OBJECT-TYPE
11         SYNTAX      Wman2IfBsMsPowerSavingStatusEntry
12         MAX-ACCESS  not-accessible
13         STATUS      current
14         DESCRIPTION
15             "This table provides one row for each CID in an MS, and
16             is indexed by ifIndex, wman2IfBsSsMacAddress, and
17             wman2IfBsMsCid."
18         INDEX       { ifIndex,
19                     wman2IfBsSsMacAddress,
20                     wman2IfBsMsCid }
21         ::= { wman2IfBsMsPowerSavingStatusTable 1 }
22
23
24
25     Wman2IfBsMsPowerSavingStatusEntry ::= SEQUENCE {
26         wman2IfBsMsCid                Wman2IfCidType,
27         wman2IfBsMsPowerSavingClassId Wman2IfPsClassId}
28
29
30     wman2IfBsMsCid OBJECT-TYPE
31         SYNTAX      Wman2IfCidType
32         MAX-ACCESS  read-only
33         STATUS      current
34         DESCRIPTION
35             "A 16 bit channel identifier to identify a connection."
36         ::= { wman2IfBsMsPowerSavingStatusEntry 1 }
37
38
39
40     wman2IfBsMsPowerSavingClassId OBJECT-TYPE
41         SYNTAX      Wman2IfPsClassId
42         MAX-ACCESS  read-only
43         STATUS      current
44         DESCRIPTION
45             "wman2IfBsMsPowerSavingClassId identifies the power
46             saving class associated with this CID. It maps to an
47             entry in wman2IfBsMsPowerSavingClassesTable."
48         ::= { wman2IfBsMsPowerSavingStatusEntry 2 }
49
50
51
52     --
53     -- wman2IfBsMsPowerSavingClassesTable contains the power saving classes
54     -- information
55     --
56     wman2IfBsMsPowerSavingClassesTable OBJECT-TYPE
57         SYNTAX      SEQUENCE OF Wman2IfBsMsPowerSavingClassesEntry
58         MAX-ACCESS  not-accessible
59         STATUS      current
60         DESCRIPTION
61             "This table contains the power saving classes definitions,
62             and activation / deactivation information that are provided
63             by MOB_SLP-REQ and MOB_SLP-RSP messages. When the BS roams
64             by MOB_SLP-REQ and MOB_SLP-RSP messages. When the BS roams
65             by MOB_SLP-REQ and MOB_SLP-RSP messages. When the BS roams

```

```

1         to a different BS, all entries associated with such MS will
2         be deleted."
3         ::= { wman2IfBsCps 7 }
4
5
6 wman2IfBsMsPowerSavingClassesEntry OBJECT-TYPE
7     SYNTAX      Wman2IfBsMsPowerSavingClassesEntry
8     MAX-ACCESS  not-accessible
9     STATUS      current
10    DESCRIPTION
11        "This table is indexed by ifIndex, wman2IfBsSsMacAddress,
12         and wman2IfBsMsPsClassesId. It is intended to support both
13         unicast and multicast service flows.
14         wman2IfBsSsMacAddress contains the MAC address of the MS
15         to which the power saving classes are associated."
16    INDEX { ifIndex,
17            wman2IfBsSsMacAddress,
18            wman2IfBsMsPsClassId }
19    ::= { wman2IfBsMsPowerSavingClassesTable 1 }
20
21
22
23
24 Wman2IfBsMsPowerSavingClassesEntry ::= SEQUENCE {
25     wman2IfBsMsPsClassId          Wman2IfPsClassId,
26     wman2IfBsMsStartFrameNumber   INTEGER,
27     wman2IfBsMsPowerSavingClassType Wman2PsClassType,
28     wman2IfBsMsPsClassCidDirection Wman2PsClassCidDirection,
29     wman2IfBsMsTrafficTriggeredWakening INTEGER,
30     wman2IfBsMsInitialSleepWindow  INTEGER,
31     wman2IfBsMsFinalSleepWindowBase INTEGER,
32     wman2IfBsMsFinalSleepWindowExponent INTEGER,
33     wman2IfBsMsListeningWindow     INTEGER,
34     wman2IfBsMsPowerSavingMode     Wman2IfPowerSavingMode,
35     wman2IfBsMsSlpId               INTEGER}
36
37
38
39 wman2IfBsMsPsClassId OBJECT-TYPE
40     SYNTAX      Wman2IfPsClassId
41     MAX-ACCESS  not-accessible
42     STATUS      current
43     DESCRIPTION
44        "This object uniquely identifies the power saving classes
45         in a MS."
46     ::= { wman2IfBsMsPowerSavingClassesEntry 1 }
47
48
49
50 wman2IfBsMsStartFrameNumber OBJECT-TYPE
51     SYNTAX      INTEGER
52     MAX-ACCESS  read-write
53     STATUS      current
54     DESCRIPTION
55        "Start frame number for first sleep window."
56     REFERENCE
57        "Subclause 6.3.2.3.44 in IEEE Std 802.16e-2005"
58     ::= { wman2IfBsMsPowerSavingClassesEntry 2 }
59
60
61
62 wman2IfBsMsPowerSavingClassType OBJECT-TYPE
63     SYNTAX      Wman2PsClassType
64     MAX-ACCESS  read-write
65

```



```

1      STATUS      current
2      DESCRIPTION
3          "Power saving classes type I - BE & NRT-VR,
4          Power saving classes type II - UGS & RT-VR,
5          Power saving classes type III - multicast, management CID"
6      REFERENCE
7          "Subclause 6.3.21.2-4, in IEEE Std 802.16e-2005"
8      ::= { wman2IfBsMsPowerSavingClassesEntry 3 }
9
10
11
12      wman2IfBsMsPsClassCidDirection OBJECT-TYPE
13          SYNTAX      Wman2PsClassCidDirection
14          MAX-ACCESS  read-write
15          STATUS      current
16          DESCRIPTION
17              "The direction of power saving class's CIDs."
18          REFERENCE
19              "Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
20          ::= { wman2IfBsMsPowerSavingClassesEntry 4 }
21
22
23
24      wman2IfBsMsTrafficTriggeredWakening OBJECT-TYPE
25          SYNTAX      INTEGER (0..1)
26          MAX-ACCESS  read-write
27          STATUS      current
28          DESCRIPTION
29              "0 = Power Saving Class shall not be deactivated if
30              traffic appears at the connection as per 6.3.19.2.
31              1 = Power Saving Class shall be deactivated if
32              traffic appears at the connection as 6.3.19.2."
33          REFERENCE
34              "Subclause 6.3.19.2, in IEEE Std 802.16e-2005"
35          ::= { wman2IfBsMsPowerSavingClassesEntry 5 }
36
37
38
39      wman2IfBsMsInitialSleepWindow OBJECT-TYPE
40          SYNTAX      INTEGER (0..255)
41          UNITS        "frame"
42          MAX-ACCESS  read-write
43          STATUS      current
44          DESCRIPTION
45              "The initial duration for the sleep window. It is not
46              relevant for Power Saving Class type III, and shall
47              return '0'."
48          REFERENCE
49              "Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
50          ::= { wman2IfBsMsPowerSavingClassesEntry 6 }
51
52
53
54
55      wman2IfBsMsFinalSleepWindowBase OBJECT-TYPE
56          SYNTAX      INTEGER (0..1023)
57          UNITS        "frame"
58          MAX-ACCESS  read-write
59          STATUS      current
60          DESCRIPTION
61              "The final value for the sleep interval. It is not
62              relevant for Power Saving Class type II, and shall
63              return '0'. For Power Saving Class type III, it is the
64
65

```

```

1         base for duration of single sleep window request."
2     REFERENCE
3         "Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
4     ::= { wman2IfBsMsPowerSavingClassesEntry 7 }
5
6
7     wman2IfBsMsFinalSleepWindowExponent OBJECT-TYPE
8         SYNTAX      INTEGER (0..7)
9         MAX-ACCESS   read-write
10        STATUS      current
11        DESCRIPTION
12            "The factor by which the final-sleep window base is
13             multiplied in order to calculate the final-sleep window.
14             The following formula is used:
15             final-sleep window = final-sleep window base x
16                                 2^(final-sleep window exponent)
17             For Power Saving Class type III, it is the exponent for
18             the duration of single sleep window request."
19        REFERENCE
20            "Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
21        ::= { wman2IfBsMsPowerSavingClassesEntry 8 }
22
23
24     wman2IfBsMsListeningWindow OBJECT-TYPE
25         SYNTAX      INTEGER (0..255)
26         UNITS       "frame"
27         MAX-ACCESS   read-write
28         STATUS      current
29         DESCRIPTION
30            "The Duration of MS listening window. It is not
31             relevant for Power Saving Class type III, and shall
32             return '0'."
33        REFERENCE
34            "Subclause 6.3.2.3.44, in IEEE Std 802.16e-2005"
35        ::= { wman2IfBsMsPowerSavingClassesEntry 9 }
36
37
38     wman2IfBsMsPowerSavingMode OBJECT-TYPE
39         SYNTAX      Wman2IfPowerSavingMode
40         MAX-ACCESS   read-write
41         STATUS      current
42         DESCRIPTION
43            "Indicate whether the power saving class mode of such
44             CID is active or not.
45             wman2IfBsMsPowerSavingMode = Sleep_Approved && Operation."
46        REFERENCE
47            "Subclause 6.3.2.3.45, in IEEE Std 802.16e-2005"
48        ::= { wman2IfBsMsPowerSavingClassesEntry 10 }
49
50
51     wman2IfBsMsSlpId OBJECT-TYPE
52         SYNTAX      INTEGER (0..1023)
53         MAX-ACCESS   read-only
54         STATUS      current
55         DESCRIPTION
56            "wman2IfBsMsSlpId is assigned by the BS whenever an MS is
57             instructed to enter sleep mode. This number shall be unique
58             among all MSs that are in sleep mode."
59
60
61

```

```

1      REFERENCE
2          "Subclause 6.3.2.3.45, in IEEE Std 802.16e-2005"
3      ::= { wman2IfBsMsPowerSavingClassesEntry 11 }
4
5
6      --
7      -- Mobile Station Sleep Mode Statistics Table
8      --
9      wman2IfBsMsSleepModeStatisticsTable OBJECT-TYPE
10         SYNTAX      SEQUENCE OF Wman2IfBsMsSleepModeStatisticsEntry
11         MAX-ACCESS   not-accessible
12         STATUS       current
13         DESCRIPTION
14             "This table contains the sleep mode statistic for MS. This
15              table shall be maintained as FIFO to store the sleep mode
16              statistics over a period of time that is subject to
17              implementation. This statistics information can be to
18              monitor, fine tuning, or debugging the power saving
19              performance of each MS. When the statistics entry for an
20              MS reaches the limit, it wraps around to the beginning, and
21              overwrites the oldest entry with the new entry. When the BS
22              roams to a different BS, all entries associated with such
23              MS will be deleted."
24         REFERENCE
25             "6.3.21 in IEEE Std 802.16e-2005"
26         ::= { wman2IfBsCps 8 }
27
28      wman2IfBsMsSleepModeStatisticsEntry OBJECT-TYPE
29         SYNTAX      Wman2IfBsMsSleepModeStatisticsEntry
30         MAX-ACCESS   not-accessible
31         STATUS       current
32         DESCRIPTION
33             "Each entry in the table contains the event of an MS
34              entering the sleep mode. It is indexed by ifIndex,
35              wman2IfBsSsMacAddress, and wman2IfBsMsStatisticsIndex.
36              wman2IfBsMsStatisticsIndex is the index to sleep mode event
37              entry in the table, and should be increased monotonically,
38              and wraps around when it reaches the implementation
39              specific limit. A time stamp is provided in each entry to
40              indicate when the sleep mode event took place."
41         INDEX      { ifIndex,
42                     wman2IfBsSsMacAddress,
43                     wman2IfBsMsCid,
44                     wman2IfBsMsStatisticsIndex }
45         ::= { wman2IfBsMsSleepModeStatisticsTable 1 }
46
47      Wman2IfBsMsSleepModeStatisticsEntry ::= SEQUENCE {
48          wman2IfBsMsStatisticsIndex      Unsigned32,
49          wman2IfBsMsSleepWindowStarted    Unsigned32,
50          wman2IfBsMsListeningWindowStarted Unsigned32,
51          wman2IfBsMsPendingMsdu           INTEGER,
52          wman2IfBsMsSleepWindowTimeStamp  DateAndTime}
53
54      wman2IfBsMsStatisticsIndex OBJECT-TYPE
55         SYNTAX      Unsigned32 (1 .. 4294967295)
56

```

```

1      MAX-ACCESS  read-only
2      STATUS      current
3      DESCRIPTION
4          "wman2IfBsMsStatisticsIndex identifies the entry in the
5          table where the latest sleep mode event took place."
6      ::= { wman2IfBsMsSleepModeStatisticsEntry 1 }
7
8
9
10     wman2IfBsMsSleepWindowStarted OBJECT-TYPE
11         SYNTAX      Unsigned32 (1 .. 166777215)
12         UNITS        "frame"
13         MAX-ACCESS  read-only
14         STATUS      current
15         DESCRIPTION
16             "wman2IfBsMsSleepWindowStarted identifies when the sleep
17             mode is activated.
18             wman2IfBsMsSleepWindowStarted = current frame number +
19             Start_frame_number.
20             The frame number is provided in the DL-MAP, and is
21             incremented by 1 MOD 2^24 each frame."
22         ::= { wman2IfBsMsSleepModeStatisticsEntry 2 }
23
24
25
26     wman2IfBsMsListeningWindowStarted OBJECT-TYPE
27         SYNTAX      Unsigned32 (1 .. 166777215)
28         UNITS        "frame"
29         MAX-ACCESS  read-only
30         STATUS      current
31         DESCRIPTION
32             "wman2IfBsMsListeningWindowStarted identifies when the sleep
33             mode is deactivated.
34             wman2IfBsMsListeningWindowStarted =
35             wman2IfBsMsListeningWindowStarted + sleep window
36             The frame number is provided in the DL-MAP, and is
37             incremented by 1 MOD 2^24 each frame."
38         ::= { wman2IfBsMsSleepModeStatisticsEntry 3 }
39
40
41
42
43     wman2IfBsMsPendingMsdu OBJECT-TYPE
44         SYNTAX      INTEGER
45         MAX-ACCESS  read-only
46         STATUS      current
47         DESCRIPTION
48             "Indicate the number of MAC SDU that are received from the
49             network during the sleep window."
50         ::= { wman2IfBsMsSleepModeStatisticsEntry 4 }
51
52
53
54     wman2IfBsMsSleepWindowTimeStamp OBJECT-TYPE
55         SYNTAX      DateAndTime
56         MAX-ACCESS  read-only
57         STATUS      current
58         DESCRIPTION
59             "This is the time when sleep window is started in seconds.
60             The definition of time is as in IETF RFC 868."
61         ::= { wman2IfBsMsSleepModeStatisticsEntry 5 }
62
63
64     --
65

```

```

1  -- Base station PKM group
2  -- wman2IfBsPkmObjects contain the Base Station Privacy Sublayer objects
3  --
4  wman2IfBsPkmObjects OBJECT IDENTIFIER ::= { wman2IfBsObjects 3 }
5
6
7  --
8  -- Table wman2IfBsPkmBaseTable
9  --
10
11 wman2IfBsPkmBaseTable OBJECT-TYPE
12     SYNTAX      SEQUENCE OF Wman2IfBsPkmBaseEntry
13     MAX-ACCESS  not-accessible
14     STATUS      current
15     DESCRIPTION
16         "This table describes the basic PKM attributes of each Base
17         Station wireless interface."
18     ::= { wman2IfBsPkmObjects 1 }
19
20
21 wman2IfBsPkmBaseEntry OBJECT-TYPE
22     SYNTAX      Wman2IfBsPkmBaseEntry
23     MAX-ACCESS  not-accessible
24     STATUS      current
25     DESCRIPTION
26         "Each entry contains objects describing attributes of one
27         BS wireless interface."
28     INDEX       { ifIndex }
29     ::= { wman2IfBsPkmBaseTable 1 }
30
31
32
33 Wman2IfBsPkmBaseEntry ::= SEQUENCE {
34     wman2IfBsPkmDefaultAuthLifetime      Integer32,
35     wman2IfBsPkmDefaultTekLifetime       Integer32,
36     wman2IfBsPkmDefaultSelfSigManufCertTrust INTEGER,
37     wman2IfBsPkmCheckCertValidityPeriods TruthValue,
38     wman2IfBsPkmAuthentInfos             Counter32,
39     wman2IfBsPkmAuthRequests             Counter32,
40     wman2IfBsPkmAuthReplies             Counter32,
41     wman2IfBsPkmAuthRejects             Counter32,
42     wman2IfBsPkmAuthInvalids            Counter32,
43     wman2IfBsPkmAuthGraceTime           Integer32,
44     wman2IfBsPkmTekGraceTime            Integer32,
45     wman2IfBsPkmAuthWaitTimeout         Integer32,
46     wman2IfBsPkmReauthWaitTimeout       Integer32,
47     wman2IfBsPkmOpWaitTimeout           Integer32,
48     wman2IfBsPkmRekeyWaitTimeout        Integer32,
49     wman2IfBsPkmAuthRejectWaitTimeout   Integer32 }
50
51
52
53
54
55 wman2IfBsPkmDefaultAuthLifetime OBJECT-TYPE
56     SYNTAX      Integer32 (86400..6048000)
57     UNITS       "seconds"
58     MAX-ACCESS  read-write
59     STATUS      current
60     DESCRIPTION
61         "The value of this object is the default lifetime, in
62         seconds, the BS assigns to a new authorization key."
63     REFERENCE
64
65

```

```

1           "Table 343 in IEEE Std 802.16-2004"
2   DEFVAL      { 604800 }
3   ::= { wman2IfBsPkmBaseEntry 1 }
4
5
6   wman2IfBsPkmDefaultTekLifetime OBJECT-TYPE
7       SYNTAX      Integer32 (1800..604800)
8       UNITS       "seconds"
9       MAX-ACCESS   read-write
10      STATUS      current
11      DESCRIPTION
12          "The value of this object is the default lifetime, in
13           seconds, the BS assigns to a new Traffic Encryption
14           Key (TEK). "
15      REFERENCE
16          "Table 343 in IEEE Std 802.16-2004"
17      DEFVAL      { 43200 }
18      ::= { wman2IfBsPkmBaseEntry 2 }
19
20
21
22
23
24   wman2IfBsPkmDefaultSelfSigManufCertTrust OBJECT-TYPE
25       SYNTAX      INTEGER {trusted (1),
26                       untrusted (2)}
27       MAX-ACCESS   read-write
28       STATUS      current
29       DESCRIPTION
30          "This object determines the default trust of all (new)
31           self-signed manufacturer certificates obtained after
32           setting the object."
33      ::= { wman2IfBsPkmBaseEntry 3 }
34
35
36
37   wman2IfBsPkmCheckCertValidityPeriods OBJECT-TYPE
38       SYNTAX      TruthValue
39       MAX-ACCESS   read-write
40       STATUS      current
41       DESCRIPTION
42          "Setting this object to TRUE causes all certificates
43           received thereafter to have their validity periods (and
44           their chain's validity periods) checked against the current
45           time of day. A FALSE setting will cause all certificates
46           received Thereafter to not have their validity periods
47           (nor their chain's validity periods) checked against the
48           current time of day."
49      ::= { wman2IfBsPkmBaseEntry 4 }
50
51
52
53
54   wman2IfBsPkmAuthentInfos OBJECT-TYPE
55       SYNTAX      Counter32
56       MAX-ACCESS   read-only
57       STATUS      current
58       DESCRIPTION
59          "The value of this object is the count of times the BS has
60           received an Authentication Information message from any
61           SS."
62      ::= { wman2IfBsPkmBaseEntry 5 }
63
64
65

```

```

1  wman2IfBsPkmAuthRequests OBJECT-TYPE
2      SYNTAX      Counter32
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "The value of this object is the count of times the BS has
7           received an Authorization Request message from any SS."
8      ::= { wman2IfBsPkmBaseEntry 6 }
9
10
11  wman2IfBsPkmAuthReplies OBJECT-TYPE
12      SYNTAX      Counter32
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "The value of this object is the count of times the BS has
17           transmitted an Authorization Reply message to any SS."
18      ::= { wman2IfBsPkmBaseEntry 7 }
19
20
21  wman2IfBsPkmAuthRejects OBJECT-TYPE
22      SYNTAX      Counter32
23      MAX-ACCESS  read-only
24      STATUS      current
25      DESCRIPTION
26          "The value of this object is the count of times the BS has
27           transmitted an Authorization Reject message to any SS."
28      ::= { wman2IfBsPkmBaseEntry 8 }
29
30
31  wman2IfBsPkmAuthInvalids OBJECT-TYPE
32      SYNTAX      Counter32
33      MAX-ACCESS  read-only
34      STATUS      current
35      DESCRIPTION
36          "The value of this object is the count of times the BS has
37           transmitted an Authorization Invalid message to any SS."
38      ::= { wman2IfBsPkmBaseEntry 9 }
39
40
41  wman2IfBsPkmAuthGraceTime OBJECT-TYPE
42      SYNTAX      Integer32 (300..3024000)
43      UNITS       "seconds"
44      MAX-ACCESS  read-write
45      STATUS      current
46      DESCRIPTION
47          "The value of this object is the grace time for an
48           authorization key. A SS is expected to start trying to get
49           a new authorization key beginning AuthGraceTime seconds
50           before the authorization key actually expires."
51      REFERENCE
52          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
53      DEFVAL      { 600 }
54      ::= { wman2IfBsPkmBaseEntry 10 }
55
56
57  wman2IfBsPkmTekGraceTime OBJECT-TYPE
58      SYNTAX      Integer32 (300..3024000)
59      UNITS       "seconds"
60
61
62
63
64
65

```

```

1      MAX-ACCESS    read-write
2      STATUS        current
3      DESCRIPTION
4          "The value of this object is the grace time for the TEK in
5          seconds. The SS is expected to start trying to acquire a
6          new TEK beginning TEK GraceTime seconds before the
7          expiration of the most recent TEK."
8
9      REFERENCE
10         "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
11
12      DEFVAL        { 3600 }
13      ::= { wman2IfBsPkmBaseEntry 11 }
14
15  wman2IfBsPkmAuthWaitTimeout OBJECT-TYPE
16      SYNTAX          Integer32 (2..30)
17      UNITS            "seconds"
18      MAX-ACCESS      read-write
19      STATUS           current
20      DESCRIPTION
21          "The value of this object is the Authorize Wait Timeout."
22
23      REFERENCE
24          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
25
26      DEFVAL          { 10 }
27      ::= { wman2IfBsPkmBaseEntry 12 }
28
29
30  wman2IfBsPkmReauthWaitTimeout OBJECT-TYPE
31      SYNTAX          Integer32 (2..30)
32      UNITS            "seconds"
33      MAX-ACCESS      read-write
34      STATUS           current
35      DESCRIPTION
36          "The value of this object is the Reauthorize Wait Timeout
37          in seconds."
38
39      REFERENCE
40          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
41
42      DEFVAL          { 10 }
43      ::= { wman2IfBsPkmBaseEntry 13 }
44
45
46  wman2IfBsPkmOpWaitTimeout OBJECT-TYPE
47      SYNTAX          Integer32 (1..10)
48      UNITS            "seconds"
49      MAX-ACCESS      read-write
50      STATUS           current
51      DESCRIPTION
52          "The value of this object is the Operational Wait Timeout
53          in seconds."
54
55      REFERENCE
56          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
57
58      DEFVAL          { 1 }
59      ::= { wman2IfBsPkmBaseEntry 14 }
60
61
62  wman2IfBsPkmRekeyWaitTimeout OBJECT-TYPE
63      SYNTAX          Integer32 (1..10)
64      UNITS            "seconds"
65      MAX-ACCESS      read-write

```



```

1      STATUS      current
2      DESCRIPTION
3          "The value of this object is the Rekey Wait Timeout in
4          seconds."
5      REFERENCE
6          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
7      DEFVAL      { 1 }
8      ::= { wman2IfBsPkmBaseEntry 15 }
9
10
11
12  wman2IfBsPkmAuthRejectWaitTimeout OBJECT-TYPE
13      SYNTAX      Integer32 (10..600)
14      UNITS       "seconds"
15      MAX-ACCESS  read-write
16      STATUS      current
17      DESCRIPTION
18          "The value of this object is the Authorization Reject Wait
19          Timeout in seconds."
20      REFERENCE
21          "Table 343 and subclause 11.9.19 in IEEE Std 802.16-2004"
22      DEFVAL      { 60 }
23      ::= { wman2IfBsPkmBaseEntry 16 }
24
25
26
27  --
28  -- Table wman2IfBsSsPkmAuthTable
29  --
30
31  wman2IfBsSsPkmAuthTable OBJECT-TYPE
32      SYNTAX      SEQUENCE OF Wman2IfBsSsPkmAuthEntry
33      MAX-ACCESS  not-accessible
34      STATUS      current
35      DESCRIPTION
36          "This table describes PKM attributes related
37          to the authorization for each SS. The BS maintains one
38          Primary Security Association with each Baseline
39          Privacy-enabled SS on each BS wireless interface."
40      ::= { wman2IfBsPkmObjects 2 }
41
42
43
44  wman2IfBsSsPkmAuthEntry OBJECT-TYPE
45      SYNTAX      Wman2IfBsSsPkmAuthEntry
46      MAX-ACCESS  not-accessible
47      STATUS      current
48      DESCRIPTION
49          "The BS MUST create one entry per SS per wireless
50          interface, based on the receipt of an Authorization
51          Request message and MUST not delete the entry before
52          the SS authorization permanently expires."
53      INDEX      { ifIndex, wman2IfBsSsPkmAuthMacAddress }
54      ::= { wman2IfBsSsPkmAuthTable 1 }
55
56
57
58
59  Wman2IfBsSsPkmAuthEntry ::= SEQUENCE {
60      wman2IfBsSsPkmAuthMacAddress      MacAddress,
61      wman2IfBsSsPkmAuthKeySequenceNumber Integer32,
62      wman2IfBsSsPkmAuthExpiresOld      DateAndTime,
63      wman2IfBsSsPkmAuthExpiresNew      DateAndTime,
64      wman2IfBsSsPkmAuthLifetime         Integer32,
65

```

```

1      wman2IfBsSsPkmAuthReset          INTEGER,
2      wman2IfBsSsPkmAuthInfos          Counter64,
3      wman2IfBsSsPkmAuthRequests       Counter64,
4      wman2IfBsSsPkmAuthReplies        Counter64,
5      wman2IfBsSsPkmAuthRejects        Counter64,
6      wman2IfBsSsPkmAuthInvalids       Counter64,
7      wman2IfBsSsPkmAuthRejectErrorCode INTEGER,
8      wman2IfBsSsPkmAuthRejectErrorString SnmpAdminString,
9      wman2IfBsSsPkmAuthInvalidErrorCode INTEGER,
10     wman2IfBsSsPkmAuthInvalidErrorString SnmpAdminString,
11     wman2IfBsSsPkmAuthPrimarySAId      INTEGER,
12     wman2IfBsSsPkmAuthValidStatus     INTEGER}
13
14
15
16
17 wman2IfBsSsPkmAuthMacAddress OBJECT-TYPE
18     SYNTAX      MacAddress
19     MAX-ACCESS  not-accessible
20     STATUS      current
21     DESCRIPTION
22         "The value of this object is the physical address of the SS
23         to which the authorization association applies."
24     ::= { wman2IfBsSsPkmAuthEntry 1 }
25
26
27
28 wman2IfBsSsPkmAuthKeySequenceNumber OBJECT-TYPE
29     SYNTAX      Integer32 (0..15)
30     MAX-ACCESS  read-only
31     STATUS      current
32     DESCRIPTION
33         "The value of this object is the most recent authorization
34         key sequence number for this SS."
35     ::= { wman2IfBsSsPkmAuthEntry 2 }
36
37
38
39 wman2IfBsSsPkmAuthExpiresOld OBJECT-TYPE
40     SYNTAX      DateAndTime
41     MAX-ACCESS  read-only
42     STATUS      current
43     DESCRIPTION
44         "The value of this object is the actual clock time for
45         expiration of the immediate predecessor of the most recent
46         authorization key for this FSM. If this FSM has only one
47         authorization key, then the value is the time of activation
48         of this FSM."
49     ::= { wman2IfBsSsPkmAuthEntry 3 }
50
51
52
53 wman2IfBsSsPkmAuthExpiresNew OBJECT-TYPE
54     SYNTAX      DateAndTime
55     MAX-ACCESS  read-only
56     STATUS      current
57     DESCRIPTION
58         "The value of this object is the actual clock time for
59         expiration of the most recent authorization key for this
60         FSM"
61     ::= { wman2IfBsSsPkmAuthEntry 4 }
62
63
64
65 wman2IfBsSsPkmAuthLifetime OBJECT-TYPE

```

```

1      SYNTAX      Integer32 (86400..6048000)
2      UNITS       "seconds"
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "The value of this object is the lifetime, in seconds, the
7          BS assigns to an authorization key for this SS."
8
9      REFERENCE
10         "Table 343 in IEEE Std 802.16-2004"
11
12      DEFVAL      { 604800 }
13      ::= { wman2IfBsSsPkmAuthEntry 5 }
14
15  wman2IfBsSsPkmAuthReset OBJECT-TYPE
16      SYNTAX      INTEGER {noResetRequested(1),
17                          invalidateAuth(2),
18                          sendAuthInvalid(3),
19                          invalidateTeks(4)}
20
21      MAX-ACCESS  read-write
22      STATUS      current
23      DESCRIPTION
24          "Setting this object to invalidateAuth(2) causes the BS to
25          invalidate the current SS authorization key(s), but not to
26          transmit an Authorization Invalid message nor to invalidate
27          unicast TEKs. Setting this object to sendAuthInvalid(3)
28          causes the BS to invalidate the current SS authorization
29          key(s), and to transmit an Authorization Invalid message to
30          the SS, but not to invalidate unicast TEKs. Setting this
31          object to invalidateTeks(4) causes the BS to invalidate the
32          current SS authorization key(s), to transmit an
33          Authorization Invalid message to the SS, and to
34          invalidate all unicast TEKs associated with this SS
35          authorization. Reading this object returns the
36          most-recently-set value of this object, or returns
37          noResetRequested(1) if the object has not been set since
38          the last BS reboot."
39
40      ::= { wman2IfBsSsPkmAuthEntry 6 }
41
42  wman2IfBsSsPkmAuthInfos OBJECT-TYPE
43      SYNTAX      Counter64
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47          "The value of this object is the count of times the BS has
48          received an Authentication Information message from this
49          SS."
50
51      ::= { wman2IfBsSsPkmAuthEntry 7 }
52
53  wman2IfBsSsPkmAuthRequests OBJECT-TYPE
54      SYNTAX      Counter64
55      MAX-ACCESS  read-only
56      STATUS      current
57      DESCRIPTION
58          "The value of this object is the count of times the BS has
59          received an Authorization Request message from this SS."
60
61
62
63
64
65

```

```

1      ::= { wman2IfBsSsPkmAuthEntry 8 }
2
3  wman2IfBsSsPkmAuthReplies OBJECT-TYPE
4      SYNTAX      Counter64
5      MAX-ACCESS  read-only
6      STATUS      current
7      DESCRIPTION
8          "The value of this object is the count of times the BS has
9          transmitted an Authorization Reply message to this SS."
10     ::= { wman2IfBsSsPkmAuthEntry 9 }
11
12
13
14  wman2IfBsSsPkmAuthRejects OBJECT-TYPE
15      SYNTAX      Counter64
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "The value of this object is the count of times the BS has
20          transmitted an Authorization Reject message to this SS."
21     ::= { wman2IfBsSsPkmAuthEntry 10 }
22
23
24
25  wman2IfBsSsPkmAuthInvalids OBJECT-TYPE
26      SYNTAX      Counter64
27      MAX-ACCESS  read-only
28      STATUS      current
29      DESCRIPTION
30          "The value of this object is the count of times the BS has
31          transmitted an Authorization Invalid message to this SS."
32     ::= { wman2IfBsSsPkmAuthEntry 11 }
33
34
35
36  wman2IfBsSsPkmAuthRejectErrorCode OBJECT-TYPE
37      SYNTAX      INTEGER {noInformation(0),
38                          unauthorizedSs(1),
39                          unauthorizedSaid(2),
40                          permanentAuthorizationFailure(6)}
41      MAX-ACCESS  read-only
42      STATUS      current
43      DESCRIPTION
44          "The value of this object is the enumerated description of
45          the Error-Code in most recent Authorization Reject message
46          transmitted to the SS."
47      REFERENCE
48          "IEEE Std 802.16-2004; Table 371"
49     ::= { wman2IfBsSsPkmAuthEntry 12 }
50
51
52
53  wman2IfBsSsPkmAuthRejectErrorString OBJECT-TYPE
54      SYNTAX      SnmpAdminString (SIZE (0..128))
55      MAX-ACCESS  read-only
56      STATUS      current
57      DESCRIPTION
58          "The value of this object is the Display-String in most
59          recent Authorization Reject message transmitted to the SS.
60          This is a zero length string if no Authorization Reject
61          message has been transmitted to the SS."
62     ::= { wman2IfBsSsPkmAuthEntry 13 }
63
64
65

```

```

1
2 wman2IfBsSsPkmAuthInvalidErrorCode OBJECT-TYPE
3     SYNTAX          INTEGER {noInformation(0),
4                       unauthorizedSs(1),
5                       unsolicited(3),
6                       invalidKeySequence(4),
7                       keyRequestAuthenticationFailure(5)}
8
9     MAX-ACCESS      read-only
10    STATUS           current
11    DESCRIPTION
12        "The value of this object is the enumerated description of
13        the Error-Code in most recent Authorization Invalid message
14        transmitted to the SS."
15    REFERENCE
16        "IEEE Std 802.16-2004; Table 371"
17    ::= { wman2IfBsSsPkmAuthEntry 14 }
18
19 wman2IfBsSsPkmAuthInvalidErrorString OBJECT-TYPE
20     SYNTAX          SnmpAdminString (SIZE (0..128))
21     MAX-ACCESS      read-only
22     STATUS           current
23     DESCRIPTION
24        "The value of this object is the Display-String in most
25        recent Authorization Invalid message transmitted to the SS.
26        This is a zero length string if no Authorization Invalid
27        message has been transmitted to the SS."
28    ::= { wman2IfBsSsPkmAuthEntry 15 }
29
30 wman2IfBsSsPkmAuthPrimarySAId OBJECT-TYPE
31     SYNTAX          INTEGER (0..65535)
32     MAX-ACCESS      read-only
33     STATUS           current
34     DESCRIPTION
35        "The value of this object is the Primary Security
36        Association identifier."
37    REFERENCE
38        "IEEE Std 802.16-2004; 11.9.7"
39    ::= { wman2IfBsSsPkmAuthEntry 16 }
40
41 wman2IfBsSsPkmAuthValidStatus OBJECT-TYPE
42     SYNTAX          INTEGER {unknown (0),
43                               validSsChained (1),
44                               validSsTrusted (2),
45                               invalidSsUntrusted (3),
46                               invalidCAUntrusted (4),
47                               invalidSsOther (5),
48                               invalidCAOther (6)}
49
50     MAX-ACCESS      read-only
51     STATUS           current
52     DESCRIPTION
53        "Contains the reason why a SS's certificate is deemed valid
54        or invalid. Return unknown if the SS is running PKM mode.
55        ValidSsChained means the certificate is valid because it
56        chains to a valid certificate. ValidSsTrusted means the

```

```

1         certificate is valid because it has been provisioned to be
2         trusted. InvalidSsUntrusted means the certificate is
3         invalid because it has been provisioned to be untrusted.
4         InvalidCAUntrusted means the certificate is invalid
5         because it chains to an untrusted certificate.
6         InvalidSsOther and InvalidCAOther refer to errors in
7         parsing, validity periods, etc, which are attributable to
8         the SS certificate or its chain respectively."
9
10        ::= { wman2IfBsSsPkmAuthEntry 17 }
11
12
13        --
14        -- Table wman2IfBsPkmTekTable
15        --
16        wman2IfBsPkmTekTable OBJECT-TYPE
17            SYNTAX      SEQUENCE OF Wman2IfBsPkmTekEntry
18            MAX-ACCESS   not-accessible
19            STATUS       current
20            DESCRIPTION
21                "This table describes the attributes of each Traffic
22                 Encryption Key (TEK) association. The BS maintains one TEK
23                 association per SAID on each BS wireless interface."
24            ::= { wman2IfBsPkmObjects 3 }
25
26
27
28        wman2IfBsPkmTekEntry OBJECT-TYPE
29            SYNTAX      Wman2IfBsPkmTekEntry
30            MAX-ACCESS   not-accessible
31            STATUS       current
32            DESCRIPTION
33                "Each entry contains objects describing attributes of one
34                 TEK association on a particular BS wireless interface. The
35                 BS MUST create one entry per SAID per wireless interface,
36                 based on the receipt of a Key Request message, and MUST not
37                 delete the entry before the SS authorization for the SAID
38                 permanently expires."
39            INDEX        { ifIndex, wman2IfBsPkmTekSAId }
40            ::= { wman2IfBsPkmTekTable 1 }
41
42
43
44        Wman2IfBsPkmTekEntry ::= SEQUENCE {
45            wman2IfBsPkmTekSAId          INTEGER,
46            wman2IfBsPkmTekSAType        INTEGER,
47            wman2IfBsPkmTekDataEncryptAlg Wman2IfDataEncryptAlgId,
48            wman2IfBsPkmTekDataAuthAlg   Wman2IfDataAuthAlgId,
49            wman2IfBsPkmTekEncryptAlg    Wman2IfTekEncryptAlgId,
50            wman2IfBsPkmTekLifetime      Integer32,
51            wman2IfBsPkmTekKeySequenceNumber Integer32,
52            wman2IfBsPkmTekExpiresOld    DateAndTime,
53            wman2IfBsPkmTekExpiresNew    DateAndTime,
54            wman2IfBsPkmTekReset         TruthValue,
55            wman2IfBsPkmKeyRequests      Counter32,
56            wman2IfBsPkmKeyReplies       Counter32,
57            wman2IfBsPkmKeyRejects       Counter32,
58            wman2IfBsPkmTekInvalids      Counter32,
59            wman2IfBsPkmKeyRejectErrorCode INTEGER,
60            wman2IfBsPkmKeyRejectErrorString SnmpAdminString,
61
62
63
64
65

```

```

1          wman2IfBsPkmTekInvalidErrorCode      INTEGER,
2          wman2IfBsPkmTekInvalidErrorString    SnmpAdminString}
3
4
5  wman2IfBsPkmTekSAId OBJECT-TYPE
6      SYNTAX      INTEGER (0..65535)
7      MAX-ACCESS  not-accessible
8      STATUS      current
9      DESCRIPTION
10         "The value of this object is the Security Association
11         ID (SAID)."
```

REFERENCE

```

14         "IEEE Std 802.16-2004; 11.9.7"
15         ::= { wman2IfBsPkmTekEntry 1 }
16
17
18  wman2IfBsPkmTekSAType OBJECT-TYPE
19      SYNTAX      INTEGER {primarySA(0),
20                        staticSA(1),
21                        dynamicSA(2)}
22
23      MAX-ACCESS  read-only
24      STATUS      current
25      DESCRIPTION
26         "The value of this object is the type of security
27         association. Dynamic does not apply to SSs running in PKM
28         mode."
```

REFERENCE

```

31         "IEEE Std 802.16-2004; subclause 11.9.18"
32         ::= { wman2IfBsPkmTekEntry 2 }
33
34
35  wman2IfBsPkmTekDataEncryptAlg OBJECT-TYPE
36      SYNTAX      Wman2IfDataEncryptAlgId
37      MAX-ACCESS  read-only
38      STATUS      current
39      DESCRIPTION
40         "The value of this object is the data encryption algorithm
41         being utilized."
```

REFERENCE

```

44         "Table 375, IEEE Std 802.16-2004"
45         ::= { wman2IfBsPkmTekEntry 3 }
46
47
48  wman2IfBsPkmTekDataAuthentAlg OBJECT-TYPE
49      SYNTAX      Wman2IfDataAuthAlgId
50      MAX-ACCESS  read-only
51      STATUS      current
52      DESCRIPTION
53         "The value of this object is the data authentication
54         algorithm being utilized."
```

REFERENCE

```

57         "Table 376, IEEE Std 802.16-2004"
58         ::= { wman2IfBsPkmTekEntry 4 }
59
60
61  wman2IfBsPkmTekEncryptAlg OBJECT-TYPE
62      SYNTAX      Wman2IfTekEncryptAlgId
63      MAX-ACCESS  read-only
64      STATUS      current
65
```

```

1      DESCRIPTION
2          "The value of this object is the TEK key encryption
3          algorithm being utilized."
4
5      REFERENCE
6          "Table 377, IEEE Std 802.16-2004"
7      ::= { wman2IfBsPkmTekEntry 5 }
8
9
10     wman2IfBsPkmTekLifetime OBJECT-TYPE
11         SYNTAX      Integer32 (1800..604800)
12         UNITS        "seconds"
13         MAX-ACCESS   read-only
14         STATUS       current
15         DESCRIPTION
16             "The value of this object is the lifetime, in seconds, the
17             BS assigns to keys for this TEK association."
18
19         REFERENCE
20             "Table 343 in IEEE Std 802.16-2004"
21
22         DEFVAL       { 43200 }
23         ::= { wman2IfBsPkmTekEntry 6 }
24
25     wman2IfBsPkmTekKeySequenceNumber OBJECT-TYPE
26         SYNTAX      Integer32 (0..3)
27         MAX-ACCESS   read-only
28         STATUS       current
29         DESCRIPTION
30             "The value of this object is the most recent TEK key
31             sequence number for this SAID."
32
33         REFERENCE
34             "IEEE Std 802.16-2004; subclause 11.9.5"
35         ::= { wman2IfBsPkmTekEntry 7 }
36
37
38     wman2IfBsPkmTekExpiresOld OBJECT-TYPE
39         SYNTAX      DateAndTime
40         MAX-ACCESS   read-only
41         STATUS       current
42         DESCRIPTION
43             "The value of this object is the actual clock time for
44             expiration of the immediate predecessor of the most recent
45             TEK for this FSM. If this FSM has only one TEK, then the
46             value is the time of activation of this FSM."
47
48         ::= { wman2IfBsPkmTekEntry 8 }
49
50
51     wman2IfBsPkmTekExpiresNew OBJECT-TYPE
52         SYNTAX      DateAndTime
53         MAX-ACCESS   read-only
54         STATUS       current
55         DESCRIPTION
56             "The value of this object is the actual clock time for
57             expiration of the most recent TEK for this FSM."
58
59         ::= { wman2IfBsPkmTekEntry 9 }
60
61
62     wman2IfBsPkmTekReset OBJECT-TYPE
63         SYNTAX      TruthValue
64         MAX-ACCESS   read-write
65

```



```

1      STATUS      current
2      DESCRIPTION
3          "Setting this object to TRUE causes the BS to invalidate
4          the current active TEK(s) (plural due to key transition
5          periods), and to generate a new TEK for the associated
6          SAID; the BS MAY also generate an unsolicited TEK Invalid
7          message, to optimize the TEK synchronization between the BS
8          and the SS. Reading this object always returns  FALSE."
9      ::= { wman2IfBsPkmTekEntry 10 }
10
11
12
13      wman2IfBsPkmKeyRequests OBJECT-TYPE
14          SYNTAX      Counter32
15          MAX-ACCESS  read-only
16          STATUS      current
17          DESCRIPTION
18              "The value of this object is the count of times the BS has
19              received a Key Request message."
20          ::= { wman2IfBsPkmTekEntry 11 }
21
22
23
24      wman2IfBsPkmKeyReplies OBJECT-TYPE
25          SYNTAX      Counter32
26          MAX-ACCESS  read-only
27          STATUS      current
28          DESCRIPTION
29              "The value of this object is the count of times the BS has
30              transmitted a Key Reply message."
31          ::= { wman2IfBsPkmTekEntry 12 }
32
33
34
35      wman2IfBsPkmKeyRejects OBJECT-TYPE
36          SYNTAX      Counter32
37          MAX-ACCESS  read-only
38          STATUS      current
39          DESCRIPTION
40              "The value of this object is the count of times the BS has
41              transmitted a Key Reject message."
42          ::= { wman2IfBsPkmTekEntry 13 }
43
44
45
46      wman2IfBsPkmTekInvalids OBJECT-TYPE
47          SYNTAX      Counter32
48          MAX-ACCESS  read-only
49          STATUS      current
50          DESCRIPTION
51              "The value of this object is the count of times the BS has
52              transmitted a TEK Invalid message."
53          ::= { wman2IfBsPkmTekEntry 14 }
54
55
56
57      wman2IfBsPkmKeyRejectErrorCode OBJECT-TYPE
58          SYNTAX      INTEGER {noInformation(0),
59                          unauthorizedSaid(2)}
60          MAX-ACCESS  read-only
61          STATUS      current
62          DESCRIPTION
63              "The value of this object is the enumerated; description of
64              the Error-Code in the most recent Key Reject message sent
65

```

```

1         in response to a Key Request for this SAID."
2     REFERENCE
3         "IEEE Std 802.16-2004; Table 371"
4     ::= { wman2IfBsPkmTekEntry 15 }
5
6
7 wman2IfBsPkmKeyRejectErrorString OBJECT-TYPE
8     SYNTAX      SnmpAdminString (SIZE (0..128))
9     MAX-ACCESS  read-only
10    STATUS      current
11    DESCRIPTION
12        "The value of this object is the Display-String in the most
13         recent Key Reject message sent in response to a Key Request
14         for this SAID. This is a zero length string if no Key
15         Reject message has been received since reboot."
16    ::= { wman2IfBsPkmTekEntry 16 }
17
18
19
20 wman2IfBsPkmTekInvalidErrorCode OBJECT-TYPE
21     SYNTAX      INTEGER {noInformation(0),
22                        invalidKeySequence(4)}
23     MAX-ACCESS  read-only
24     STATUS      current
25     DESCRIPTION
26         "The value of this object is the enumerated description of
27         the Error-Code in the most recent TEK Invalid message sent
28         in association with this SAID."
29     REFERENCE
30         "IEEE Std 802.16-2004; Table 371"
31     ::= { wman2IfBsPkmTekEntry 17 }
32
33
34
35
36 wman2IfBsPkmTekInvalidErrorString OBJECT-TYPE
37     SYNTAX      SnmpAdminString (SIZE (0..128))
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41         "The value of this object is the Display-String in the most
42         recent TEK Invalid message sent in association with this
43         SAID. This is a zero length string if no TEK Invalid
44         message has been received since reboot."
45     ::= { wman2IfBsPkmTekEntry 18 }
46
47
48
49 --
50 -- Base station Notification Group
51 -- wman2IfBsNotificationObjects contains the BS SNMP Trap objects
52 --
53 wman2IfBsNotification OBJECT IDENTIFIER ::= { wman2IfBsObjects 4 }
54 wman2IfBsTrapControl OBJECT IDENTIFIER ::= { wman2IfBsNotification 1
55 }
56 wman2IfBsTrapDefinitions OBJECT IDENTIFIER ::= { wman2IfBsNotification 2
57 }
58
59
60
61 -- This object groups all NOTIFICATION-TYPE objects for BS.
62 -- It is defined following RFC2758 sections 8.5 and 8.6
63 -- for the compatibility with SNMPv1.
64 wman2IfBsTrapPrefix OBJECT IDENTIFIER ::= { wman2IfBsTrapDefinitions 0 }
65

```

```

1
2 wman2IfBsTrapControlRegister OBJECT-TYPE
3     SYNTAX      BITS {wman2IfBsSsStatusNotification      (0),
4                     wman2IfBsSsDynamicServiceFail        (1),
5                     wman2IfBsSsRssiStatusChange           (2),
6                     wman2IfBsSsRegistrer                  (3),
7                     wman2IfBsSsPkmFail                     (4),
8                     wman2IfBsSsDynamicServiceFail2        (5),
9                     wman2IfBsSsRegister2Trap              (6)}
10
11     MAX-ACCESS read-write
12     STATUS      current
13     DESCRIPTION
14         "The object is used to enable or disable Base Station traps.
15         From left to right, the set bit indicates the corresponding
16         Base Station trap is enabled."
17     ::= { wman2IfBsTrapControl 1 }
18
19
20
21 wman2IfBsStatusTrapControlRegister OBJECT-TYPE
22     SYNTAX      BITS {unused(0),
23                     ssInitRangingSucc(1),
24                     ssInitRangingFail(2),
25                     ssRegistered(3),
26                     ssRegistrationFail(4),
27                     ssDeregistered(5),
28                     ssBasicCapabilitySucc(6),
29                     ssBasicCapabilityFail(7),
30                     ssAuthorizationSucc(8),
31                     ssAuthorizationFail(9),
32                     tftpSucc(10),
33                     tftpFail(11),
34                     sfCreationSucc(12),
35                     sfCreationFail(13)}
36
37     MAX-ACCESS read-write
38     STATUS      current
39     DESCRIPTION
40         "The object is used to enable or disable Base Station status
41         notification traps. The set bit indicates the corresponding
42         Base Station trap is enabled."
43     ::= { wman2IfBsTrapControl 2 }
44
45
46
47
48
49 --
50 -- BS threshold Definitions
51 --
52
53 wman2IfBsThresholdConfigTable OBJECT-TYPE
54     SYNTAX      SEQUENCE OF Wman2IfBsThresholdConfigEntry
55     MAX-ACCESS not-accessible
56     STATUS      current
57     DESCRIPTION
58         "This table contains threshold objects that can be set
59         to detect the threshold crossing events."
60     ::= { wman2IfBsTrapControl 3 }
61
62
63
64 wman2IfBsThresholdConfigEntry OBJECT-TYPE
65     SYNTAX      Wman2IfBsThresholdConfigEntry

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table provides one row for each BS sector, and is
5          indexed by ifIndex."
6      INDEX          { ifIndex }
7      ::= { wman2IfBsThresholdConfigTable 1 }
8
9
10
11  Wman2IfBsThresholdConfigEntry ::= SEQUENCE {
12      wman2IfBsRssiLowThreshold      Integer32,
13      wman2IfBsRssiHighThreshold     Integer32}
14
15  wman2IfBsRssiLowThreshold OBJECT-TYPE
16      SYNTAX      Integer32
17      UNITS        "dBm"
18      MAX-ACCESS  read-write
19      STATUS        current
20      DESCRIPTION
21          "Low threshold for generating the RSSI alarm."
22      ::= { wman2IfBsThresholdConfigEntry 1 }
23
24
25
26  wman2IfBsRssiHighThreshold OBJECT-TYPE
27      SYNTAX      Integer32
28      UNITS        "dBm"
29      MAX-ACCESS  read-write
30      STATUS        current
31      DESCRIPTION
32          "High threshold for clearing the RSSI alarm."
33      ::= { wman2IfBsThresholdConfigEntry 2 }
34
35
36
37  --
38  -- Subscriber station Notification Objects Definitions
39  --
40
41  wman2IfBsSsNotificationObjectsTable OBJECT-TYPE
42      SYNTAX      SEQUENCE OF Wman2IfBsSsNotificationObjectsEntry
43      MAX-ACCESS  not-accessible
44      STATUS        current
45      DESCRIPTION
46          "This table contains SS notification objects that have been
47          reported by the trap."
48      ::= { wman2IfBsTrapDefinitions 1 }
49
50
51
52  wman2IfBsSsNotificationObjectsEntry OBJECT-TYPE
53      SYNTAX      Wman2IfBsSsNotificationObjectsEntry
54      MAX-ACCESS  not-accessible
55      STATUS        current
56      DESCRIPTION
57          "This table provides one row for each SS that has
58          generated traps, and is double indexed by
59          wman2IfBsSsNotificationMacAddr and ifIndex for BS sector."
60      INDEX          { ifIndex, wman2IfBsSsNotificationMacAddr }
61      ::= { wman2IfBsSsNotificationObjectsTable 1 }
62
63
64  Wman2IfBsSsNotificationObjectsEntry ::= SEQUENCE {
65

```

```

1      wman2IfBsSsNotificationMacAddr      MacAddress,
2      wman2IfBsSsStatusValue              INTEGER,
3      wman2IfBsSsStatusInfo               OCTET STRING,
4      wman2IfBsDynamicServiceType         INTEGER,
5      wman2IfBsDynamicServiceFailReason   OCTET STRING,
6      wman2IfBsSsRssiStatus               INTEGER,
7      wman2IfBsSsRssiStatusInfo           OCTET STRING,
8      wman2IfBsSsRegisterStatus           INTEGER,
9      wman2IfBsDynamicServiceFailSfid     Unsigned32}
10
11
12
13  wman2IfBsSsNotificationMacAddr OBJECT-TYPE
14      SYNTAX      MacAddress
15      MAX-ACCESS  read-only
16      STATUS      current
17      DESCRIPTION
18          "The MAC address of the SS, reporing the notofiation."
19      ::= { wman2IfBsSsNotificationObjectsEntry 1 }
20
21
22
23  wman2IfBsSsStatusValue OBJECT-TYPE
24      SYNTAX      INTEGER {ssInitRangingSucc(1),
25                      ssInitRangingFail(2),
26                      ssRegistered(3),
27                      ssRegistrationFail(4),
28                      ssDeregistered(5),
29                      ssBasicCapabilitySucc(6),
30                      ssBasicCapabilityFail(7),
31                      ssAuthorizationSucc(8),
32                      ssAuthorizationFail(9),
33                      tftpSucc(10),
34                      tftpFail(11),
35                      sfCreationSucc(12),
36                      sfCreationFail(13)}
37
38      MAX-ACCESS  read-only
39      STATUS      current
40      DESCRIPTION
41          "This object indicates the status of a SS, as it goes
42          through network entry and initialization procedure."
43      ::= { wman2IfBsSsNotificationObjectsEntry 2 }
44
45
46
47
48  wman2IfBsSsStatusInfo OBJECT-TYPE
49      SYNTAX      OCTET STRING (SIZE(0..255))
50      MAX-ACCESS  read-only
51      STATUS      current
52      DESCRIPTION
53          "This object indicates the reason of SS's status change."
54      ::= { wman2IfBsSsNotificationObjectsEntry 3 }
55
56
57
58  wman2IfBsDynamicServiceType OBJECT-TYPE
59      SYNTAX      INTEGER {bsSfCreationReq(1),
60                      bsSfCreationRsp(2),
61                      bsSfCreationAck(3)}
62
63      MAX-ACCESS  read-only
64      STATUS      current
65      DESCRIPTION

```

```

1         "This object indicates the dynamic service flow
2         creation command type."
3         ::= { wman2IfBsSsNotificationObjectsEntry 4 }
4
5
6 wman2IfBsDynamicServiceFailReason OBJECT-TYPE
7     SYNTAX      OCTET STRING (SIZE(0..255))
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "This object indicates the reason why the service flow
12        creation has failed."
13        ::= { wman2IfBsSsNotificationObjectsEntry 5 }
14
15
16 wman2IfBsSsRssiStatus OBJECT-TYPE
17     SYNTAX      INTEGER {bsRssiAlarm(1),
18                          bsRssiNoAlarm(2)}
19     MAX-ACCESS  read-only
20     STATUS      current
21     DESCRIPTION
22        "A RSSI alarm is generated when RSSI becomes lower than
23        wman2IfBsLowRssiThreshold and is cleared when RSSI becomes
24        higher than wman2IfBsLowRssiThreshold."
25        ::= { wman2IfBsSsNotificationObjectsEntry 6 }
26
27
28 wman2IfBsSsRssiStatusInfo OBJECT-TYPE
29     SYNTAX      OCTET STRING (SIZE(0..255))
30     MAX-ACCESS  read-only
31     STATUS      current
32     DESCRIPTION
33        "This object indicates the reason why RSSI alarm is
34        generated."
35        ::= { wman2IfBsSsNotificationObjectsEntry 7 }
36
37
38 wman2IfBsSsRegisterStatus OBJECT-TYPE
39     SYNTAX      INTEGER {ssRegister(1),
40                          ssDeregister(2)}
41     MAX-ACCESS  read-only
42     STATUS      current
43     DESCRIPTION
44        "This object indicates the status of SS registration."
45        ::= { wman2IfBsSsNotificationObjectsEntry 8 }
46
47
48 wman2IfBsDynamicServiceFailSfid OBJECT-TYPE
49     SYNTAX      Unsigned32 (1..4294967295)
50     MAX-ACCESS  read-only
51     STATUS      current
52     DESCRIPTION
53        "This object identifies the dynamic service flow
54        for notification purposes."
55        ::= { wman2IfBsSsNotificationObjectsEntry 9 }
56
57
58 --
59 -- Subscriber station Notification Trap Definitions
60 --
61
62
63
64
65

```

```

1  wman2IfBsSsStatusNotificationTrap NOTIFICATION-TYPE
2      OBJECTS      {ifIndex,
3                    wman2IfBsSsNotificationMacAddr,
4                    wman2IfBsSsStatusValue,
5                    wman2IfBsSsStatusInfo}
6
7      STATUS      current
8
9      DESCRIPTION
10         "This trap reports the status of a SS. Based on this
11         notification the NMS will issue an alarm with certain
12         severity depending on the status and the reason received."
13         ::= { wman2IfBsTrapPrefix 1 }
14
15  wman2IfBsSsDynamicServiceFailTrap NOTIFICATION-TYPE
16      OBJECTS      {ifIndex,
17                    wman2IfBsSsNotificationMacAddr,
18                    wman2IfBsDynamicServiceType,
19                    wman2IfBsDynamicServiceFailReason}
20
21      STATUS      deprecated
22
23      DESCRIPTION
24         "Trap deprecated due to limited value without object
25         reporting SFID of victim service flow.
26         An event to report the failure of a dynamic service
27         operation happened during the dynamic services process
28         and detected in the Bs side."
29         ::= { wman2IfBsTrapPrefix 2 }
30
31
32  wman2IfBsSsRssiStatusChangeTrap NOTIFICATION-TYPE
33      OBJECTS      {ifIndex,
34                    wman2IfBsSsNotificationMacAddr,
35                    wman2IfBsSsRssiStatus,
36                    wman2IfBsSsRssiStatusInfo}
37
38      STATUS      current
39
40      DESCRIPTION
41         "An event to report that the uplink RSSI is below
42         wman2IfBsLowRssiThreshold, or above
43         wman2IfBsHighRssiThreshold after restore."
44         ::= { wman2IfBsTrapPrefix 3 }
45
46
47  wman2IfBsSsPkmFailTrap NOTIFICATION-TYPE
48      OBJECTS      {wman2IfBsSsNotificationMacAddr}
49
50      STATUS      current
51
52      DESCRIPTION
53         "An event to report the failure of a Pkm operation."
54         ::= { wman2IfBsTrapPrefix 4 }
55
56  wman2IfBsSsRegistrerTrap NOTIFICATION-TYPE
57      OBJECTS      {wman2IfBsSsNotificationMacAddr,
58                    wman2IfBsSsRegisterStatus}
59
60      STATUS      deprecated
61
62      DESCRIPTION
63         "Trap deprecated due to limited value without object ifIndex
64         reported.
65         An event to report SS registration status."
66         ::= { wman2IfBsTrapPrefix 5 }

```

```

1
2 wman2IfBsSsDynamicServiceFail2Trap NOTIFICATION-TYPE
3     OBJECTS      {ifIndex,
4                   wman2IfBsSsNotificationMacAddr,
5                   wman2IfBsDynamicServiceType,
6                   wman2IfBsDynamicServiceFailReason,
7                   wman2IfBsDynamicServiceFailSfid}
8
9     STATUS      current
10
11    DESCRIPTION
12        "An event reporting failure of DSx operation for a service
13        flow identified by wman2IfBsDynamicServiceFailSfid and
14        detected in the Bs side."
15    ::= { wman2IfBsTrapPrefix 6 }
16
17
18 wman2IfBsSsRegister2Trap NOTIFICATION-TYPE
19     OBJECTS      {ifIndex,
20                   wman2IfBsSsNotificationMacAddr,
21                   wman2IfBsSsRegisterStatus}
22
23     STATUS      current
24
25    DESCRIPTION
26        "An event to report SS registration status for a given sector
27        identified by ifIndex."
28    ::= { wman2IfBsTrapPrefix 7 }
29
30 --
31 -- Base station PHY Group
32 --
33 wman2IfBsPhy OBJECT IDENTIFIER ::= { wman2IfBsObjects 6 }
34
35 --
36 -- BS OFDM PHY objects
37 --
38
39 wman2IfBsOfdmPhy OBJECT IDENTIFIER ::= { wman2IfBsPhy 1 }
40
41
42 wman2IfBsOfdmUplinkChannelTable OBJECT-TYPE
43     SYNTAX      SEQUENCE OF Wman2IfBsOfdmUplinkChannelEntry
44     MAX-ACCESS  not-accessible
45     STATUS      current
46     DESCRIPTION
47         "This table contains UCD channel attributes, defining the
48         transmission characteristics of uplink channels"
49
50     REFERENCE
51         "Table 349 and Table 352, in IEEE Std 802.16-2004"
52     ::= { wman2IfBsOfdmPhy 1 }
53
54
55 wman2IfBsOfdmUplinkChannelEntry OBJECT-TYPE
56     SYNTAX      Wman2IfBsOfdmUplinkChannelEntry
57     MAX-ACCESS  not-accessible
58     STATUS      current
59     DESCRIPTION
60         "This table provides one row for each uplink channel of
61         multi-sector BS, and is indexed by BS ifIndex. An entry
62         in this table exists for each ifEntry of BS with an
63         ifType of propBWA2Mp."
64
65

```



```

1      INDEX { ifIndex }
2      ::= { wman2IfBsOfdmUplinkChannelTable 1 }
3
4
5      Wman2IfBsOfdmUplinkChannelEntry ::= SEQUENCE {
6          wman2IfBsOfdmCtBasedResvTimeout      INTEGER,
7          wman2IfBsOfdmBwReqOppSize            INTEGER,
8          wman2IfBsOfdmRangReqOppSize          INTEGER,
9          wman2IfBsOfdmUplinkCenterFreq        Unsigned32,
10         wman2IfBsOfdmNumSubChReqRegionFull    INTEGER,
11         wman2IfBsOfdmNumSymbolsReqRegionFull  INTEGER,
12         wman2IfBsOfdmSubChFocusCtCode        INTEGER,
13         wman2IfBsOfdmUpLinkChannelId         INTEGER}
14
15
16      wman2IfBsOfdmCtBasedResvTimeout OBJECT-TYPE
17          SYNTAX      INTEGER (1..255)
18          MAX-ACCESS  read-write
19          STATUS      current
20          DESCRIPTION
21              "The number of UL-MAPs to receive before contention-based
22              reservation is attempted again for the same connection."
23          REFERENCE
24              "Table 349, in IEEE Std 802.16-2004"
25      ::= { wman2IfBsOfdmUplinkChannelEntry 1 }
26
27
28      wman2IfBsOfdmBwReqOppSize OBJECT-TYPE
29          SYNTAX      INTEGER (1..65535)
30          UNITS       "PS"
31          MAX-ACCESS  read-write
32          STATUS      current
33          DESCRIPTION
34              "Size (in units of PS) of PHY payload that SS may use to
35              format and transmit a bandwidth request message in a
36              contention request opportunity. The value includes all
37              PHY overhead as well as allowance for the MAC data the
38              message may hold."
39          REFERENCE
40              "Table 349, in IEEE Std 802.16-2004"
41      ::= { wman2IfBsOfdmUplinkChannelEntry 2 }
42
43
44      wman2IfBsOfdmRangReqOppSize OBJECT-TYPE
45          SYNTAX      INTEGER (1..65535)
46          UNITS       "PS"
47          MAX-ACCESS  read-write
48          STATUS      current
49          DESCRIPTION
50              "Size (in units of PS) of PHY payload that SS may use to
51              format and transmit a RNG-REQ message in a contention
52              request opportunity. The value includes all PHY overhead
53              as well as allowance for the MAC data the message may
54              hold and the maximum SS/BS roundtrip propagation delay."
55          REFERENCE
56              "Table 349, in IEEE Std 802.16-2004"
57      ::= { wman2IfBsOfdmUplinkChannelEntry 3 }
58
59
60
61
62
63
64
65

```

```

1  wman2IfBsOfdmUplinkCenterFreq OBJECT-TYPE
2      SYNTAX      Unsigned32
3      UNITS       "kHz"
4      MAX-ACCESS  read-write
5      STATUS      current
6      DESCRIPTION
7          " Uplink center frequency (kHz)"
8
9      REFERENCE
10         "Table 349, in IEEE Std 802.16-2004"
11
12     ::= { wman2IfBsOfdmUplinkChannelEntry 4 }
13
14  wman2IfBsOfdmNumSubChReqRegionFull OBJECT-TYPE
15      SYNTAX      INTEGER {oneSubchannel(0),
16                          twoSubchannels(1),
17                          fourSubchannels(2),
18                          eightSubchannels(3),
19                          sixteenSubchannels(4)}
20
21      MAX-ACCESS  read-write
22      STATUS      current
23      DESCRIPTION
24          "Number of subchannels used by each transmit
25          opportunity when REQ Region-Full is allocated in
26          subchannelization region."
27
28      REFERENCE
29          "Table 352, in IEEE Std 802.16-2004"
30
31     ::= { wman2IfBsOfdmUplinkChannelEntry 5 }
32
33  wman2IfBsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
34      SYNTAX      INTEGER (0..31)
35      MAX-ACCESS  read-write
36      STATUS      current
37      DESCRIPTION
38          "Number of OFDM symbols used by each transmit
39          opportunity when REQ Region-Full is allocated in
40          subchannelization region."
41
42      REFERENCE
43          "Table 352, in IEEE Std 802.16-2004"
44
45     ::= { wman2IfBsOfdmUplinkChannelEntry 6 }
46
47  wman2IfBsOfdmSubChFocusCtCode OBJECT-TYPE
48      SYNTAX      INTEGER (0..8)
49      MAX-ACCESS  read-write
50      STATUS      current
51      DESCRIPTION
52          "Number of contention codes (CSE) that shall only be used to
53          request a subchannelized allocation. Default value 0.
54          Allowed values 0-8."
55
56      REFERENCE
57          "Table 352, in IEEE Std 802.16-2004"
58
59      DEFVAL      { 0 }
60
61     ::= { wman2IfBsOfdmUplinkChannelEntry 7 }
62
63  wman2IfBsOfdmUpLinkChannelId OBJECT-TYPE
64      SYNTAX      INTEGER (0..255)
65

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "The identifier of the uplink channel to which the relevant
5          RNG-RSP or RNG-REQ message refers."
6      REFERENCE
7          "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
8      ::= { wman2IfBsOfdmUplinkChannelEntry 8 }
9
10
11
12  wman2IfBsOfdmDownlinkChannelTable OBJECT-TYPE
13      SYNTAX      SEQUENCE OF Wman2IfBsOfdmDownlinkChannelEntry
14      MAX-ACCESS  not-accessible
15      STATUS      current
16      DESCRIPTION
17          "This table contains DCD channel attributes, defining the
18          transmission characteristics of downlink channels"
19      REFERENCE
20          "Table 358, in IEEE Std 802.16-2004"
21      ::= { wman2IfBsOfdmPhy 2 }
22
23
24
25  wman2IfBsOfdmDownlinkChannelEntry OBJECT-TYPE
26      SYNTAX      Wman2IfBsOfdmDownlinkChannelEntry
27      MAX-ACCESS  not-accessible
28      STATUS      current
29      DESCRIPTION
30          "This table provides one row for each downlink channel of
31          multi-sector BS, and is indexed by BS ifIndex. An entry
32          in this table exists for each ifEntry of BS with an
33          ifType of propBWA2Mp."
34      INDEX { ifIndex }
35      ::= { wman2IfBsOfdmDownlinkChannelTable 1 }
36
37
38
39  Wman2IfBsOfdmDownlinkChannelEntry ::= SEQUENCE {
40      wman2IfBsOfdmBsEIRP          INTEGER,
41      wman2IfBsOfdmChannelNumber   Wman2IfChannelNumber,
42      wman2IfBsOfdmTTG             INTEGER,
43      wman2IfBsOfdmRTG             INTEGER,
44      wman2IfBsOfdmInitRngMaxRSS   INTEGER,
45      wman2IfBsOfdmDownlinkCenterFreq Unsigned32,
46      wman2IfBsOfdmBsId            Wman2IfBsIdType,
47      wman2IfBsOfdmMacVersion       Wman2IfMacVersion,
48      wman2IfBsOfdmFrameDurationCode INTEGER,
49      wman2IfBsOfdmDownLinkChannelId INTEGER}
50
51
52
53  wman2IfBsOfdmBsEIRP OBJECT-TYPE
54      SYNTAX      INTEGER (-32768..32767)
55      UNITS       "dBm"
56      MAX-ACCESS  read-write
57      STATUS      current
58      DESCRIPTION
59          "The EIRP is the equivalent isotropic radiated power of
60          the base station, which is computed for a simple
61          single-antenna transmitter."
62      REFERENCE
63
64
65

```

```

1         "Table 358, in IEEE Std 802.16-2004"
2         ::= { wman2IfBsOfdmDownlinkChannelEntry 1 }
3
4
5 wman2IfBsOfdmChannelNumber OBJECT-TYPE
6     SYNTAX      Wman2IfChannelNumber
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "Downlink channel number as defined in 8.5.
11         Used for license-exempt operation only."
12
13     REFERENCE
14        "Table 358, in IEEE Std 802.16-2004"
15        ::= { wman2IfBsOfdmDownlinkChannelEntry 2 }
16
17
18 wman2IfBsOfdmTTG OBJECT-TYPE
19     SYNTAX      INTEGER (0..255)
20     MAX-ACCESS  read-write
21     STATUS      current
22     DESCRIPTION
23        "Transmit / Receive Transition Gap."
24
25     REFERENCE
26        "Table 358, in IEEE Std 802.16-2004"
27        ::= { wman2IfBsOfdmDownlinkChannelEntry 3 }
28
29
30 wman2IfBsOfdmRTG OBJECT-TYPE
31     SYNTAX      INTEGER (0..255)
32     MAX-ACCESS  read-write
33     STATUS      current
34     DESCRIPTION
35        "Receive / Transmit Transition Gap."
36
37     REFERENCE
38        "Table 358, in IEEE Std 802.16-2004"
39        ::= { wman2IfBsOfdmDownlinkChannelEntry 4 }
40
41
42 wman2IfBsOfdmInitRngMaxRSS OBJECT-TYPE
43     SYNTAX      INTEGER (-32768..32767)
44     UNITS       "dBm"
45     MAX-ACCESS  read-write
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        "Table 358, in IEEE Std 802.16-2004"
53        ::= { wman2IfBsOfdmDownlinkChannelEntry 5 }
54
55
56 wman2IfBsOfdmDownlinkCenterFreq OBJECT-TYPE
57     SYNTAX      Unsigned32
58     UNITS       "kHz"
59     MAX-ACCESS  read-write
60     STATUS      current
61     DESCRIPTION
62        "Downlink center frequency (kHz)."
63
64     REFERENCE
65

```

```

1         "Table 358, in IEEE Std 802.16-2004"
2         ::= { wman2IfBsOfdmDownlinkChannelEntry 6 }
3
4
5 wman2IfBsOfdmBsId OBJECT-TYPE
6     SYNTAX      Wman2IfBsIdType
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "Base station ID."
11
12    REFERENCE
13        "Table 358, in IEEE Std 802.16-2004"
14    ::= { wman2IfBsOfdmDownlinkChannelEntry 7 }
15
16
17 wman2IfBsOfdmMacVersion OBJECT-TYPE
18     SYNTAX      Wman2IfMacVersion
19     MAX-ACCESS  read-write
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        "Table 358, in IEEE Std 802.16-2004"
27    ::= { wman2IfBsOfdmDownlinkChannelEntry 8 }
28
29
30 wman2IfBsOfdmFrameDurationCode OBJECT-TYPE
31     SYNTAX      INTEGER {duration2dot5ms(0),
32                          duration4ms(1),
33                          duration5ms(2),
34                          duration8ms(3),
35                          duration10ms(4),
36                          duration12dot5ms(5),
37                          duration20ms(6)}
38
39     MAX-ACCESS  read-write
40     STATUS      current
41     DESCRIPTION
42        "The duration of the frame. The frame duration code
43         values are specified in Table 230."
44
45    REFERENCE
46        "Subclause 11.4.1, Table 230, in IEEE Std 802.16-2004"
47    ::= { wman2IfBsOfdmDownlinkChannelEntry 9 }
48
49
50 wman2IfBsOfdmDownLinkChannelId OBJECT-TYPE
51     SYNTAX      INTEGER (0..255)
52     MAX-ACCESS  read-write
53     STATUS      current
54     DESCRIPTION
55        "The identifier of the downlink channel to which this
56         message refers."
57
58    REFERENCE
59        "Subclause 6.3.2.3.1. Table 15, in IEEE Std 802.16-2004"
60    ::= { wman2IfBsOfdmDownlinkChannelEntry 10 }
61
62
63 wman2IfBsOfdmUcdBurstProfileTable OBJECT-TYPE
64     SYNTAX      SEQUENCE OF Wman2IfBsOfdmUcdBurstProfileEntry
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table contains UCD burst profiles for each uplink
5          channel"
6      REFERENCE
7          "Table 356, in IEEE Std 802.16-2004"
8      ::= { wman2IfBsOfdmPhy 3 }
9
10
11
12  wman2IfBsOfdmUcdBurstProfileEntry OBJECT-TYPE
13      SYNTAX          Wman2IfBsOfdmUcdBurstProfileEntry
14      MAX-ACCESS      not-accessible
15      STATUS          current
16      DESCRIPTION
17          "This table provides one row for each UCD burst profile.
18          This table is double indexed. The primary index is an
19          ifIndex with an ifType of propBWA2Mp. The secondary index
20          is wman2IfBsOfdmUiucIndex."
21      INDEX { ifIndex, wman2IfBsOfdmUiucIndex }
22      ::= { wman2IfBsOfdmUcdBurstProfileTable 1 }
23
24
25
26  Wman2IfBsOfdmUcdBurstProfileEntry ::= SEQUENCE {
27      wman2IfBsOfdmUiucIndex          INTEGER,
28      wman2IfBsOfdmUcdFecCodeType     Wman2IfOfdmFecCodeType,
29      wman2IfBsOfdmFocusCtPowerBoost  INTEGER,
30      wman2IfBsOfdmUcdTcsEnable       INTEGER,
31      wman2IfBsOfdmUcdBurstProfileRowStatus  RowStatus}
32
33
34
35  wman2IfBsOfdmUiucIndex OBJECT-TYPE
36      SYNTAX          INTEGER (5 .. 12)
37      MAX-ACCESS      not-accessible
38      STATUS          current
39      DESCRIPTION
40          "The Uplink Interval Usage Code indicates the uplink burst
41          profile in the UCD message, and is used along with ifIndex
42          to identify an entry in the
43          wman2IfBsOfdmUcdBurstProfileTable."
44      REFERENCE
45          "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
46      ::= { wman2IfBsOfdmUcdBurstProfileEntry 1 }
47
48
49
50  wman2IfBsOfdmUcdFecCodeType OBJECT-TYPE
51      SYNTAX          Wman2IfOfdmFecCodeType
52      MAX-ACCESS      read-create
53      STATUS          current
54      DESCRIPTION
55          "Uplink FEC code type and modulation type"
56      REFERENCE
57          "Table 356, in IEEE Std 802.16-2004"
58      ::= { wman2IfBsOfdmUcdBurstProfileEntry 2 }
59
60
61
62  wman2IfBsOfdmFocusCtPowerBoost OBJECT-TYPE
63      SYNTAX          INTEGER (0 .. 255)
64      MAX-ACCESS      read-create
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "The power boost in dB of focused contention carriers, as
4              described in 8.3.6.3.3."
5      REFERENCE
6          "Table 356, in IEEE Std 802.16-2004"
7      ::= { wman2IfBsOfdmUcdBurstProfileEntry 3 }
8
9
10     wman2IfBsOfdmUcdTcsEnable OBJECT-TYPE
11         SYNTAX      INTEGER {tcsDisabled(0),
12             tcsEnabled(1)}
13         MAX-ACCESS   read-create
14         STATUS       current
15         DESCRIPTION
16             "This parameter determines the transmission convergence
17                 sublayer, as described in 8.1.4.3, can be enabled on a
18                 per-burst basis for both uplink and downlink. through
19                 DIUC/UIUC messages."
20         REFERENCE
21             "Table 356, in IEEE Std 802.16-2004"
22         ::= { wman2IfBsOfdmUcdBurstProfileEntry 4 }
23
24
25     wman2IfBsOfdmUcdBurstProfileRowStatus OBJECT-TYPE
26         SYNTAX      RowStatus
27         MAX-ACCESS   read-create
28         STATUS       current
29         DESCRIPTION
30             "This object is used to create a new row or modify or
31                 delete an existing row in this table.
32
33             If the implementator of this MIB has choosen not
34             to implement 'dynamic assignment' of profiles, this
35             object is not useful and should return noSuchName
36             upon SNMP request."
37         ::= { wman2IfBsOfdmUcdBurstProfileEntry 5 }
38
39
40     wman2IfBsOfdmDcdBurstProfileTable OBJECT-TYPE
41         SYNTAX      SEQUENCE OF Wman2IfBsOfdmDcdBurstProfileEntry
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
48                 index is wman2IfBsOfdmDiucIndex."
49         REFERENCE
50             "Table 362, in IEEE Std 802.16-2004"
51         ::= { wman2IfBsOfdmPhy 4 }
52
53
54     wman2IfBsOfdmDcdBurstProfileEntry OBJECT-TYPE
55         SYNTAX      Wman2IfBsOfdmDcdBurstProfileEntry
56         MAX-ACCESS   not-accessible
57         STATUS       current
58         DESCRIPTION

```

```

1         "This table provides one row for each DCD burst profile.
2         This table is double indexed. The primary index is an
3         ifIndex with an ifType of propBWA2Mp. The secondary index
4         is wman2IfBsOfdmDiucIndex."
5
6         INDEX { ifIndex, wman2IfBsOfdmDiucIndex }
7         ::= { wman2IfBsOfdmDcdBurstProfileTable 1 }
8
9
10        Wman2IfBsOfdmDcdBurstProfileEntry ::= SEQUENCE {
11            wman2IfBsOfdmDiucIndex          INTEGER,
12            wman2IfBsOfdmDownlinkFrequency  Unsigned32,
13            wman2IfBsOfdmDcdFecCodeType     Wman2IfOfdmFecCodeType,
14            wman2IfBsOfdmDiucMandatoryExitThresh  INTEGER,
15            wman2IfBsOfdmDiucMinEntryThresh  INTEGER,
16            wman2IfBsOfdmTcsEnable          INTEGER,
17            wman2IfBsOfdmDcdBurstProfileRowStatus  RowStatus}
18
19
20        wman2IfBsOfdmDiucIndex OBJECT-TYPE
21            SYNTAX      INTEGER (1..11)
22            MAX-ACCESS  not-accessible
23            STATUS      current
24            DESCRIPTION
25                "The Downlink Interval Usage Code indicates the downlink
26                burst profile in the DCD message, and is used along with
27                ifIndex to identify an entry in the
28                wman2IfBsOfdmDcdBurstProfileTable."
29            REFERENCE
30                "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
31            ::= { wman2IfBsOfdmDcdBurstProfileEntry 1 }
32
33
34        wman2IfBsOfdmDownlinkFrequency OBJECT-TYPE
35            SYNTAX      Unsigned32
36            UNITS       "kHz"
37            MAX-ACCESS  read-create
38            STATUS      current
39            DESCRIPTION
40                "Downlink Frequency (kHz)."

```



```

1      DESCRIPTION
2          "DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or
3          below where this DIUC can no longer be used and where this
4          change to a more robust DIUC is required in 0.25 dB units."
5
6      REFERENCE
7          "Table 362, in IEEE Std 802.16-2004"
8
9      ::= { wman2IfBsOfdmDcdBurstProfileEntry 4 }
10
11  wman2IfBsOfdmDiucMinEntryThresh OBJECT-TYPE
12      SYNTAX      INTEGER (0..255)
13      MAX-ACCESS  read-create
14      STATUS      current
15      DESCRIPTION
16          "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
17          required to start using this DIUC when changing from a more
18          robust DIUC is required, in 0.25 dB units."
19
20      REFERENCE
21          "Table 362, in IEEE Std 802.16-2004"
22
23      ::= { wman2IfBsOfdmDcdBurstProfileEntry 5 }
24
25  wman2IfBsOfdmTcsEnable OBJECT-TYPE
26      SYNTAX      INTEGER {tcsDisabled (0),
27                      tcsEnabled (1)}
28      MAX-ACCESS  read-create
29      STATUS      current
30      DESCRIPTION
31          "Indicates whether Transmission CONvergence Sublayer
32          is enabled or disabled."
33
34      REFERENCE
35          "Table 362, in IEEE Std 802.16-2004"
36
37      ::= { wman2IfBsOfdmDcdBurstProfileEntry 6 }
38
39  wman2IfBsOfdmDcdBurstProfileRowStatus OBJECT-TYPE
40      SYNTAX      RowStatus
41      MAX-ACCESS  read-create
42      STATUS      current
43      DESCRIPTION
44          "This object is used to create a new row or modify or
45          delete an existing row in this table.
46
47          If the implementator of this MIB has choosen not
48          to implement 'dynamic assignment' of profiles, this
49          object is not useful and should return noSuchName
50          upon SNMP request."
51
52      ::= { wman2IfBsOfdmDcdBurstProfileEntry 7 }
53
54
55  wman2IfBsOfdmConfigurationTable OBJECT-TYPE
56      SYNTAX      SEQUENCE OF Wman2IfBsOfdmConfigurationEntry
57      MAX-ACCESS  not-accessible
58      STATUS      current
59      DESCRIPTION
60          "This table contains BS configuration objects, specific to
61          OFDM PHY."
62
63      ::= { wman2IfBsOfdmPhy 5 }
64
65

```

```

1
2 wman2IfBsOfdmConfigurationEntry OBJECT-TYPE
3     SYNTAX      Wman2IfBsOfdmConfigurationEntry
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "This table is indexed by ifIndex with an ifType of
8         propBWA2Mp."
9     INDEX { ifIndex }
10    ::= { wman2IfBsOfdmConfigurationTable 1 }
11
12
13
14 Wman2IfBsOfdmConfigurationEntry ::= SEQUENCE {
15     wman2IfBsOfdmMinReqRegionFullTxOpp    INTEGER,
16     wman2IfBsOfdmMinFocusedCtTxOpp        INTEGER,
17     wman2IfBsOfdmMaxRoundTripDelay        INTEGER,
18     wman2IfBsOfdmRangeAbortTimingThold    INTEGER,
19     wman2IfBsOfdmRangeAbortPowerThold     INTEGER,
20     wman2IfBsOfdmRangeAbortFreqThold     INTEGER,
21     wman2IfBsOfdmDnlkRateId              INTEGER,
22     wman2IfBsOfdmRatioG                  INTEGER}
23
24
25
26 wman2IfBsOfdmMinReqRegionFullTxOpp OBJECT-TYPE
27     SYNTAX      INTEGER (1..65535)
28     UNITS       "1/sec"
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32         "The minimum number of Full bandwidth Req-Region Full
33         Transmit opportunities scheduled in the UL per second."
34     REFERENCE
35         "Subclause 6.3.7.4.3 in IEEE Std 802.16-2004"
36     ::= { wman2IfBsOfdmConfigurationEntry 1 }
37
38
39
40
41 wman2IfBsOfdmMinFocusedCtTxOpp OBJECT-TYPE
42     SYNTAX      INTEGER (0..65535)
43     UNITS       "1/sec"
44     MAX-ACCESS  read-write
45     STATUS      current
46     DESCRIPTION
47         "The minimum number of focused contention Transmit
48         opportunities scheduled in the UL per second. The value may
49         be 0 if the focused contention is not implemented."
50     REFERENCE
51         "Subclauses 6.3.6.4 and 8.3.7.3.3 in IEEE Std 802.16-2004"
52     ::= { wman2IfBsOfdmConfigurationEntry 2 }
53
54
55
56 wman2IfBsOfdmMaxRoundTripDelay OBJECT-TYPE
57     SYNTAX      INTEGER (1..65535)
58     UNITS       "us"
59     MAX-ACCESS  read-write
60     STATUS      current
61     DESCRIPTION
62         "Maximum supported round trip delay.
63         It is required to limit the cell size."
64
65

```

```

1      REFERENCE
2          "Subclause 8.3.5.1 in IEEE Std 802.16-2004"
3      ::= { wman2IfBsOfdmConfigurationEntry 3 }
4
5
6  wman2IfBsOfdmRangeAbortTimingThold OBJECT-TYPE
7      SYNTAX      INTEGER (0..255)
8      UNITS       "1/Fs"
9      MAX-ACCESS  read-write
10     STATUS      current
11     DESCRIPTION
12         "This object defines Tolerable Timing Offset. BS performs
13         Initial Ranging until the SS transmissions are within
14         limits that are deemed tolerable by the BS. If the SS does
15         not transmit within these limits after a number of
16         correction attempts then the BS aborts Initial Ranging."
17     REFERENCE
18         "Figure 63 and Table 365 in IEEE Std 802.16-2004"
19     ::= { wman2IfBsOfdmConfigurationEntry 4 }
20
21
22  wman2IfBsOfdmRangeAbortPowerThold OBJECT-TYPE
23     SYNTAX      INTEGER (0..255)
24     UNITS       "0.25dB"
25     MAX-ACCESS  read-write
26     STATUS      current
27     DESCRIPTION
28         "This object defines Tolerable Power Offset. BS performs
29         Initial Ranging until the SS transmissions are within
30         limits that are deemed tolerable by the BS. If the SS does
31         not transmit within these limits after a number of
32         correction attempts then the BS aborts Initial Ranging."
33     REFERENCE
34         "Figure 63 and Table 365 in IEEE Std 802.16-2004"
35     ::= { wman2IfBsOfdmConfigurationEntry 5 }
36
37
38  wman2IfBsOfdmRangeAbortFreqThold OBJECT-TYPE
39     SYNTAX      INTEGER (0..255)
40     UNITS       "Hz"
41     MAX-ACCESS  read-write
42     STATUS      current
43     DESCRIPTION
44         "This object defines Tolerable Frequency Offset. BS performs
45         Initial Ranging until the SS transmissions are within
46         limits that are deemed tolerable by the BS. If the SS does
47         not transmit within these limits after a number of
48         correction attempts then the BS aborts Initial Ranging."
49     REFERENCE
50         "Figure 63 and Table 365 in IEEE Std 802.16-2004"
51     ::= { wman2IfBsOfdmConfigurationEntry 6 }
52
53
54  wman2IfBsOfdmDnlkRateId OBJECT-TYPE
55     SYNTAX      INTEGER { dnlkRateIdBpsk1Over2(0),
56                          dnlkRateIdQpsk1Over2(1),
57                          dnlkRateIdQpsk3Over4(2),
58                          dnlkRateId16Qam1Over2(3),
59

```

```

1          dnlkRateId16Qam3Over4(4),
2          dnlkRateId64Qam2Over3(5),
3          dnlkRateId64Qam3Over4(6)}
4
5      MAX-ACCESS    read-write
6      STATUS        current
7      DESCRIPTION
8          "The Rate ID to be used in the first downlink burst
9          immediately following the FCH. The Rate ID encoding is
10         static and cannot be changed during system operation. The
11         change of the Rate ID should be applied on system
12         re-intialisation (e.g. following sector or BS reset)."

```

```

1  Wman2IfBsSsOfdmReqCapabilitiesEntry ::= SEQUENCE {
2      wman2IfBsSsOfdmReqCapFftSizes          Wman2IfOfdmFftSizes,
3      wman2IfBsSsOfdmReqCapSsDemodulator      Wman2IfOfdmSsDeModType,
4      wman2IfBsSsOfdmReqCapSsModulator        Wman2IfOfdmSsModType,
5      wman2IfBsSsOfdmReqCapFocusedCtSupport    Wman2IfOfdmFocusedCt,
6      wman2IfBsSsOfdmReqCapTcSublayerSupport  Wman2IfOfdmTcSublayer}
7
8
9
10 wman2IfBsSsOfdmReqCapFftSizes OBJECT-TYPE
11     SYNTAX      Wman2IfOfdmFftSizes
12     MAX-ACCESS  read-only
13     STATUS      current
14     DESCRIPTION
15         "This field indicates the FFT sizes supported by SS.
16         The usage is defined by Wman2IfOfdmFftSizes."
17     ::= { wman2IfBsSsOfdmReqCapabilitiesEntry 1 }
18
19
20 wman2IfBsSsOfdmReqCapSsDemodulator OBJECT-TYPE
21     SYNTAX      Wman2IfOfdmSsDeModType
22     MAX-ACCESS  read-only
23     STATUS      current
24     DESCRIPTION
25         "This field indicates the different demodulator options
26         supported by SS for downlink.
27         The usage is defined by Wman2IfOfdmSsDeModType."
28     ::= { wman2IfBsSsOfdmReqCapabilitiesEntry 2 }
29
30
31
32 wman2IfBsSsOfdmReqCapSsModulator OBJECT-TYPE
33     SYNTAX      Wman2IfOfdmSsModType
34     MAX-ACCESS  read-only
35     STATUS      current
36     DESCRIPTION
37         "This field indicates the different modulator options
38         supported by SS for uplink.
39         The usage is defined by Wman2IfOfdmSsModType."
40     ::= { wman2IfBsSsOfdmReqCapabilitiesEntry 3 }
41
42
43
44 wman2IfBsSsOfdmReqCapFocusedCtSupport OBJECT-TYPE
45     SYNTAX      Wman2IfOfdmFocusedCt
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49         "This field indicates whether the SS supports Focused
50         Contention. The usage is defined by
51         Wman2IfOfdmFocusedCt."
52     ::= { wman2IfBsSsOfdmReqCapabilitiesEntry 4 }
53
54
55
56 wman2IfBsSsOfdmReqCapTcSublayerSupport OBJECT-TYPE
57     SYNTAX      Wman2IfOfdmTcSublayer
58     MAX-ACCESS  read-only
59     STATUS      current
60     DESCRIPTION
61         "This field indicates whether or not the SS supports
62         the TC sublayer. The usage is defined by
63         Wman2IfOfdmTcSublayer."
64
65

```

```

1      ::= { wman2IfBsSsOfdmReqCapabilitiesEntry 5 }
2
3  wman2IfBsSsOfdmRspCapabilitiesTable OBJECT-TYPE
4      SYNTAX      SEQUENCE OF Wman2IfBsSsOfdmRspCapabilitiesEntry
5      MAX-ACCESS  not-accessible
6      STATUS      current
7      DESCRIPTION
8          "This table contains the basic capability information,
9           specific to OFDM Phy, of SSs that have been negotiated
10          and agreed between BS and SS via RNG-REQ/RSP,
11          SBC-REQ/RSP and REG-REQ/RSP messages. This table
12          augments the wman2IfBsRegisteredSsTable."
13      REFERENCE
14          "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
15      ::= { wman2IfBsOfdmPhy 7 }
16
17  wman2IfBsSsOfdmRspCapabilitiesEntry OBJECT-TYPE
18      SYNTAX      Wman2IfBsSsOfdmRspCapabilitiesEntry
19      MAX-ACCESS  not-accessible
20      STATUS      current
21      DESCRIPTION
22          "This table provides one row for each SS that has been
23          registered in the BS. This table augments the
24          wman2IfBsRegisteredSsTable. "
25      AUGMENTS { wman2IfBsRegisteredSsEntry }
26      ::= { wman2IfBsSsOfdmRspCapabilitiesTable 1 }
27
28  Wman2IfBsSsOfdmRspCapabilitiesEntry ::= SEQUENCE {
29      wman2IfBsSsOfdmRspCapFftSizes      Wman2IfOfdmFftSizes,
30      wman2IfBsSsOfdmRspCapSsDemodulator  Wman2IfOfdmSsDeModType,
31      wman2IfBsSsOfdmRspCapSsModulator    Wman2IfOfdmSsModType,
32      wman2IfBsSsOfdmRspCapFocusedCtSupport Wman2IfOfdmFocusedCt,
33      wman2IfBsSsOfdmRspCapTcSublayerSupport Wman2IfOfdmTcSublayer}
34
35  wman2IfBsSsOfdmRspCapFftSizes OBJECT-TYPE
36      SYNTAX      Wman2IfOfdmFftSizes
37      MAX-ACCESS  read-only
38      STATUS      current
39      DESCRIPTION
40          "This field indicates the FFT sizes negotiated with the
41          SS. The usage is defined by Wman2IfOfdmFftSizes."
42      ::= { wman2IfBsSsOfdmRspCapabilitiesEntry 1 }
43
44  wman2IfBsSsOfdmRspCapSsDemodulator OBJECT-TYPE
45      SYNTAX      Wman2IfOfdmSsDeModType
46      MAX-ACCESS  read-only
47      STATUS      current
48      DESCRIPTION
49          "This field indicates the different demodulator options
50          negotiated for SS for downlink. The usage is defined by
51          Wman2IfOfdmSsDeModType."
52      ::= { wman2IfBsSsOfdmRspCapabilitiesEntry 2 }
53
54  wman2IfBsSsOfdmRspCapSsModulator OBJECT-TYPE

```

```

1      SYNTAX      Wman2IfOfdmSsModType
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "This field indicates the different modulator options
6          negotiated for SS for uplink. The usage is defined by
7          Wman2IfOfdmSsModType."
8      ::= { wman2IfBsSsOfdmRspCapabilitiesEntry 3 }
9
10
11
12  wman2IfBsSsOfdmRspCapFocusedCtSupport OBJECT-TYPE
13      SYNTAX      Wman2IfOfdmFocusedCt
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "This field indicates whether the SS has negotiated the
18          support for Focused Contention. The usage is defined by
19          Wman2IfOfdmFocusedCt."
20      ::= { wman2IfBsSsOfdmRspCapabilitiesEntry 4 }
21
22
23
24  wman2IfBsSsOfdmRspCapTcSublayerSupport OBJECT-TYPE
25      SYNTAX      Wman2IfOfdmTcSublayer
26      MAX-ACCESS  read-only
27      STATUS      current
28      DESCRIPTION
29          "This field indicates whether the SS has negotiated
30          support for the TC sublayer. The usage is defined by
31          Wman2IfOfdmTcSublayer."
32      ::= { wman2IfBsSsOfdmRspCapabilitiesEntry 5 }
33
34
35
36  wman2IfBsOfdmCapabilitiesTable OBJECT-TYPE
37      SYNTAX      SEQUENCE OF Wman2IfBsOfdmCapabilitiesEntry
38      MAX-ACCESS  not-accessible
39      STATUS      current
40      DESCRIPTION
41          "This table contains the basic capabilities, specific to
42          OFDM Phy, of the BS as implemented in BS hardware and
43          software. These capabilities along with the configuration
44          for them (wman2IfBsOfdmCapabilitiesConfigTable) are used
45          for negotiation of basic capabilities with SS using
46          RNG-RSP, SBC-RSP and REG-RSP messages. The negotiated
47          capabilities are obtained by interSubclause of SS raw
48          reported capabilities, BS raw capabilities and BS
49          configured capabilities. The objects in the table have
50          read-only access. The table is maintained by BS."
51      ::= { wman2IfBsOfdmPhy 8 }
52
53
54
55
56  wman2IfBsOfdmCapabilitiesEntry OBJECT-TYPE
57      SYNTAX      Wman2IfBsOfdmCapabilitiesEntry
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      ::= { wman2IfBsOfdmCapabilitiesTable 1 }
2
3  Wman2IfBsOfdmCapabilitiesEntry ::= SEQUENCE {
4      wman2IfBsOfdmCapFftSizes          Wman2IfOfdmFftSizes,
5      wman2IfBsOfdmCapSsDemodulator      Wman2IfOfdmSsDeModType,
6      wman2IfBsOfdmCapSsModulator        Wman2IfOfdmSsModType,
7      wman2IfBsOfdmCapFocusedCtSupport    Wman2IfOfdmFocusedCt,
8      wman2IfBsOfdmCapTcSublayerSupport   Wman2IfOfdmTcSublayer}
9
10
11
12  wman2IfBsOfdmCapFftSizes OBJECT-TYPE
13      SYNTAX          Wman2IfOfdmFftSizes
14      MAX-ACCESS      read-only
15      STATUS          current
16      DESCRIPTION
17          "This field indicates the FFT sizes supported by the BS.
18           The usage is defined by Wman2IfOfdmCapFftSizes."
19      ::= { wman2IfBsOfdmCapabilitiesEntry 1 }
20
21
22
23  wman2IfBsOfdmCapSsDemodulator OBJECT-TYPE
24      SYNTAX          Wman2IfOfdmSsDeModType
25      MAX-ACCESS      read-only
26      STATUS          current
27      DESCRIPTION
28          "This field indicates the different BS demodulator options
29           for uplink supported by the BS. The usage is defined by
30           Wman2IfOfdmSsDeModType."
31      ::= { wman2IfBsOfdmCapabilitiesEntry 2 }
32
33
34
35  wman2IfBsOfdmCapSsModulator OBJECT-TYPE
36      SYNTAX          Wman2IfOfdmSsModType
37      MAX-ACCESS      read-only
38      STATUS          current
39      DESCRIPTION
40          "This field indicates the different BS modulator options
41           for downlink supported by the BS. The usage is defined by
42           Wman2IfOfdmSsModType."
43      ::= { wman2IfBsOfdmCapabilitiesEntry 3 }
44
45
46
47  wman2IfBsOfdmCapFocusedCtSupport OBJECT-TYPE
48      SYNTAX          Wman2IfOfdmFocusedCt
49      MAX-ACCESS      read-only
50      STATUS          current
51      DESCRIPTION
52          "This field indicates the BS support for Focused
53           Contention. The usage is defined by
54           Wman2IfOfdmFocusedCt."
55      ::= { wman2IfBsOfdmCapabilitiesEntry 4 }
56
57
58
59  wman2IfBsOfdmCapTcSublayerSupport OBJECT-TYPE
60      SYNTAX          Wman2IfOfdmTcSublayer
61      MAX-ACCESS      read-only
62      STATUS          current
63      DESCRIPTION
64          "This field indicates the BS supports for TC sublayer. The
65

```



```

1         usage is defined by Wman2IfOfdmTcSublayer."
2     ::= { wman2IfBsOfdmCapabilitiesEntry 5 }
3
4
5 wman2IfBsOfdmCapabilitiesConfigTable OBJECT-TYPE
6     SYNTAX      SEQUENCE OF Wman2IfBsOfdmCapabilitiesConfigEntry
7     MAX-ACCESS  not-accessible
8     STATUS      current
9     DESCRIPTION
10        "This table contains the configuration for basic
11        capabilities of BS, specific to OFDM Phy. The table is
12        intended to be used to restrict the Capabilities
13        implemented by BS, for example in order to comply with
14        local regulatory requirements. The BS should use the
15        configuration along with the implemented Capabilities
16        (wman2IfBsOfdmPhyTable) for negotiation of basic
17        capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP
18        messages. The negotiated capabilities are obtained by
19        interSubclause of SS reported capabilities, BS raw
20        capabilities and BS configured capabilities. The objects
21        in the table have read-write access. The rows are created
22        by BS as a copy of wman2IfBsBasicCapabilitiesTable
23        and can be modified by NMS."
24    ::= { wman2IfBsOfdmPhy 9 }
25
26
27
28
29
30 wman2IfBsOfdmCapabilitiesConfigEntry OBJECT-TYPE
31     SYNTAX      Wman2IfBsOfdmCapabilitiesConfigEntry
32     MAX-ACCESS  not-accessible
33     STATUS      current
34     DESCRIPTION
35        "This table provides one row for each BS sector and is
36        indexed by ifIndex."
37     INDEX { ifIndex }
38     ::= { wman2IfBsOfdmCapabilitiesConfigTable 1 }
39
40
41
42 Wman2IfBsOfdmCapabilitiesConfigEntry ::= SEQUENCE {
43     wman2IfBsOfdmCapCfgFftSizes          Wman2IfOfdmFftSizes,
44     wman2IfBsOfdmCapCfgSsDemodulator      Wman2IfOfdmSsDeModType,
45     wman2IfBsOfdmCapCfgSsModulator        Wman2IfOfdmSsModType,
46     wman2IfBsOfdmCapCfgFocusedCtSupport   Wman2IfOfdmFocusedCt,
47     wman2IfBsOfdmCapCfgTcSublayerSupport  Wman2IfOfdmTcSublayer}
48
49
50 wman2IfBsOfdmCapCfgFftSizes OBJECT-TYPE
51     SYNTAX      Wman2IfOfdmFftSizes
52     MAX-ACCESS  read-write
53     STATUS      current
54     DESCRIPTION
55        "This field indicates the FFT sizes support configured for
56        the BS. The usage is defined by
57        Wman2IfOfdmCapFftSizes."
58     ::= { wman2IfBsOfdmCapabilitiesConfigEntry 1 }
59
60
61
62 wman2IfBsOfdmCapCfgSsDemodulator OBJECT-TYPE
63     SYNTAX      Wman2IfOfdmSsDeModType
64     MAX-ACCESS  read-write
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This field indicates the different BS demodulator options
4          configured for uplink. The usage is defined by
5          Wman2IfOfdmSsDeModType."
6      ::= { wman2IfBsOfdmCapabilitiesConfigEntry 2 }
7
8
9
10     wman2IfBsOfdmCapCfgSsModulator OBJECT-TYPE
11         SYNTAX      Wman2IfOfdmSsModType
12         MAX-ACCESS  read-write
13         STATUS      current
14         DESCRIPTION
15             "This field indicates the different BS modulator options
16             configured for downlink. The usage is defined by
17             Wman2IfOfdmSsModType."
18         ::= { wman2IfBsOfdmCapabilitiesConfigEntry 3 }
19
20
21
22     wman2IfBsOfdmCapCfgFocusedCtSupport OBJECT-TYPE
23         SYNTAX      Wman2IfOfdmFocusedCt
24         MAX-ACCESS  read-write
25         STATUS      current
26         DESCRIPTION
27             "This field indicates the BS support configured for
28             Focused Contention. The usage is defined by
29             Wman2IfOfdmFocusedCt."
30         ::= { wman2IfBsOfdmCapabilitiesConfigEntry 4 }
31
32
33
34     wman2IfBsOfdmCapCfgTcSublayerSupport OBJECT-TYPE
35         SYNTAX      Wman2IfOfdmTcSublayer
36         MAX-ACCESS  read-write
37         STATUS      current
38         DESCRIPTION
39             "This field indicates the BS support configured for TC
40             sublayer. The usage is defined by
41             Wman2IfOfdmTcSublayer."
42         ::= { wman2IfBsOfdmCapabilitiesConfigEntry 5 }
43
44
45
46     --
47     -- BS OFDMA PHY objects
48     --
49     wman2IfBsOfdmaPhy OBJECT IDENTIFIER ::= { wman2IfBsPhy 2 }
50
51
52     wman2IfBsOfdmaUplinkChannelTable OBJECT-TYPE
53         SYNTAX      SEQUENCE OF Wman2IfBsOfdmaUplinkChannelEntry
54         MAX-ACCESS  not-accessible
55         STATUS      current
56         DESCRIPTION
57             "This table contains UCD channel attributes, defining the
58             transmission characteristics of uplink channels"
59         REFERENCE
60             "Table 349 and Table 353, in IEEE Std 802.16-2004"
61         ::= { wman2IfBsOfdmaPhy 1 }
62
63
64     wman2IfBsOfdmaUplinkChannelEntry OBJECT-TYPE
65

```

```

1      SYNTAX      Wman2IfBsOfdmaUplinkChannelEntry
2      MAX-ACCESS  not-accessible
3      STATUS      current
4      DESCRIPTION
5
6          "This table provides one row for each uplink channel of
7          multi-sector BS, and is indexed by BS ifIndex. An entry
8          in this table exists for each ifEntry of BS with an
9          ifType of propBWA2Mp."
10     INDEX      { ifIndex }
11     ::= { wman2IfBsOfdmaUplinkChannelTable 1 }
12
13
14     Wman2IfBsOfdmaUplinkChannelEntry ::= SEQUENCE {
15         wman2IfBsOfdmaCtBasedResvTimeout      INTEGER,
16         wman2IfBsOfdmaBwReqOppSize            INTEGER,
17         wman2IfBsOfdmaRangReqOppSize          INTEGER,
18         wman2IfBsOfdmaUplinkCenterFreq        Unsigned32,
19         wman2IfBsOfdmaInitRngCodes            INTEGER,
20         wman2IfBsOfdmaPeriodicRngCodes        INTEGER,
21         wman2IfBsOfdmaBWRngCodes             INTEGER,
22         wman2IfBsOfdmaPerRngBackoffStart      INTEGER,
23         wman2IfBsOfdmaPerRngBackoffEnd        INTEGER,
24         wman2IfBsOfdmaStartOfRngCodes         INTEGER,
25         wman2IfBsOfdmaPermutationBase         INTEGER,
26         wman2IfBsOfdmaULAllocSubchBitmap      OCTET STRING,
27         wman2IfBsOfdmaOptPermULAllocSubchBitmap OCTET STRING,
28         wman2IfBsOfdmaBandAMCAallocThreshold  INTEGER,
29         wman2IfBsOfdmaBandAMCReleaseThreshold INTEGER,
30         wman2IfBsOfdmaBandAMCAallocTimer      INTEGER,
31         wman2IfBsOfdmaBandAMCReleaseTimer     INTEGER,
32         wman2IfBsOfdmaBandStatRepMAXPeriod    INTEGER,
33         wman2IfBsOfdmaBandAMCRetryTimer       INTEGER,
34         wman2IfBsOfdmaSafetyChAllocThreshold  INTEGER,
35         wman2IfBsOfdmaSafetyChReleaseThreshold INTEGER,
36         wman2IfBsOfdmaSafetyChAllocTimer      INTEGER,
37         wman2IfBsOfdmaSafetyChReleaseTimer    INTEGER,
38         wman2IfBsOfdmaBinStatRepMAXPeriod     INTEGER,
39         wman2IfBsOfdmaSafetyChARetryTimer     INTEGER,
40         wman2IfBsOfdmaHARQAackDelayULBurst    INTEGER,
41         wman2IfBsOfdmaCQICHBandAMCTranaDelay  INTEGER }
42
43
44     wman2IfBsOfdmaCtBasedResvTimeout OBJECT-TYPE
45     SYNTAX      INTEGER (1..255)
46     MAX-ACCESS  read-write
47     STATUS      current
48     DESCRIPTION
49
50         "The number of UL-MAPs to receive before contention-based
51         reservation is attempted again for the same connection."
52     REFERENCE
53
54         "Table 349, in IEEE Std 802.16-2004"
55     ::= { wman2IfBsOfdmaUplinkChannelEntry 1 }
56
57
58     wman2IfBsOfdmaBwReqOppSize OBJECT-TYPE
59     SYNTAX      INTEGER (1..65535)
60     UNITS       "PS"

```

```

1      MAX-ACCESS    read-write
2      STATUS        current
3      DESCRIPTION
4          "Size (in units of PS) of PHY payload that SS may use to
5          format and transmit a bandwidth request message in a
6          contention request opportunity. The value includes all
7          PHY overhead as well as allowance for the MAC data the
8          message may hold."
9
10     REFERENCE
11         "Table 349, in IEEE Std 802.16-2004"
12     ::= { wman2IfBsOfdmaUplinkChannelEntry 2 }
13
14
15 wman2IfBsOfdmaRangReqOppSize OBJECT-TYPE
16     SYNTAX      INTEGER (1..65535)
17     UNITS        "PS"
18     MAX-ACCESS  read-write
19     STATUS      current
20     DESCRIPTION
21         "Size (in units of PS) of PHY payload that SS may use to
22         format and transmit a RNG-REQ message in a contention
23         request opportunity. The value includes all PHY overhead
24         as well as allowance for the MAC data the message may
25         hold and the maximum SS/BS roundtrip propagation delay."
26     REFERENCE
27         "Table 349, in IEEE Std 802.16-2004"
28     ::= { wman2IfBsOfdmaUplinkChannelEntry 3 }
29
30
31 wman2IfBsOfdmaUplinkCenterFreq OBJECT-TYPE
32     SYNTAX      Unsigned32
33     UNITS        "kHz"
34     MAX-ACCESS  read-write
35     STATUS      current
36     DESCRIPTION
37         " Uplink center frequency (kHz)"
38     REFERENCE
39         "Table 349, in IEEE Std 802.16-2004"
40     ::= { wman2IfBsOfdmaUplinkChannelEntry 4 }
41
42
43 wman2IfBsOfdmaInitRngCodes OBJECT-TYPE
44     SYNTAX      INTEGER (0..255)
45     MAX-ACCESS  read-write
46     STATUS      current
47     DESCRIPTION
48         "Number of initial ranging CDMA codes. Possible values are
49         0..255. The total number of wman2IfBsOfdmaInitRngCodes,
50         wman2IfBsOfdmaPeriodicRngCodes and wman2IfBsOfdmaBWRngCodes
51         shall be equal or less than 256."
52     REFERENCE
53         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
54     DEFVAL      { 30 }
55     ::= { wman2IfBsOfdmaUplinkChannelEntry 5 }
56
57
58 wman2IfBsOfdmaPeriodicRngCodes OBJECT-TYPE
59     SYNTAX      INTEGER (0..255)
60
61
62
63
64
65

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "Number of periodic ranging CDMA codes. Possible values are
5              0..255. The total number of wman2IfBsOfdmaInitRngCodes,
6              wman2IfBsOfdmaPeriodicRngCodes and wman2IfBsOfdmaBWReqCodes
7              shall be equal or less than 256."
8
9      REFERENCE
10         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
11
12      DEFVAL      { 30 }
13      ::= { wman2IfBsOfdmaUplinkChannelEntry 6 }
14
15  wman2IfBsOfdmaBWReqCodes OBJECT-TYPE
16      SYNTAX      INTEGER (0..255)
17      MAX-ACCESS  read-write
18      STATUS      current
19      DESCRIPTION
20          "Number of bandwidth request codes. Possible values are
21              0..255. The total number of wman2IfBsOfdmaInitRngCodes,
22              wman2IfBsOfdmaPeriodicRngCodes and wman2IfBsOfdmaBWReqCodes
23              shall be equal or less than 256."
24
25      REFERENCE
26         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
27
28      DEFVAL      { 30 }
29      ::= { wman2IfBsOfdmaUplinkChannelEntry 7 }
30
31
32  wman2IfBsOfdmaPerRngBackoffStart OBJECT-TYPE
33      SYNTAX      INTEGER (0..15)
34      MAX-ACCESS  read-write
35      STATUS      current
36      DESCRIPTION
37          "Initial backoff window size for periodic ranging contention,
38              , expressed as a power of 2. Range: 0..15 (the highest order
39              bits shall be unused and set to 0)."
40
41      REFERENCE
42         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
43
44      DEFVAL      { 0 }
45      ::= { wman2IfBsOfdmaUplinkChannelEntry 8 }
46
47
48  wman2IfBsOfdmaPerRngBackoffEnd OBJECT-TYPE
49      SYNTAX      INTEGER (0 .. 15)
50      MAX-ACCESS  read-write
51      STATUS      current
52      DESCRIPTION
53          "Final backoff window size for periodic ranging contention,
54              expressed as a power of 2. Range: 0..15 (the highest order
55              bits shall be unused and set to 0)."
56
57      REFERENCE
58         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
59
60      DEFVAL      { 15 }
61      ::= { wman2IfBsOfdmaUplinkChannelEntry 9 }
62
63
64  wman2IfBsOfdmaStartOfRngCodes OBJECT-TYPE
65      SYNTAX      INTEGER (0..255)

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "Indicates the starting number, S, of the group of codes
5          used for this uplink. All the ranging codes used on this
6          uplink will be between S and ((S+N+M+L) mod 256). Where,
7          N is the number of initial-ranging codes M is the number
8          of periodic-ranging codes L is the number of
9          bandwidth-request codes The range of values is 0 S255"
10
11      REFERENCE
12          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
13
14      DEFVAL      { 0 }
15      ::= { wman2IfBsOfdmaUplinkChannelEntry 10 }
16
17
18  wman2IfBsOfdmaPermutationBase OBJECT-TYPE
19      SYNTAX      INTEGER (0..255)
20      MAX-ACCESS  read-write
21      STATUS      current
22      DESCRIPTION
23          "Determines the UL_IDcell parameter for the subcarrier
24          permutation to be used on this uplink channel"
25
26      REFERENCE
27          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
28
29      DEFVAL      { 0 }
30      ::= { wman2IfBsOfdmaUplinkChannelEntry 11 }
31
32
33  wman2IfBsOfdmaULAllocSubchBitmap OBJECT-TYPE
34      SYNTAX      OCTET STRING (SIZE (9))
35      MAX-ACCESS  read-write
36      STATUS      current
37      DESCRIPTION
38          "This is a bitmap describing the sub-channels allocated
39          to the segment in the UL, when using the uplink PUSC
40          permutation. The LSB of the first byte shall correspond to
41          subchannel 0. For any bit that is not set,
42          the corresponding subchannel shall not be used by the SS
43          on that segment"
44
45      REFERENCE
46          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
47
48      ::= { wman2IfBsOfdmaUplinkChannelEntry 12 }
49
50
51  wman2IfBsOfdmaOptPermULAllocSubchBitmap OBJECT-TYPE
52      SYNTAX      OCTET STRING (SIZE (13))
53      MAX-ACCESS  read-write
54      STATUS      current
55      DESCRIPTION
56          "This is a bitmap describing the sub-channels allocated to
57          the segment in the UL, when using the uplink optional PUSC
58          permutation (see 8.4.6.2.5 in IEEE Std 802.16-2004). The
59          LSB of the first byte shall correspond to subchannel 0.
60          For any bit that is not set, the corresponding subchannel
61          shall not be used by the SS on that segment"
62
63      REFERENCE
64          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
65

```

```

1      ::= { wman2IfBsOfdmaUplinkChannelEntry 13 }
2
3
4  wman2IfBsOfdmaBandAMCAallocThreshold OBJECT-TYPE
5      SYNTAX      INTEGER (0 .. 255)
6      UNITS        "dB"
7      MAX-ACCESS   read-write
8      STATUS       current
9      DESCRIPTION
10         "dB unit"
11
12      REFERENCE
13         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
14      ::= { wman2IfBsOfdmaUplinkChannelEntry 14 }
15
16
17  wman2IfBsOfdmaBandAMCReleaseThreshold OBJECT-TYPE
18      SYNTAX      INTEGER (0 .. 255)
19      UNITS        "dB"
20      MAX-ACCESS   read-write
21      STATUS       current
22      DESCRIPTION
23         "This object defines the OFDMA band AMC release
24         threshold."
25
26      REFERENCE
27         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
28      ::= { wman2IfBsOfdmaUplinkChannelEntry 15 }
29
30
31  wman2IfBsOfdmaBandAMCAallocTimer OBJECT-TYPE
32      SYNTAX      INTEGER (0 .. 255)
33      UNITS        "Frame"
34      MAX-ACCESS   read-write
35      STATUS       current
36      DESCRIPTION
37         "This object defines the OFDMA band AMC allocation
38         timer."
39
40      REFERENCE
41         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
42      ::= { wman2IfBsOfdmaUplinkChannelEntry 16 }
43
44
45  wman2IfBsOfdmaBandAMCReleaseTimer OBJECT-TYPE
46      SYNTAX      INTEGER (0 .. 255)
47      UNITS        "Frame"
48      MAX-ACCESS   read-write
49      STATUS       current
50      DESCRIPTION
51         "This object defines the OFDMA band AMC release
52         timer."
53
54      REFERENCE
55         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
56      ::= { wman2IfBsOfdmaUplinkChannelEntry 17 }
57
58
59  wman2IfBsOfdmaBandStatRepMAXPeriod OBJECT-TYPE
60      SYNTAX      INTEGER (0 .. 255)
61      UNITS        "Frame"
62      MAX-ACCESS   read-write
63      STATUS       current
64
65

```

```

1      DESCRIPTION
2          "This object defines the OFDMA band status reporting
3          maximum period."
4
5      REFERENCE
6          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
7      ::= { wman2IfBsOfdmaUplinkChannelEntry 18 }
8
9
10     wman2IfBsOfdmaBandAMCRetryTimer OBJECT-TYPE
11         SYNTAX      INTEGER (0 .. 255)
12         UNITS        "Frame"
13         MAX-ACCESS   read-write
14         STATUS       current
15         DESCRIPTION
16             "This object defines the OFDMA band AMC retry
17             timer."
18
19         REFERENCE
20             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
21         ::= { wman2IfBsOfdmaUplinkChannelEntry 19 }
22
23
24     wman2IfBsOfdmaSafetyChAllocThreshold OBJECT-TYPE
25         SYNTAX      INTEGER (0 .. 255)
26         UNITS        "dB"
27         MAX-ACCESS   read-write
28         STATUS       current
29         DESCRIPTION
30             "This object defines the OFDMA safety channel allocation
31             threshold."
32
33         REFERENCE
34             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
35         ::= { wman2IfBsOfdmaUplinkChannelEntry 20 }
36
37
38     wman2IfBsOfdmaSafetyChReleaseThreshold OBJECT-TYPE
39         SYNTAX      INTEGER (0 .. 255)
40         UNITS        "dB"
41         MAX-ACCESS   read-write
42         STATUS       current
43         DESCRIPTION
44             "This object defines the OFDMA safety channel release
45             threshold."
46
47         REFERENCE
48             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
49         ::= { wman2IfBsOfdmaUplinkChannelEntry 21 }
50
51
52     wman2IfBsOfdmaSafetyChAllocTimer OBJECT-TYPE
53         SYNTAX      INTEGER (0 .. 255)
54         UNITS        "Frame"
55         MAX-ACCESS   read-write
56         STATUS       current
57         DESCRIPTION
58             "This object defines the OFDMA safety channel allocation
59             timer."
60
61         REFERENCE
62             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
63         ::= { wman2IfBsOfdmaUplinkChannelEntry 22 }
64
65

```



```

1
2 wman2IfBsOfdmaSafetyChReleaseTimer OBJECT-TYPE
3     SYNTAX      INTEGER (0 .. 255)
4     UNITS       "Frame"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "This object defines the OFDMA safety channel release
9         timer."
10
11     REFERENCE
12         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
13     ::= { wman2IfBsOfdmaUplinkChannelEntry 23 }
14
15
16 wman2IfBsOfdmaBinStatRepMAXPeriod OBJECT-TYPE
17     SYNTAX      INTEGER (0 .. 255)
18     UNITS       "Frame"
19     MAX-ACCESS  read-write
20     STATUS      current
21     DESCRIPTION
22         "This object defines the OFDMA bin status reporting
23         maximum period."
24
25     REFERENCE
26         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
27     ::= { wman2IfBsOfdmaUplinkChannelEntry 24 }
28
29
30
31 wman2IfBsOfdmaSafetyChaRetryTimer OBJECT-TYPE
32     SYNTAX      INTEGER (0 .. 255)
33     UNITS       "Frame"
34     MAX-ACCESS  read-write
35     STATUS      current
36     DESCRIPTION
37         "This object defines the OFDMA safety channel retry
38         timer."
39
40     REFERENCE
41         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
42     ::= { wman2IfBsOfdmaUplinkChannelEntry 25 }
43
44
45
46 wman2IfBsOfdmaHARQAackDelayULBurst OBJECT-TYPE
47     SYNTAX      INTEGER {oneframeoffset(1),
48                        twoframesoffset(2),
49                        threeframesoffset(3)}
50     MAX-ACCESS  read-write
51     STATUS      current
52     DESCRIPTION
53         "This object defines the OFDMA H-ARQ ACK delay for UL burst.
54         1 = one frame offset
55         2 = two frames offset
56         3 = three frames offset"
57
58     REFERENCE
59         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
60     ::= { wman2IfBsOfdmaUplinkChannelEntry 26 }
61
62
63
64 wman2IfBsOfdmaCQICHBandAMCTranaDelay OBJECT-TYPE
65     SYNTAX      INTEGER (0 .. 255)

```

```

1      UNITS          "Frame"
2      MAX-ACCESS    read-write
3      STATUS        current
4      DESCRIPTION
5          "This object defines the OFDMA CQICH band AMC transition
6          delay."
7      REFERENCE
8          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
9      ::= { wman2IfBsOfdmaUplinkChannelEntry 27 }
10
11
12
13 wman2IfBsOfdmaDownlinkChannelTable OBJECT-TYPE
14     SYNTAX          SEQUENCE OF Wman2IfBsOfdmaDownlinkChannelEntry
15     MAX-ACCESS      not-accessible
16     STATUS          current
17     DESCRIPTION
18         "This table contains DCD channel attributes, defining the
19         transmission characteristics of downlink channels"
20     REFERENCE
21         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
22     ::= { wman2IfBsOfdmaPhy 2 }
23
24
25
26 wman2IfBsOfdmaDownlinkChannelEntry OBJECT-TYPE
27     SYNTAX          Wman2IfBsOfdmaDownlinkChannelEntry
28     MAX-ACCESS      not-accessible
29     STATUS          current
30     DESCRIPTION
31         "This table provides one row for each downlink channel of
32         multi-sector BS, and is indexed by BS ifIndex. An entry in
33         this table exists for each ifEntry of BS with an ifType of
34         propBWA2Mp."
35     INDEX            { ifIndex }
36     ::= { wman2IfBsOfdmaDownlinkChannelTable 1 }
37
38
39
40
41 Wman2IfBsOfdmaDownlinkChannelEntry ::= SEQUENCE {
42     wman2IfBsOfdmaBsEIRP                INTEGER,
43     wman2IfBsOfdmaChannelNumber          Wman2IfChannelNumber,
44     wman2IfBsOfdmaATTG                   INTEGER,
45     wman2IfBsOfdmaARTG                   INTEGER,
46     wman2IfBsOfdmaInitRngMaxRSS          INTEGER,
47     wman2IfBsOfdmaDownlinkCenterFreq     Unsigned32,
48     wman2IfBsOfdmaBsId                   Wman2IfBsIdType,
49     wman2IfBsOfdmaMacVersion              Wman2IfMacVersion,
50     wman2IfBsOfdmaFrameDurationCode      INTEGER,
51     wman2IfBsOfdmaSizeCqichIdField       INTEGER,
52     wman2IfBsOfdmaHARQAackDelayBurst     INTEGER}
53
54
55
56 wman2IfBsOfdmaBsEIRP OBJECT-TYPE
57     SYNTAX          INTEGER (-32768..32767)
58     UNITS            "dBm"
59     MAX-ACCESS      read-write
60     STATUS          current
61     DESCRIPTION
62         "The EIRP is the equivalent isotropic radiated power of
63         the base station, which is computed for a simple
64
65

```

```

1         single-antenna transmitter."
2     REFERENCE
3         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
4     ::= { wman2IfBsOfdmaDownlinkChannelEntry 1 }
5
6
7     wman2IfBsOfdmaChannelNumber OBJECT-TYPE
8         SYNTAX      Wman2IfChannelNumber
9         MAX-ACCESS   read-write
10        STATUS      current
11        DESCRIPTION
12            "Downlink channel number as defined in 8.5. Used for
13             license-exempt operation only."
14        REFERENCE
15            "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
16        ::= { wman2IfBsOfdmaDownlinkChannelEntry 2 }
17
18
19
20    wman2IfBsOfdmaTTG OBJECT-TYPE
21        SYNTAX      INTEGER (0..255)
22        MAX-ACCESS   read-write
23        STATUS      current
24        DESCRIPTION
25            "Transmit / Receive Transition Gap."
26        REFERENCE
27            "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
28        ::= { wman2IfBsOfdmaDownlinkChannelEntry 3 }
29
30
31
32    wman2IfBsOfdmaRTG OBJECT-TYPE
33        SYNTAX      INTEGER (0..255)
34        MAX-ACCESS   read-write
35        STATUS      current
36        DESCRIPTION
37            "Receive / Transmit Transition Gap."
38        REFERENCE
39            "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
40        ::= { wman2IfBsOfdmaDownlinkChannelEntry 4 }
41
42
43
44    wman2IfBsOfdmaInitRngMaxRSS OBJECT-TYPE
45        SYNTAX      INTEGER (-32768..32767)
46        UNITS        "dBm"
47        MAX-ACCESS   read-write
48        STATUS      current
49        DESCRIPTION
50            "Initial Ranging Max. equivalent isotropic received power
51             at BS Signed in units of 1 dBm."
52        REFERENCE
53            "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
54        ::= { wman2IfBsOfdmaDownlinkChannelEntry 5 }
55
56
57
58    wman2IfBsOfdmaDownlinkCenterFreq OBJECT-TYPE
59        SYNTAX      Unsigned32
60        UNITS        "kHz"
61        MAX-ACCESS   read-write
62        STATUS      current
63        DESCRIPTION
64
65

```

```

1         "Downlink center frequency (kHz)."
```

REFERENCE

```

3         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
```

```

4         ::= { wman2IfBsOfdmaDownlinkChannelEntry 6 }
```

wman2IfBsOfdmaBsId OBJECT-TYPE

```

8         SYNTAX      Wman2IfBsIdType
9         MAX-ACCESS   read-write
10        STATUS      current
11        DESCRIPTION
12
13            "Base station ID."
```

REFERENCE

```

15        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
```

```

16        ::= { wman2IfBsOfdmaDownlinkChannelEntry 7 }
```

wman2IfBsOfdmaMacVersion OBJECT-TYPE

```

20        SYNTAX      Wman2IfMacVersion
21        MAX-ACCESS   read-write
22        STATUS      current
23        DESCRIPTION
24
25            "This parameter specifies the version of 802.16 to which
26            the message originator conforms."
```

REFERENCE

```

28        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
```

```

29        ::= { wman2IfBsOfdmaDownlinkChannelEntry 8 }
```

wman2IfBsOfdmaFrameDurationCode OBJECT-TYPE

```

33        SYNTAX      INTEGER { aASGap(0),
34                               duration2ms(1),
35                               duration2dot5ms(2),
36                               duration4ms(3),
37                               duration5ms(4),
38                               duration8ms(5),
39                               duration10ms(6),
40                               duration12dot5ms(7),
41                               duration20ms(8) }
```

```

44        MAX-ACCESS   read-write
45        STATUS      current
46        DESCRIPTION
47
48            "The duration of the frame. The frame duration code values
49            are specified in Table 274."
```

REFERENCE

```

51        "Table 273, in IEEE Std 802.16-2004"
```

```

52        ::= { wman2IfBsOfdmaDownlinkChannelEntry 9 }
```

wman2IfBsOfdmaSizeCqichIdField OBJECT-TYPE

```

56        SYNTAX      INTEGER { threebits(1),
57                               fourbits(2),
58                               fivebits(3),
59                               sixbits(4),
60                               sevenbits(5),
61                               eightbits(6),
62                               ninebits(7) }
```

```

64        MAX-ACCESS   read-write
```

```

1      STATUS      current
2      DESCRIPTION
3          "This object defines the size of CQICH ID field.
4              0 = Reserved
5              1 = 3 bits
6              2 = 4 bits
7              3 = 5 bits
8              4 = 6 bits
9              5 = 7 bits
10             6 = 8 bits
11             7 = 9 bits
12             8...255 = Reserved"
13
14      REFERENCE
15          "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
16      ::= { wman2IfBsOfdmaDownlinkChannelEntry 10 }
17
18  wman2IfBsOfdmaHARQAackDelayBurst OBJECT-TYPE
19      SYNTAX      INTEGER {oneframeoffset(1),
20                      twoframesoffset(2),
21                      threeframesoffset(3)}
22      MAX-ACCESS  read-write
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      ::= { wman2IfBsOfdmaDownlinkChannelEntry 11 }
33
34  wman2IfBsOfdmaUcdBurstProfileTable OBJECT-TYPE
35      SYNTAX      SEQUENCE OF Wman2IfBsOfdmaUcdBurstProfileEntry
36      MAX-ACCESS  not-accessible
37      STATUS      current
38      DESCRIPTION
39          "This table contains UCD burst profiles for each uplink
40          channel"
41
42      REFERENCE
43          "Table 356, in IEEE Std 802.16-2004"
44      ::= { wman2IfBsOfdmaPhy 3 }
45
46  wman2IfBsOfdmaUcdBurstProfileEntry OBJECT-TYPE
47      SYNTAX      Wman2IfBsOfdmaUcdBurstProfileEntry
48      MAX-ACCESS  not-accessible
49      STATUS      current
50      DESCRIPTION
51          "This table provides one row for each UCD burst profile.
52          This table is double indexed. The primary index is an
53          ifIndex with an ifType of propBWApm2Mp. The secondary index
54          is wman2IfBsOfdmaUiucIndex."
55
56      INDEX      { ifIndex, wman2IfBsOfdmaUiucIndex }
57      ::= { wman2IfBsOfdmaUcdBurstProfileTable 1 }
58
59
60
61
62
63
64
65

```

```

1  Wman2IfBsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
2      wman2IfBsOfdmaUiucIndex          INTEGER,
3      wman2IfBsOfdmaUcdFecCodeType     Wman2IfOfdmaFecCodeType,
4      wman2IfBsOfdmaRangingDataRatio   INTEGER,
5      wman2IfBsOfdmaNorCOverNOVERRIDE  OCTET STRING,
6      wman2IfBsOfdmaUcdBurstProfileRowStatus RowStatus}
7
8
9
10 wman2IfBsOfdmaUiucIndex OBJECT-TYPE
11     SYNTAX      INTEGER (1 .. 10)
12     MAX-ACCESS  not-accessible
13     STATUS      current
14     DESCRIPTION
15         "The Uplink Interval Usage Code indicates the uplink burst
16         profile in the UCD message, and is used along with ifIndex
17         to identify an entry in the
18         wman2IfBsOfdmaUcdBurstProfileTable."
19     REFERENCE
20         "Subclause 8.4.5.4.1, in IEEE Std 802.16-2004"
21     ::= { wman2IfBsOfdmaUcdBurstProfileEntry 1 }
22
23
24
25 wman2IfBsOfdmaUcdFecCodeType OBJECT-TYPE
26     SYNTAX      Wman2IfOfdmaFecCodeType
27     MAX-ACCESS  read-create
28     STATUS      current
29     DESCRIPTION
30         "Uplink FEC code type and modulation type"
31     REFERENCE
32         "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
33     ::= { wman2IfBsOfdmaUcdBurstProfileEntry 2 }
34
35
36
37 wman2IfBsOfdmaRangingDataRatio OBJECT-TYPE
38     SYNTAX      INTEGER (0 .. 255)
39     MAX-ACCESS  read-create
40     STATUS      current
41     DESCRIPTION
42         "Reducing factor in units of 1 dB, between the power used
43         for this burst and power should be used for CDMA Ranging."
44     REFERENCE
45         "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
46     ::= { wman2IfBsOfdmaUcdBurstProfileEntry 3 }
47
48
49
50 wman2IfBsOfdmaNorCOverNOVERRIDE OBJECT-TYPE
51     SYNTAX      OCTET STRING (SIZE (5))
52     MAX-ACCESS  read-create
53     STATUS      current
54     DESCRIPTION
55         "This is a list of numbers, where each number is encoded by
56         one nibble, and interpreted as a signed integer. The nibbles
57         correspond in order to the list define by Table 334 in IEEE
58         Std 802.16-2004 starting from the second line, such that
59         the LS nibble of the first byte corresponds to the second
60         line in the table. The number encoded by each nibble
61         represents the difference in normalized C/N relative to the
62         previous line in the table"
63
64
65

```

REFERENCE

"Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
 ::= { wman2IfBsOfdmaUcdBurstProfileEntry 4 }

wman2IfBsOfdmaUcdBurstProfileRowStatus 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 upon SNMP request."
 ::= { wman2IfBsOfdmaUcdBurstProfileEntry 5 }

wman2IfBsOfdmaDcdBurstProfileTable OBJECT-TYPE

SYNTAX SEQUENCE OF Wman2IfBsOfdmaDcdBurstProfileEntry
 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 wman2IfBsOfdmaDiucIndex."
 ::= { wman2IfBsOfdmaPhy 4 }

wman2IfBsOfdmaDcdBurstProfileEntry OBJECT-TYPE

SYNTAX Wman2IfBsOfdmaDcdBurstProfileEntry
 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 wman2IfBsOfdmaDiucIndex."
 INDEX { ifIndex, wman2IfBsOfdmaDiucIndex }
 ::= { wman2IfBsOfdmaDcdBurstProfileTable 1 }

Wman2IfBsOfdmaDcdBurstProfileEntry ::= SEQUENCE {

wman2IfBsOfdmaDiucIndex INTEGER,
 wman2IfBsOfdmaDownlinkFrequency Unsigned32,
 wman2IfBsOfdmaDcdFecCodeType Wman2IfOfdmaFecCodeType,
 wman2IfBsOfdmaDiucMandatoryExitThresh INTEGER,
 wman2IfBsOfdmaDiucMinEntryThresh INTEGER,
 wman2IfBsOfdmaDcdBurstProfileRowStatus RowStatus }

wman2IfBsOfdmaDiucIndex OBJECT-TYPE

SYNTAX INTEGER (0 .. 12)
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION

"The Downlink Interval Usage Code indicates the downlink burst profile in the DCD message, and is used along with

```

1         ifIndex to identify an entry in the
2         wman2IfBsOfdmaDcdBurstProfileTable."
3     REFERENCE
4         "Subclause 8.4.5.3.1, in IEEE Std 802.16-2004"
5     ::= { wman2IfBsOfdmaDcdBurstProfileEntry 1 }
6
7
8     wman2IfBsOfdmaDownlinkFrequency OBJECT-TYPE
9         SYNTAX      Unsigned32
10        UNITS       "kHz"
11        MAX-ACCESS   read-create
12        STATUS       current
13        DESCRIPTION
14            "Downlink Frequency (kHz)."

```



```

1      MAX-ACCESS    read-create
2      STATUS        current
3      DESCRIPTION
4          "This object is used to create a new row or modify or delete
5          an existing row in this table. If the implementator of this
6          MIB has choosen not to implement 'dynamic assignment' of
7          profiles, this object is not useful and should return
8          noSuchName upon SNMP request."
9      ::= { wman2IfBsOfdmaDcdBurstProfileEntry 6 }
10
11
12
13 wman2IfBsMsOfdmaReqCapabilitiesTable OBJECT-TYPE
14     SYNTAX          SEQUENCE OF Wman2IfBsMsOfdmaReqCapabilitiesEntry
15     MAX-ACCESS      not-accessible
16     STATUS          current
17     DESCRIPTION
18         "This table contains the basic capability information,
19         specific to OFDMA Phy, of MSs that have been reported by
20         MSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.
21         Entries in this table should be created when an MS
22         registers with a BS."
23     ::= { wman2IfBsOfdmaPhy 5 }
24
25
26
27 wman2IfBsMsOfdmaReqCapabilitiesEntry OBJECT-TYPE
28     SYNTAX          Wman2IfBsMsOfdmaReqCapabilitiesEntry
29     MAX-ACCESS      not-accessible
30     STATUS          current
31     DESCRIPTION
32         "This table provides one row for each MS that has been
33         registered in the BS. This table augments the table
34         wman2IfBsRegisteredSsTable."
35     AUGMENTS { wman2IfBsRegisteredSsEntry }
36     ::= { wman2IfBsMsOfdmaReqCapabilitiesTable 1 }
37
38
39
40
41 Wman2IfBsMsOfdmaReqCapabilitiesEntry ::= SEQUENCE {
42     wman2IfBsMsOfdmaReqCapFftSizes          Wman2IfOfdmFftSizes,
43     wman2IfBsMsOfdmaReqCapDemodulator        Wman2IfOfdmaMsDeModType,
44     wman2IfBsMsOfdmaReqCapModulator          Wman2IfOfdmaMsModType,
45     wman2IfBsMsOfdmaReqCapPermutation        Wman2IfOfdmaPermutation,
46     wman2IfBsMsOfdmaReqCapMobilityFeature    Wman2IfOfdmaMobility}
47
48
49 wman2IfBsMsOfdmaReqCapFftSizes OBJECT-TYPE
50     SYNTAX          Wman2IfOfdmFftSizes
51     MAX-ACCESS      read-only
52     STATUS          current
53     DESCRIPTION
54         "This field indicates the FFT sizes supported by MS."
55     ::= { wman2IfBsMsOfdmaReqCapabilitiesEntry 1 }
56
57
58
59 wman2IfBsMsOfdmaReqCapDemodulator OBJECT-TYPE
60     SYNTAX          Wman2IfOfdmaMsDeModType
61     MAX-ACCESS      read-only
62     STATUS          current
63     DESCRIPTION
64         "This field indicates the different demodulator options
65

```

```

1         supported by MS for downlink."
2         ::= { wman2IfBsMsOfdmaReqCapabilitiesEntry 2 }
3
4
5 wman2IfBsMsOfdmaReqCapModulator OBJECT-TYPE
6     SYNTAX      Wman2IfOfdmaMsModType
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "This field indicates the different modulator options
11        supported by MS for uplink."
12        ::= { wman2IfBsMsOfdmaReqCapabilitiesEntry 3 }
13
14
15 wman2IfBsMsOfdmaReqCapPermutation OBJECT-TYPE
16     SYNTAX      Wman2IfOfdmaPermutation
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20        "This field indicates the OFDMA MS Permutation support"
21        ::= { wman2IfBsMsOfdmaReqCapabilitiesEntry 4 }
22
23
24
25 wman2IfBsMsOfdmaReqCapMobilityFeature OBJECT-TYPE
26     SYNTAX      Wman2IfOfdmaMobility
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30        "The field indicates whether or not the MS supports
31        mobility hand-over, Sleepmode, and Idle-mode."
32        ::= { wman2IfBsMsOfdmaReqCapabilitiesEntry 5 }
33
34
35
36 wman2IfBsMsOfdmaRspCapabilitiesTable OBJECT-TYPE
37     SYNTAX      SEQUENCE OF Wman2IfBsMsOfdmaRspCapabilitiesEntry
38     MAX-ACCESS  not-accessible
39     STATUS      current
40     DESCRIPTION
41        "This table contains the basic capability information,
42        specific to OFDMA Phy, of MSs that have been reported by
43        MSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.
44        Entries in this table should be created when an MS
45        registers with a BS."
46        ::= { wman2IfBsOfdmaPhy 6 }
47
48
49
50 wman2IfBsMsOfdmaRspCapabilitiesEntry OBJECT-TYPE
51     SYNTAX      Wman2IfBsMsOfdmaRspCapabilitiesEntry
52     MAX-ACCESS  not-accessible
53     STATUS      current
54     DESCRIPTION
55        "This table provides one row for each MS that has been
56        registered in the BS. This table augments the table
57        wman2IfBsRegisteredSsTable."
58        AUGMENTS { wman2IfBsRegisteredSsEntry }
59        ::= { wman2IfBsMsOfdmaRspCapabilitiesTable 1 }
60
61
62
63 Wman2IfBsMsOfdmaRspCapabilitiesEntry ::= SEQUENCE {
64     wman2IfBsMsOfdmaRspCapFftSizes      Wman2IfOfdmFftSizes,
65

```

```

1      wman2IfBsMsOfdmaRspCapDemodulator      Wman2IfOfdmaMsDeModType,
2      wman2IfBsMsOfdmaRspCapModulator         Wman2IfOfdmaMsModType,
3      wman2IfBsMsOfdmaRspCapPermutation       Wman2IfOfdmaPermutation,
4      wman2IfBsMsOfdmaRspCapMobilityFeature   Wman2IfOfdmaMobility}
5
6
7      wman2IfBsMsOfdmaRspCapFftSizes OBJECT-TYPE
8          SYNTAX      Wman2IfOfdmFftSizes
9          MAX-ACCESS  read-only
10         STATUS      current
11         DESCRIPTION
12             "This field indicates the FFT sizes negotiated with the
13             MS."
14         ::= { wman2IfBsMsOfdmaRspCapabilitiesEntry 1 }
15
16
17      wman2IfBsMsOfdmaRspCapDemodulator OBJECT-TYPE
18          SYNTAX      Wman2IfOfdmaMsDeModType
19          MAX-ACCESS  read-only
20          STATUS      current
21          DESCRIPTION
22              "This field indicates the different demodulator options
23              negotiated for MS for downlink."
24          ::= { wman2IfBsMsOfdmaRspCapabilitiesEntry 2 }
25
26
27      wman2IfBsMsOfdmaRspCapModulator OBJECT-TYPE
28          SYNTAX      Wman2IfOfdmaMsModType
29          MAX-ACCESS  read-only
30          STATUS      current
31          DESCRIPTION
32              "This field indicates the different modulator options
33              negotiated for MS for uplink."
34          ::= { wman2IfBsMsOfdmaRspCapabilitiesEntry 3 }
35
36
37      wman2IfBsMsOfdmaRspCapPermutation OBJECT-TYPE
38          SYNTAX      Wman2IfOfdmaPermutation
39          MAX-ACCESS  read-only
40          STATUS      current
41          DESCRIPTION
42              "This field indicates the OFDMA MS Permutation support
43              negotiated for MS."
44          ::= { wman2IfBsMsOfdmaRspCapabilitiesEntry 4 }
45
46
47      wman2IfBsMsOfdmaRspCapMobilityFeature OBJECT-TYPE
48          SYNTAX      Wman2IfOfdmaMobility
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          ::= { wman2IfBsMsOfdmaRspCapabilitiesEntry 5 }
55
56
57      wman2IfBsOfdmaCapabilitiesTable OBJECT-TYPE
58          SYNTAX      SEQUENCE OF Wman2IfBsOfdmaCapabilitiesEntry
59          MAX-ACCESS  not-accessible
60          STATUS      current

```

```

1      DESCRIPTION
2          "This table contains the basic capabilities, specific to
3          OFDMA Phy, of the BS as implemented in BS hardware and
4          software. These capabilities along with the configuration
5          for them (wman2IfBsOfdmaCapabilitiesConfigTable) are used
6          for negotiation of basic capabilities with SS using
7          RNG-RSP, SBC-RSP and REG-RSP messages. The negotiated
8          capabilities are obtained by interSubclause of MS raw
9          reported capabilities, BS raw capabilities and BS
10         configured capabilities. The objects in the table have
11         read-only access. The table is maintained by BS."
12         ::= { wman2IfBsOfdmaPhy 7 }
13
14 wman2IfBsOfdmaCapabilitiesEntry OBJECT-TYPE
15     SYNTAX      Wman2IfBsOfdmaCapabilitiesEntry
16     MAX-ACCESS  not-accessible
17     STATUS      current
18     DESCRIPTION
19         "This table provides one row for each BS sector and is
20         indexed by ifIndex."
21     INDEX { ifIndex }
22     ::= { wman2IfBsOfdmaCapabilitiesTable 1 }
23
24 Wman2IfBsOfdmaCapabilitiesEntry ::= SEQUENCE {
25     wman2IfBsOfdmaCapFftSizes          Wman2IfOfdmFftSizes,
26     wman2IfBsOfdmaCapDemodulator        Wman2IfOfdmaMsDeModType,
27     wman2IfBsOfdmaCapModulator          Wman2IfOfdmaMsModType,
28     wman2IfBsOfdmaCapPermutation        Wman2IfOfdmaPermutation,
29     wman2IfBsOfdmaCapMobilityFeature    Wman2IfOfdmaMobility}
30
31 wman2IfBsOfdmaCapFftSizes OBJECT-TYPE
32     SYNTAX      Wman2IfOfdmFftSizes
33     MAX-ACCESS  read-only
34     STATUS      current
35     DESCRIPTION
36         "This field indicates the FFT sizes supported by BS."
37     ::= { wman2IfBsOfdmaCapabilitiesEntry 1 }
38
39 wman2IfBsOfdmaCapDemodulator OBJECT-TYPE
40     SYNTAX      Wman2IfOfdmaMsDeModType
41     MAX-ACCESS  read-only
42     STATUS      current
43     DESCRIPTION
44         "This field indicates the different demodulator options
45         supported by BS."
46     ::= { wman2IfBsOfdmaCapabilitiesEntry 2 }
47
48 wman2IfBsOfdmaCapModulator OBJECT-TYPE
49     SYNTAX      Wman2IfOfdmaMsModType
50     MAX-ACCESS  read-only
51     STATUS      current
52     DESCRIPTION
53         "This field indicates the different modulator options
54         supported by BS."
55
56
57
58
59
60
61
62
63
64
65

```

```

1      ::= { wman2IfBsOfdmaCapabilitiesEntry 3 }
2
3
4  wman2IfBsOfdmaCapPermutation OBJECT-TYPE
5      SYNTAX      Wman2IfOfdmaPermutation
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "This field indicates the OFDMA MS Permutation support
10         supported by BS."
11      ::= { wman2IfBsOfdmaCapabilitiesEntry 4 }
12
13
14  wman2IfBsOfdmaCapMobilityFeature OBJECT-TYPE
15      SYNTAX      Wman2IfOfdmaMobility
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "The field indicates the mobility hand-over, Sleepmode,
20         and Idle-mode supported by BS."
21      ::= { wman2IfBsOfdmaCapabilitiesEntry 5 }
22
23
24
25  wman2IfBsOfdmaCapabilitiesConfigTable OBJECT-TYPE
26      SYNTAX      SEQUENCE OF Wman2IfBsOfdmaCapabilitiesConfigEntry
27      MAX-ACCESS  not-accessible
28      STATUS      current
29      DESCRIPTION
30          "This table contains the configuration for basic
31         capabilities of BS, specific to OFDMA Phy. The table is
32         intended to be used to restrict the Capabilities
33         implemented by BS, for example in order to comply with
34         local regulatory requirements. The BS should use the
35         configuration along with the implemented Capabilities
36         (wman2IfBsOfdmaPhyTable) for negotiation of basic
37         capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP
38         messages. The negotiated capabilities are obtained by
39         interSubclause of MS reported capabilities, BS raw
40         capabilities and BS configured capabilities. The objects
41         in the table have read-write access. The rows are created
42         by BS as a copy of wman2IfBsBasicCapabilitiesTable
43         and can be modified by NMS."
44      ::= { wman2IfBsOfdmaPhy 8 }
45
46
47
48
49
50  wman2IfBsOfdmaCapabilitiesConfigEntry OBJECT-TYPE
51      SYNTAX      Wman2IfBsOfdmaCapabilitiesConfigEntry
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      ::= { wman2IfBsOfdmaCapabilitiesConfigTable 1 }
59
60
61
62  Wman2IfBsOfdmaCapabilitiesConfigEntry ::= SEQUENCE {
63      wman2IfBsOfdmaCapCfgFftSizes          Wman2IfOfdmFftSizes,
64      wman2IfBsOfdmaCapCfgDemodulator       Wman2IfOfdmaMsDeModType,
65

```

```

1          wman2IfBsOfdmaCapCfgModulator          Wman2IfOfdmaMsModType,
2          wman2IfBsOfdmaCapCfgPermutation        Wman2IfOfdmaPermutation,
3          wman2IfBsOfdmaCapCfgMobilityFeature    Wman2IfOfdmaMobility}
4
5
6  wman2IfBsOfdmaCapCfgFftSizes OBJECT-TYPE
7      SYNTAX      Wman2IfOfdmFftSizes
8      MAX-ACCESS  read-only
9      STATUS      current
10     DESCRIPTION
11         "This field indicates the FFT sizes configured for the BS."
12         ::= { wman2IfBsOfdmaCapabilitiesConfigEntry 1 }
13
14
15  wman2IfBsOfdmaCapCfgDemodulator OBJECT-TYPE
16      SYNTAX      Wman2IfOfdmaMsDeModType
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20         "This field indicates the different demodulator options
21         configured for the BS."
22         ::= { wman2IfBsOfdmaCapabilitiesConfigEntry 2 }
23
24
25  wman2IfBsOfdmaCapCfgModulator OBJECT-TYPE
26      SYNTAX      Wman2IfOfdmaMsModType
27      MAX-ACCESS  read-only
28      STATUS      current
29      DESCRIPTION
30         "This field indicates the different modulator options
31         configured for the BS."
32         ::= { wman2IfBsOfdmaCapabilitiesConfigEntry 3 }
33
34
35  wman2IfBsOfdmaCapCfgPermutation OBJECT-TYPE
36      SYNTAX      Wman2IfOfdmaPermutation
37      MAX-ACCESS  read-only
38      STATUS      current
39      DESCRIPTION
40         "This field indicates the OFDMA MS Permutation support
41         configured for the BS."
42         ::= { wman2IfBsOfdmaCapabilitiesConfigEntry 4 }
43
44
45  wman2IfBsOfdmaCapCfgMobilityFeature OBJECT-TYPE
46      SYNTAX      Wman2IfOfdmaMobility
47      MAX-ACCESS  read-only
48      STATUS      current
49      DESCRIPTION
50         "The field indicates the mobility hand-over, Sleepmode,
51         and Idle-mode configured for the BS."
52         ::= { wman2IfBsOfdmaCapabilitiesConfigEntry 5 }
53
54
55  --
56  -- SS object group - containing tables and objects to be implemented in
57  -- the Subscriber station
58
59  --
60  -- wman2IfSsCps contain the SS Common Part Sublayer objects
61
62
63
64
65

```

```

1  --
2  wman2IfSsCps OBJECT IDENTIFIER ::= { wman2IfSsObjects 1 }
3
4  --
5
6  -- wman2IfSsConfigurationTable contains global parameters for SS
7  --
8  wman2IfSsConfigurationTable OBJECT-TYPE
9
10     SYNTAX      SEQUENCE OF Wman2IfSsConfigurationEntry
11     MAX-ACCESS  not-accessible
12     STATUS      current
13     DESCRIPTION
14         "This table contains one row for the SS system
15         parameters."
16     REFERENCE
17         "Subclause 10.1 in IEEE Std 802.16-2004"
18     ::= { wman2IfSsCps 1 }
19
20
21 wman2IfSsConfigurationEntry OBJECT-TYPE
22
23     SYNTAX      Wman2IfSsConfigurationEntry
24     MAX-ACCESS  not-accessible
25     STATUS      current
26     DESCRIPTION
27         "This table is indexed by ifIndex."
28     INDEX { ifIndex }
29     ::= { wman2IfSsConfigurationTable 1 }
30
31
32 Wman2IfSsConfigurationEntry ::= SEQUENCE {
33     wman2IfSsLostDLMapInterval      INTEGER,
34     wman2IfSsLostULMapInterval      INTEGER,
35     wman2IfSsContentionRangRetries  INTEGER,
36     wman2IfSsRequestRetries         INTEGER,
37     wman2IfSsRegRequestRetries      INTEGER,
38     wman2IfSsTftpBackoffStart       INTEGER,
39     wman2IfSsTftpBackoffEnd         INTEGER,
40     wman2IfSsTftpRequestRetries     INTEGER,
41     wman2IfSsTftpDownloadRetries    INTEGER,
42     wman2IfSsTftpWait               INTEGER,
43     wman2IfSsToDRetries             INTEGER,
44     wman2IfSsToDRetryPeriod         INTEGER,
45     wman2IfSsT1Timeout              INTEGER,
46     wman2IfSsT2Timeout              INTEGER,
47     wman2IfSsT3Timeout              INTEGER,
48     wman2IfSsT4Timeout              INTEGER,
49     wman2IfSsT6Timeout              INTEGER,
50     wman2IfSsT12Timeout             INTEGER,
51     wman2IfSsT14Timeout             INTEGER,
52     wman2IfSsT16Timeout             INTEGER,
53     wman2IfSsT18Timeout             INTEGER,
54     wman2IfSsT19Timeout             INTEGER,
55     wman2IfSsT20Timeout             INTEGER,
56     wman2IfSsT21Timeout             INTEGER,
57     wman2IfSsSBCRequestRetries      INTEGER,
58     wman2IfSsTftpCpltRetries        INTEGER,
59     wman2IfSsT26Timeout             INTEGER,
60
61
62
63
64
65

```

```

1          wman2IfSsDLManagProcTime          INTEGER}
2
3
4 wman2IfSsLostDLMapInterval OBJECT-TYPE
5     SYNTAX      INTEGER (0..600)
6     UNITS       "milliseconds"
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "Time since last received DL-MAP message before downlink
11        synchronization is considered lost in ms."
12        ::= { wman2IfSsConfigurationEntry 1 }
13
14
15 wman2IfSsLostULMapInterval OBJECT-TYPE
16     SYNTAX      INTEGER (0..600)
17     UNITS       "milliseconds"
18     MAX-ACCESS  read-write
19     STATUS      current
20     DESCRIPTION
21        "Time since last received UL-MAP message before uplink
22        synchronization is considered lost in ms."
23        ::= { wman2IfSsConfigurationEntry 2 }
24
25
26
27 wman2IfSsContentionRangRetries OBJECT-TYPE
28     SYNTAX      INTEGER (16..65535)
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32        "Number of retries on contention Ranging Requests."
33        ::= { wman2IfSsConfigurationEntry 3 }
34
35
36
37 wman2IfSsRequestRetries OBJECT-TYPE
38     SYNTAX      INTEGER (16..65535)
39     MAX-ACCESS  read-write
40     STATUS      current
41     DESCRIPTION
42        "Number of retries on bandwidth allocation requests."
43        ::= { wman2IfSsConfigurationEntry 4 }
44
45
46
47 wman2IfSsRegRequestRetries OBJECT-TYPE
48     SYNTAX      INTEGER (3..65535)
49     MAX-ACCESS  read-write
50     STATUS      current
51     DESCRIPTION
52        "Number of retries on registration requests."
53        ::= { wman2IfSsConfigurationEntry 5 }
54
55
56
57 wman2IfSsTftpBackoffStart OBJECT-TYPE
58     SYNTAX      INTEGER (1..65535)
59     UNITS       "seconds"
60     MAX-ACCESS  read-write
61     STATUS      current
62     DESCRIPTION
63        "Initial value for TFTP backoff in second."
64        ::= { wman2IfSsConfigurationEntry 6 }
65

```



```

1
2 wman2IfSsTftpBackoffEnd OBJECT-TYPE
3     SYNTAX      INTEGER (16..65535)
4     UNITS       "seconds"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "Last value for TFTP backoff in second."
9         ::= { wman2IfSsConfigurationEntry 7 }
10
11
12
13 wman2IfSsTftpRequestRetries OBJECT-TYPE
14     SYNTAX      INTEGER (16..65535)
15     MAX-ACCESS  read-write
16     STATUS      current
17     DESCRIPTION
18         "Number of retries on TFTP request."
19         ::= { wman2IfSsConfigurationEntry 8 }
20
21
22
23 wman2IfSsTftpDownloadRetries OBJECT-TYPE
24     SYNTAX      INTEGER (3..65535)
25     MAX-ACCESS  read-write
26     STATUS      current
27     DESCRIPTION
28         "Number of retries on entire TFTP downloads."
29         ::= { wman2IfSsConfigurationEntry 9 }
30
31
32
33 wman2IfSsTftpWait OBJECT-TYPE
34     SYNTAX      INTEGER (2..65535)
35     UNITS       "minutes"
36     MAX-ACCESS  read-write
37     STATUS      current
38     DESCRIPTION
39         "The duration between two consecutive Transfer
40         operational parameters (TFTP) retries in min."
41         ::= { wman2IfSsConfigurationEntry 10 }
42
43
44
45 wman2IfSsToDRetries OBJECT-TYPE
46     SYNTAX      INTEGER (3..65535)
47     MAX-ACCESS  read-write
48     STATUS      current
49     DESCRIPTION
50         "Number of Retries to establish the Time of Day."
51         ::= { wman2IfSsConfigurationEntry 11 }
52
53
54
55 wman2IfSsToDRetryPeriod OBJECT-TYPE
56     SYNTAX      INTEGER (5..65535)
57     UNITS       "minutes"
58     MAX-ACCESS  read-write
59     STATUS      current
60     DESCRIPTION
61         "The retry period to re-establish the Time of Day, as
62         describe in the network entry procedure."
63         ::= { wman2IfSsConfigurationEntry 12 }
64
65

```

```

1  wman2IfSsT1Timeout OBJECT-TYPE
2      SYNTAX          INTEGER (0..50000)
3      UNITS           "milliseconds"
4      MAX-ACCESS      read-write
5      STATUS          current
6      DESCRIPTION
7          "Wait for DCD timeout in ms."
8      ::= { wman2IfSsConfigurationEntry 13 }
9
10
11  wman2IfSsT2Timeout OBJECT-TYPE
12      SYNTAX          INTEGER (0..10000)
13      UNITS           "milliseconds"
14      MAX-ACCESS      read-write
15      STATUS          current
16      DESCRIPTION
17          "Wait for broadcast ranging timeout in ms."
18      ::= { wman2IfSsConfigurationEntry 14 }
19
20
21  wman2IfSsT3Timeout OBJECT-TYPE
22      SYNTAX          INTEGER (0..200)
23      UNITS           "milliseconds"
24      MAX-ACCESS      read-write
25      STATUS          current
26      DESCRIPTION
27          "Ranging Response reception timeout following the
28          transmission of a Ranging Request in ms."
29      ::= { wman2IfSsConfigurationEntry 15 }
30
31
32  wman2IfSsT4Timeout OBJECT-TYPE
33      SYNTAX          INTEGER (30..35)
34      UNITS           "seconds"
35      MAX-ACCESS      read-write
36      STATUS          current
37      DESCRIPTION
38          "Wait for unicast ranging opportunity. If the pending until
39          complete field was used earlier by this SS, then the value
40          of that field shall be added to this interval in second."
41      ::= { wman2IfSsConfigurationEntry 16 }
42
43
44  wman2IfSsT6Timeout OBJECT-TYPE
45      SYNTAX          INTEGER (0..3000)
46      UNITS           "milliseconds"
47      MAX-ACCESS      read-write
48      STATUS          current
49      DESCRIPTION
50          "Wait for registration response in ms."
51      ::= { wman2IfSsConfigurationEntry 17 }
52
53
54  wman2IfSsT12Timeout OBJECT-TYPE
55      SYNTAX          INTEGER (0..50000)
56      UNITS           "milliseconds"
57      MAX-ACCESS      read-write
58      STATUS          current
59      DESCRIPTION
60

```

```

1         "Wait for UCD descriptor in ms."
2         ::= { wman2IfSsConfigurationEntry 18 }
3
4
5     wman2IfSsT14Timeout OBJECT-TYPE
6         SYNTAX      INTEGER (0..200)
7         UNITS        "milliseconds"
8         MAX-ACCESS   read-write
9         STATUS       current
10        DESCRIPTION
11            "Wait for DSX-RVD Timeout in ms."
12        ::= { wman2IfSsConfigurationEntry 19 }
13
14
15    wman2IfSsT16Timeout OBJECT-TYPE
16        SYNTAX      INTEGER (10..65535)
17        UNITS        "milliseconds"
18        MAX-ACCESS   read-write
19        STATUS       current
20        DESCRIPTION
21            "wait for bandwidth request grant in ms."
22        ::= { wman2IfSsConfigurationEntry 20 }
23
24
25
26    wman2IfSsT18Timeout OBJECT-TYPE
27        SYNTAX      INTEGER (0..65535)
28        UNITS        "milliseconds"
29        MAX-ACCESS   read-write
30        STATUS       current
31        DESCRIPTION
32            "wait for SBC-RSP timeout in ms."
33        ::= { wman2IfSsConfigurationEntry 21 }
34
35
36
37    wman2IfSsT19Timeout OBJECT-TYPE
38        SYNTAX      INTEGER (0..1048575)
39        UNITS        "milliseconds"
40        MAX-ACCESS   read-write
41        STATUS       current
42        DESCRIPTION
43            "Time DL-channel remains unusable in ms."
44        ::= { wman2IfSsConfigurationEntry 22 }
45
46
47
48    wman2IfSsT20Timeout OBJECT-TYPE
49        SYNTAX      INTEGER (0..65535)
50        UNITS        "milliseconds"
51        MAX-ACCESS   read-write
52        STATUS       current
53        DESCRIPTION
54            "Time SS searches for preambles on a given channel in ms."
55        ::= { wman2IfSsConfigurationEntry 23 }
56
57
58
59    wman2IfSsT21Timeout OBJECT-TYPE
60        SYNTAX      INTEGER (0..10000)
61        UNITS        "milliseconds"
62        MAX-ACCESS   read-write
63        STATUS       current
64        DESCRIPTION
65

```

```

1         "Time SS searches for DL-MAP on a given channel in ms."
2         ::= { wman2IfSsConfigurationEntry 24 }
3
4
5 wman2IfSsSBCRequestRetries OBJECT-TYPE
6     SYNTAX      INTEGER (3..16)
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "Number of retries on SBC Request."
11        ::= { wman2IfSsConfigurationEntry 25 }
12
13
14 wman2IfSsTftpCpltRetries OBJECT-TYPE
15     SYNTAX      INTEGER (3..16)
16     MAX-ACCESS  read-write
17     STATUS      current
18     DESCRIPTION
19        "Number of retries on TFTP-CPLT."
20        ::= { wman2IfSsConfigurationEntry 26 }
21
22
23
24 wman2IfSsT26Timeout OBJECT-TYPE
25     SYNTAX      INTEGER (10..200)
26     UNITS       "milliseconds"
27     MAX-ACCESS  read-write
28     STATUS      current
29     DESCRIPTION
30        "Wait for TFTP-RSP in ms."
31        ::= { wman2IfSsConfigurationEntry 27 }
32
33
34
35 wman2IfSsDLManagProcTime OBJECT-TYPE
36     SYNTAX      INTEGER (0..200)
37     UNITS       "micro seconds"
38     MAX-ACCESS  read-write
39     STATUS      current
40     DESCRIPTION
41        "Max. time between reception of Fast Power Control
42         management message and compliance to its instructions
43         by SS in us."
44        ::= { wman2IfSsConfigurationEntry 28 }
45
46
47
48 --
49 -- Subscriber Channel Measurement Table
50 --
51
52 wman2IfSsChannelMeasurementTable OBJECT-TYPE
53     SYNTAX      SEQUENCE OF Wman2IfSsChannelMeasurementEntry
54     MAX-ACCESS  not-accessible
55     STATUS      current
56     DESCRIPTION
57        "This table contains downlink channel measurement
58         information for each SS."
59
60     REFERENCE
61        "6.3.2.3.33 in IEEE Std 802.16-2004"
62        ::= { wman2IfSsCps 2 }
63
64
65 wman2IfSsChannelMeasurementEntry OBJECT-TYPE

```

```

1      SYNTAX      Wman2IfSsChannelMeasurementEntry
2      MAX-ACCESS  not-accessible
3      STATUS      current
4      DESCRIPTION
5
6          "Each entry in the table contains RSSI and CINR
7          signal quality measurement taken from the SS. The primary
8          index is the ifIndex pointing to SS.
9          wman2IfCmnHistogramIndex is the index to histogram
10         samples. Since there is no time stamp in the table,
11         wman2IfSsHistogramIndex should be increased monotonically,
12         and wraps around when it reaches the limit.
13         When the measurement entry for a SS reaches the limit,
14         the oldest entry shall be deleted as the new entry is
15         added to the table."
16
17     INDEX      { ifIndex, wman2IfSsHistogramIndex }
18     ::= { wman2IfSsChannelMeasurementTable 1 }
19
20
21 Wman2IfSsChannelMeasurementEntry ::= SEQUENCE {
22     wman2IfSsHistogramIndex      Unsigned32,
23     wman2IfSsChannelNumber      Wman2IfChannelNumber,
24     wman2IfSsStartFrame         INTEGER,
25     wman2IfSsDuration           INTEGER,
26     wman2IfSsBasicReport        BITS,
27     wman2IfSsMeanCinrReport     INTEGER,
28     wman2IfSsStdDeviationCinrReport  INTEGER,
29     wman2IfSsMeanRssiReport     INTEGER,
30     wman2IfSsStdDeviationRssiReport  INTEGER}
31
32
33
34 wman2IfSsHistogramIndex OBJECT-TYPE
35     SYNTAX      Unsigned32 (1 .. 4294967295)
36     MAX-ACCESS  not-accessible
37     STATUS      current
38     DESCRIPTION
39
40         "wman2IfSsHistogramIndex identifies the histogram samples
41         in the table for each subscriber station."
42     ::= { wman2IfSsChannelMeasurementEntry 1 }
43
44
45 wman2IfSsChannelNumber OBJECT-TYPE
46     SYNTAX      Wman2IfChannelNumber
47     MAX-ACCESS  read-only
48     STATUS      current
49     DESCRIPTION
50
51         "Physical channel number to be reported on."
52     REFERENCE
53
54         "Subclause 8.5.1 in IEEE Std 802.16-2004"
55     ::= { wman2IfSsChannelMeasurementEntry 2 }
56
57 wman2IfSsStartFrame OBJECT-TYPE
58     SYNTAX      INTEGER (0 .. 65535)
59     MAX-ACCESS  read-only
60     STATUS      current
61     DESCRIPTION
62
63         "Frame number in which measurement for this channel
64         started."
65

```

```

1      REFERENCE
2          "Subclause 11.12 in IEEE Std 802.16-2004"
3      ::= { wman2IfSsChannelMeasurementEntry 3 }
4
5
6      wman2IfSsDuration OBJECT-TYPE
7          SYNTAX      INTEGER (0..16777215)
8          MAX-ACCESS  read-only
9          STATUS      current
10         DESCRIPTION
11             "Cumulative measurement duration on the channel in
12             multiples of Ts. For any value exceeding 0xFFFFFFFF,
13             report 0xFFFFFFFF."
14         REFERENCE
15             "Subclause 11.12 in IEEE Std 802.16-2004"
16         ::= { wman2IfSsChannelMeasurementEntry 4 }
17
18
19
20      wman2IfSsBasicReport OBJECT-TYPE
21          SYNTAX      BITS {wirelessHuman(0),
22                          unknownTransmission(1),
23                          primaryUser(2),
24                          channelNotMeasured(3)}
25          MAX-ACCESS  read-only
26          STATUS      current
27          DESCRIPTION
28              "Bit #0: WirelessHUMAN detected on the channel
29              Bit #1: Unknown transmissions detected on the channel
30              Bit #2: Primary User detected on the channel
31              Bit #3: Unmeasured. Channel not measured"
32          REFERENCE
33              "Subclause 11.12 in IEEE Std 802.16-2004"
34          ::= { wman2IfSsChannelMeasurementEntry 5 }
35
36
37
38
39      wman2IfSsMeanCinrReport OBJECT-TYPE
40          SYNTAX      INTEGER (0 .. 41)
41          UNITS        "dB"
42          MAX-ACCESS  read-only
43          STATUS      current
44          DESCRIPTION
45              "Mean CINR report."
46          REFERENCE
47              "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
48          ::= { wman2IfSsChannelMeasurementEntry 6 }
49
50
51
52
53      wman2IfSsStdDeviationCinrReport OBJECT-TYPE
54          SYNTAX      INTEGER (0 .. 41)
55          UNITS        "dB"
56          MAX-ACCESS  read-only
57          STATUS      current
58          DESCRIPTION
59              "Standard deviation CINR report."
60          REFERENCE
61              "Subclause 8.2.2 and Subclause 8.3.9 in IEEE Std 802.16-2004"
62          ::= { wman2IfSsChannelMeasurementEntry 7 }
63
64
65

```

```

1  wman2IfSsMeanRssiReport OBJECT-TYPE
2      SYNTAX          INTEGER (0 .. 83)
3      UNITS           "dBm"
4      MAX-ACCESS      read-only
5      STATUS          current
6      DESCRIPTION
7          "Mean RSSI report."
8      REFERENCE
9          "Subclause 8.2.2 and Subclause 8.3.9 in IEEE Std 802.16-2004"
10         ::= { wman2IfSsChannelMeasurementEntry 8 }
11
12
13
14  wman2IfSsStdDeviationRssiReport OBJECT-TYPE
15      SYNTAX          INTEGER (0 .. 83)
16      UNITS           "dB"
17      MAX-ACCESS      read-only
18      STATUS          current
19      DESCRIPTION
20          "Standard deviation RSSI report."
21      REFERENCE
22          "Subclause 8.2.2 and Subclause 8.3.9 in IEEE Std 802.16-2004"
23         ::= { wman2IfSsChannelMeasurementEntry 9 }
24
25
26
27  --
28  -- Subscriber station PKM group
29  -- wman2IfSsPkmObjects contain the Subscriber Station Privacy Sublayer
30  -- objects
31  --
32  --
33  wman2IfSsPkmObjects OBJECT IDENTIFIER ::= { wman2IfSsObjects 2 }
34
35
36  --
37  -- Table wman2IfSsPkmAuthTable
38  --
39  --
40  wman2IfSsPkmAuthTable OBJECT-TYPE
41      SYNTAX          SEQUENCE OF Wman2IfSsPkmAuthEntry
42      MAX-ACCESS      not-accessible
43      STATUS          current
44      DESCRIPTION
45          "This table describes the PKM attributes related
46           to the authorization for each SS wireless interface."
47         ::= { wman2IfSsPkmObjects 1 }
48
49
50  wman2IfSsPkmAuthEntry OBJECT-TYPE
51      SYNTAX          Wman2IfSsPkmAuthEntry
52      MAX-ACCESS      not-accessible
53      STATUS          current
54      DESCRIPTION
55          "Each entry contains objects describing attributes of one
56           SS wireless interface."
57      INDEX           { ifIndex }
58      ::= { wman2IfSsPkmAuthTable 1 }
59
60
61
62  Wman2IfSsPkmAuthEntry ::= SEQUENCE {
63      wman2IfSsPkmAuthState          INTEGER,
64      wman2IfSsPkmAuthKeySequenceNumber  Integer32,
65

```

```

1      wman2IfSsPkmAuthExpiresOld      DateAndTime,
2      wman2IfSsPkmAuthExpiresNew      DateAndTime,
3      wman2IfSsPkmAuthReset            TruthValue,
4      wman2IfSsPkmAuthentInfos         Counter32,
5      wman2IfSsPkmAuthRequests         Counter32,
6      wman2IfSsPkmAuthReplies          Counter32,
7      wman2IfSsPkmAuthRejects          Counter32,
8      wman2IfSsPkmAuthRejects          Counter32,
9      wman2IfSsPkmAuthInvalids         Counter32,
10     wman2IfSsPkmAuthRejectErrorCode   INTEGER,
11     wman2IfSsPkmAuthRejectErrorString SnmpAdminString,
12     wman2IfSsPkmAuthInvalidErrorCode  INTEGER,
13     wman2IfSsPkmAuthInvalidErrorString SnmpAdminString,
14     wman2IfSsPkmAuthGraceTime         Integer32,
15     wman2IfSsPkmTekGraceTime          Integer32,
16     wman2IfSsPkmAuthWaitTimeout       Integer32,
17     wman2IfSsPkmReauthWaitTimeout     Integer32,
18     wman2IfSsPkmOpWaitTimeout         Integer32,
19     wman2IfSsPkmRekeyWaitTimeout      Integer32,
20     wman2IfSsPkmAuthRejectWaitTimeout Integer32}
21
22
23
24
25 wman2IfSsPkmAuthState OBJECT-TYPE
26     SYNTAX      INTEGER {start(1),
27                     authWait(2),
28                     authorized(3),
29                     reauthWait(4),
30                     authRejectWait(5),
31                     silent(6)}
32
33     MAX-ACCESS  read-only
34     STATUS      current
35     DESCRIPTION
36         "The value of this object is the state of the SS
37         authorization FSM. The start state indicates that FSM is
38         in its initial state."
39         ::= { wman2IfSsPkmAuthEntry 1 }
40
41
42
43 wman2IfSsPkmAuthKeySequenceNumber OBJECT-TYPE
44     SYNTAX      Integer32 (0..15)
45     MAX-ACCESS  read-only
46     STATUS      current
47     DESCRIPTION
48         "The value of this object is the most recent authorization
49         key sequence number for this FSM."
50         ::= { wman2IfSsPkmAuthEntry 2 }
51
52
53
54 wman2IfSsPkmAuthExpiresOld OBJECT-TYPE
55     SYNTAX      DateAndTime
56     MAX-ACCESS  read-only
57     STATUS      current
58     DESCRIPTION
59         "The value of this object is the actual clock time for
60         expiration of the immediate predecessor of the most recent
61         authorization key for this FSM. If this FSM has only one
62         authorization key, then the value is the time of activation
63         of this FSM."
64
65

```



```

1      ::= { wman2IfSsPkmAuthEntry 3 }
2
3
4  wman2IfSsPkmAuthExpiresNew OBJECT-TYPE
5      SYNTAX      DateAndTime
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "The value of this object is the actual clock time for
10         expiration of the most recent authorization key for this
11         FSM."
12
13     ::= { wman2IfSsPkmAuthEntry 4 }
14
15  wman2IfSsPkmAuthReset OBJECT-TYPE
16      SYNTAX      TruthValue
17      MAX-ACCESS  read-write
18      STATUS      current
19      DESCRIPTION
20          "Setting this object to TRUE generates a Reauthorize event
21          in the authorization FSM. Reading this object always
22          returns FALSE."
23
24     ::= { wman2IfSsPkmAuthEntry 5 }
25
26
27  wman2IfSsPkmAuthentInfos OBJECT-TYPE
28      SYNTAX      Counter32
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32          "The value of this object is the count of times the SS has
33          transmitted an Authentication Information message."
34
35     ::= { wman2IfSsPkmAuthEntry 6 }
36
37
38  wman2IfSsPkmAuthRequests OBJECT-TYPE
39      SYNTAX      Counter32
40      MAX-ACCESS  read-only
41      STATUS      current
42      DESCRIPTION
43          "The value of this object is the count of times the SS has
44          transmitted an Authorization Request message."
45
46     ::= { wman2IfSsPkmAuthEntry 7 }
47
48
49  wman2IfSsPkmAuthReplies OBJECT-TYPE
50      SYNTAX      Counter32
51      MAX-ACCESS  read-only
52      STATUS      current
53      DESCRIPTION
54          "The value of this object is the count of times the SS has
55          received an Authorization Reply message."
56
57     ::= { wman2IfSsPkmAuthEntry 8 }
58
59
60  wman2IfSsPkmAuthRejects 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 SS has
2             received an Authorization Reject message."
3       ::= { wman2IfSsPkmAuthEntry 9 }
4
5
6   wman2IfSsPkmAuthInvalids 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 SS has
12            received an Authorization Invalid message."
13      ::= { wman2IfSsPkmAuthEntry 10 }
14
15
16   wman2IfSsPkmAuthRejectErrorCode OBJECT-TYPE
17       SYNTAX      INTEGER {none(1),
18                           unknown(2),
19                           unauthorizedSs(3),
20                           unauthorizedSaid(4),
21                           permanentAuthorizationFailure(8),
22                           timeOfDayNotAcquired(11)}
23       MAX-ACCESS  read-only
24       STATUS      current
25       DESCRIPTION
26          "The value of this object is the enumerated description of
27            the Error-Code in most recent Authorization Reject message
28            received by the SS. This has value unknown(2) if the last
29            Error-Code value was 0, and none(1) if no Authorization
30            Reject message has been received since reboot."
31      ::= { wman2IfSsPkmAuthEntry 11 }
32
33
34   wman2IfSsPkmAuthRejectErrorString OBJECT-TYPE
35       SYNTAX      SnmpAdminString (SIZE (0..128))
36       MAX-ACCESS  read-only
37       STATUS      current
38       DESCRIPTION
39          "The value of this object is the Display-String in most
40            recent Authorization Reject message received by the SS.
41            This is a zero length string if no Authorization Reject
42            message has been received since reboot."
43      ::= { wman2IfSsPkmAuthEntry 12 }
44
45
46   wman2IfSsPkmAuthInvalidErrorCode OBJECT-TYPE
47       SYNTAX      INTEGER {none(1),
48                           unknown(2),
49                           unauthorizedSs(3),
50                           unsolicited(5),
51                           invalidKeySequence(6),
52                           keyRequestAuthenticationFailure(7)}
53       MAX-ACCESS  read-only
54       STATUS      current
55       DESCRIPTION
56          "The value of this object is the enumerated description of
57            the Error-Code in most recent Authorization Invalid message
58            received by the SS. This has value unknown(2) if the last
59

```

```

1         Error-Code value was 0, and none(1) if no Authorization
2         Invalid message has been received since reboot."
3 ::= { wman2IfSsPkmAuthEntry 13 }
4
5
6 wman2IfSsPkmAuthInvalidErrorString OBJECT-TYPE
7     SYNTAX      SnmpAdminString (SIZE (0..128))
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "The value of this object is the Display-String in most
12        recent Authorization Invalid message received by the SS.
13        This is a zero length string if no Authorization Invalid
14        message has been received since reboot."
15 ::= { wman2IfSsPkmAuthEntry 14 }
16
17
18
19 wman2IfSsPkmAuthGraceTime OBJECT-TYPE
20     SYNTAX      Integer32 (300..3024000)
21     UNITS       "seconds"
22     MAX-ACCESS  read-only
23     STATUS      current
24     DESCRIPTION
25        "The value of this object is the grace time for an
26        authorization key. A SS is expected to start trying to get
27        a new authorization key beginning AuthGraceTime seconds
28        before the authorization key actually expires."
29     REFERENCE
30        "Table 343 in IEEE Std 802.16-2004"
31     DEFVAL      { 600 }
32 ::= { wman2IfSsPkmAuthEntry 15 }
33
34
35
36
37 wman2IfSsPkmTekGraceTime OBJECT-TYPE
38     SYNTAX      Integer32 (300..3024000)
39     UNITS       "seconds"
40     MAX-ACCESS  read-only
41     STATUS      current
42     DESCRIPTION
43        "The value of this object is the grace time for the TEK in
44        seconds. The SS is expected to start trying to acquire a
45        new TEK beginning TEK GraceTime seconds before the
46        expiration of the most recent TEK."
47     REFERENCE
48        "Table 343 in IEEE Std 802.16-2004"
49     DEFVAL      { 3600 }
50 ::= { wman2IfSsPkmAuthEntry 16 }
51
52
53
54
55 wman2IfSsPkmAuthWaitTimeout OBJECT-TYPE
56     SYNTAX      Integer32 (2..30)
57     UNITS       "seconds"
58     MAX-ACCESS  read-only
59     STATUS      current
60     DESCRIPTION
61        "The value of this object is the Authorize Wait Timeout."
62     REFERENCE
63        "Table 343 in IEEE Std 802.16-2004"
64
65

```

```

1      DEFVAL      { 10 }
2      ::= { wman2IfSsPkmAuthEntry 17 }
3
4
5  wman2IfSsPkmReauthWaitTimeout OBJECT-TYPE
6      SYNTAX      Integer32 (2..30)
7      UNITS       "seconds"
8      MAX-ACCESS  read-only
9      STATUS      current
10     DESCRIPTION
11         "The value of this object is the Reauthorize Wait Timeout
12         in seconds."
13     REFERENCE
14         "Table 343 in IEEE Std 802.16-2004"
15     DEFVAL      { 10 }
16     ::= { wman2IfSsPkmAuthEntry 18 }
17
18
19
20 wman2IfSsPkmOpWaitTimeout OBJECT-TYPE
21     SYNTAX      Integer32 (1..10)
22     UNITS       "seconds"
23     MAX-ACCESS  read-only
24     STATUS      current
25     DESCRIPTION
26         "The value of this object is the Operational Wait Timeout
27         in seconds."
28     REFERENCE
29         "Table 343 in IEEE Std 802.16-2004"
30     DEFVAL      { 1 }
31     ::= { wman2IfSsPkmAuthEntry 19 }
32
33
34
35
36 wman2IfSsPkmRekeyWaitTimeout OBJECT-TYPE
37     SYNTAX      Integer32 (1..10)
38     UNITS       "seconds"
39     MAX-ACCESS  read-only
40     STATUS      current
41     DESCRIPTION
42         "The value of this object is the Rekey Wait Timeout in
43         seconds."
44     REFERENCE
45         "Table 343 in IEEE Std 802.16-2004"
46     DEFVAL      { 1 }
47     ::= { wman2IfSsPkmAuthEntry 20 }
48
49
50
51
52 wman2IfSsPkmAuthRejectWaitTimeout OBJECT-TYPE
53     SYNTAX      Integer32 (10..600)
54     UNITS       "seconds"
55     MAX-ACCESS  read-only
56     STATUS      current
57     DESCRIPTION
58         "The value of this object is the Authorization Reject Wait
59         Timeout in seconds."
60     REFERENCE
61         "Table 343 in IEEE Std 802.16-2004"
62     DEFVAL      { 60 }
63     ::= { wman2IfSsPkmAuthEntry 21 }
64
65

```

```

1
2  --
3  -- Table wman2IfSsPkmTekTable
4  --
5
6  wman2IfSsPkmTekTable OBJECT-TYPE
7      SYNTAX          SEQUENCE OF Wman2IfSsPkmTekEntry
8      MAX-ACCESS      not-accessible
9      STATUS          current
10     DESCRIPTION
11         "This table describes the attributes of each SS Traffic
12         Encryption Key (TEK) association. The SS maintains (no more
13         than) one TEK association per SAID per SS wireless
14         interface."
15     ::= { wman2IfSsPkmObjects 2 }
16
17
18
19  wman2IfSsPkmTekEntry OBJECT-TYPE
20      SYNTAX          Wman2IfSsPkmTekEntry
21      MAX-ACCESS      not-accessible
22      STATUS          current
23      DESCRIPTION
24         "Each entry contains objects describing the TEK association
25         attributes of one SAID. The SS MUST create one entry per
26         SAID, regardless of whether the SAID was obtained from a
27         Registration Response message, from an Authorization Reply
28         message, or from any dynamic SAID establishment
29         mechanisms."
30     INDEX            { ifIndex, wman2IfSsPkmTekSAID }
31     ::= { wman2IfSsPkmTekTable 1 }
32
33
34
35
36  Wman2IfSsPkmTekEntry ::= SEQUENCE {
37      wman2IfSsPkmTekSAID                INTEGER,
38      wman2IfSsPkmTekSAType               INTEGER,
39      wman2IfSsPkmTekDataEncryptAlg       Wman2IfDataEncryptAlgId,
40      wman2IfSsPkmTekDataAuthAlg          Wman2IfDataAuthAlgId,
41      wman2IfSsPkmTekEncryptAlg           Wman2IfTekEncryptAlgId,
42      wman2IfSsPkmTekState                INTEGER,
43      wman2IfSsPkmTekKeySequenceNumber    Integer32,
44      wman2IfSsPkmTekExpiresOld            DateAndTime,
45      wman2IfSsPkmTekExpiresNew            DateAndTime,
46      wman2IfSsPkmTekKeyRequests           Counter32,
47      wman2IfSsPkmTekKeyReplies            Counter32,
48      wman2IfSsPkmTekKeyRejects            Counter32,
49      wman2IfSsPkmTekInvalids              Counter32,
50      wman2IfSsPkmTekAuthPends             Counter32,
51      wman2IfSsPkmTekKeyRejectErrorCode    INTEGER,
52      wman2IfSsPkmTekKeyRejectErrorString  SnmpAdminString,
53      wman2IfSsPkmTekInvalidErrorCode      INTEGER,
54      wman2IfSsPkmTekInvalidErrorString    SnmpAdminString}
55
56
57
58
59
60  wman2IfSsPkmTekSAID OBJECT-TYPE
61      SYNTAX          INTEGER (0..65535)
62      MAX-ACCESS      not-accessible
63      STATUS          current
64      DESCRIPTION
65

```

```

1         "The value of this object is the Security Association
2         ID (SAID)."
```

::= { wman2IfSsPkmTekEntry 1 }

```

5
6 wman2IfSsPkmTekSAType OBJECT-TYPE
7     SYNTAX      INTEGER {primarySA(0),
8                   staticSA(1),
9                   dynamicSA(2)}
10
11     MAX-ACCESS  read-only
12     STATUS      current
13     DESCRIPTION
14         "The value of this object is the type of security
15         association."
```

REFERENCE

```

17         "IEEE Std 802.16-2004; 11.9.18"
18     ::= { wman2IfSsPkmTekEntry 2 }
```

::= { wman2IfSsPkmTekEntry 2 }

```

20
21 wman2IfSsPkmTekDataEncryptAlg OBJECT-TYPE
22     SYNTAX      Wman2IfDataEncryptAlgId
23     MAX-ACCESS  read-only
24     STATUS      current
25     DESCRIPTION
26         "The value of this object is the data encryption algorithm
27         being utilized."
```

REFERENCE

```

29         "Table 375, IEEE Std 802.16-2004"
30     ::= { wman2IfSsPkmTekEntry 3 }
```

::= { wman2IfSsPkmTekEntry 3 }

```

33
34 wman2IfSsPkmTekDataAuthentAlg OBJECT-TYPE
35     SYNTAX      Wman2IfDataAuthAlgId
36     MAX-ACCESS  read-only
37     STATUS      current
38     DESCRIPTION
39         "The value of this object is the data authentication
40         algorithm being utilized."
```

REFERENCE

```

42         "Table 376, IEEE Std 802.16-2004"
43     ::= { wman2IfSsPkmTekEntry 4 }
```

::= { wman2IfSsPkmTekEntry 4 }

```

46
47 wman2IfSsPkmTekEncryptAlg OBJECT-TYPE
48     SYNTAX      Wman2IfTekEncryptAlgId
49     MAX-ACCESS  read-only
50     STATUS      current
51     DESCRIPTION
52         "The value of this object is the TEK key encryption
53         algorithm for this cryptographic suite capability."
```

REFERENCE

```

55         "Table 377, IEEE Std 802.16-2004"
56     ::= { wman2IfSsPkmTekEntry 5 }
```

::= { wman2IfSsPkmTekEntry 5 }

```

59
60 wman2IfSsPkmTekState OBJECT-TYPE
61     SYNTAX      INTEGER {start(1),
62                           opWait(2),
63                           opReauthWait(3),
64
65
```

```

1          operational(4),
2          rekeyWait(5),
3          rekeyReauthWait(6)}
4
5      MAX-ACCESS    read-only
6      STATUS        current
7      DESCRIPTION
8          "The value of this object is the state of the indicated TEK
9          FSM. The start(1) state indicates that FSM is in its
10         initial state."
11
12     ::= { wman2IfSsPkmTekEntry 6 }
13
14 wman2IfSsPkmTekKeySequenceNumber OBJECT-TYPE
15     SYNTAX          Integer32 (0..3)
16     MAX-ACCESS      read-only
17     STATUS          current
18     DESCRIPTION
19         "The value of this object is the most recent TEK key
20         sequence number for this TEK FSM."
21
22     REFERENCE
23         "IEEE Std 802.16-2004; 11.9.5"
24
25     ::= { wman2IfSsPkmTekEntry 7 }
26
27 wman2IfSsPkmTekExpiresOld OBJECT-TYPE
28     SYNTAX          DateAndTime
29     MAX-ACCESS      read-only
30     STATUS          current
31     DESCRIPTION
32         "The value of this object is the actual clock time for
33         expiration of the immediate predecessor of the most recent
34         TEK for this FSM. If this FSM has only one TEK, then the
35         value is the time of activation of this FSM."
36
37     ::= { wman2IfSsPkmTekEntry 8 }
38
39
40 wman2IfSsPkmTekExpiresNew OBJECT-TYPE
41     SYNTAX          DateAndTime
42     MAX-ACCESS      read-only
43     STATUS          current
44     DESCRIPTION
45         "The value of this object is the actual clock time for
46         expiration of the most recent TEK for this FSM."
47
48     ::= { wman2IfSsPkmTekEntry 9 }
49
50
51 wman2IfSsPkmTekKeyRequests OBJECT-TYPE
52     SYNTAX          Counter32
53     MAX-ACCESS      read-only
54     STATUS          current
55     DESCRIPTION
56         "The value of this object is the count of times the SS has
57         transmitted a Key Request message."
58
59     ::= { wman2IfSsPkmTekEntry 10 }
60
61
62 wman2IfSsPkmTekKeyReplies OBJECT-TYPE
63     SYNTAX          Counter32
64     MAX-ACCESS      read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "The value of this object is the count of times the SS has
4              received a Key Reply message, including a message whose
5              authentication failed."
6      ::= { wman2IfSsPkmTekEntry 11 }
7
8
9
10     wman2IfSsPkmTekKeyRejects OBJECT-TYPE
11         SYNTAX      Counter32
12         MAX-ACCESS  read-only
13         STATUS      current
14         DESCRIPTION
15             "The value of this object is the count of times the SS has
16                 received a Key Reject message, including a message whose
17                 authentication failed."
18         ::= { wman2IfSsPkmTekEntry 12 }
19
20
21
22     wman2IfSsPkmTekInvalids OBJECT-TYPE
23         SYNTAX      Counter32
24         MAX-ACCESS  read-only
25         STATUS      current
26         DESCRIPTION
27             "The value of this object is the count of times the SS has
28                 received a TEK Invalid message, including a message whose
29                 authentication failed."
30         ::= { wman2IfSsPkmTekEntry 13 }
31
32
33
34     wman2IfSsPkmTekAuthPends OBJECT-TYPE
35         SYNTAX      Counter32
36         MAX-ACCESS  read-only
37         STATUS      current
38         DESCRIPTION
39             "The value of this object is the count of times an
40                 Authorization Pending (Auth Pend) event occurred in this
41                 FSM."
42         ::= { wman2IfSsPkmTekEntry 14 }
43
44
45
46     wman2IfSsPkmTekKeyRejectErrorCode OBJECT-TYPE
47         SYNTAX      INTEGER {none(1),
48                               unknown(2),
49                               unauthorizedSaid(4)}
50         MAX-ACCESS  read-only
51         STATUS      current
52         DESCRIPTION
53             "The value of this object is the enumerated description of
54                 the Error-Code in most recent Key Reject message received
55                 by the SS. This has value unknown(2) if the last Error-Code
56                 value was 0, and none(1) if no Key Reject message has been
57                 received since reboot."
58         ::= { wman2IfSsPkmTekEntry 15 }
59
60
61
62     wman2IfSsPkmTekKeyRejectErrorString OBJECT-TYPE
63         SYNTAX      SnmpAdminString (SIZE (0..128))
64         MAX-ACCESS  read-only
65

```



```

1      STATUS      current
2      DESCRIPTION
3          "The value of this object is the Display-String in most
4          recent Key Reject message received by the SS. This is a
5          zero length string if no Key Reject message has been
6          received since reboot."
7
8      ::= { wman2IfSsPkmTekEntry 16 }
9
10
11 wman2IfSsPkmTekInvalidErrorCode OBJECT-TYPE
12     SYNTAX      INTEGER {none(1),
13                     unknown(2),
14                     invalidKeySequence(6)}
15     MAX-ACCESS  read-only
16     STATUS      current
17     DESCRIPTION
18         "The value of this object is the enumerated description of
19         the Error-Code in most recent TEK Invalid message received
20         by the SS. This has value unknown(2) if the last
21         Error-Code value was 0, and none(1) if no TEK Invalid
22         message has been received since reboot."
23
24     ::= { wman2IfSsPkmTekEntry 17 }
25
26
27 wman2IfSsPkmTekInvalidErrorString OBJECT-TYPE
28     SYNTAX      SnmpAdminString (SIZE (0..128))
29     MAX-ACCESS  read-only
30     STATUS      current
31     DESCRIPTION
32         "The value of this object is the Display-String in most
33         recent TEK Invalid message received by the SS. This is a
34         zero length string if no TEK Invalid message has been
35         received since reboot."
36
37     ::= { wman2IfSsPkmTekEntry 18 }
38
39
40
41 --
42 -- Table wman2IfSsDeviceCertTable
43 --
44 wman2IfSsDeviceCertTable OBJECT-TYPE
45     SYNTAX      SEQUENCE OF Wman2IfSsDeviceCertEntry
46     MAX-ACCESS  not-accessible
47     STATUS      current
48     DESCRIPTION
49         "This table describes the PKM device certificates for each
50         SS wireless interface."
51
52     ::= { wman2IfSsPkmObjects 3 }
53
54
55 wman2IfSsDeviceCertEntry OBJECT-TYPE
56     SYNTAX      Wman2IfSsDeviceCertEntry
57     MAX-ACCESS  not-accessible
58     STATUS      current
59     DESCRIPTION
60         "Each entry contains the device certificate of one SS."
61
62     INDEX      { ifIndex }
63     ::= { wman2IfSsDeviceCertTable 1 }
64
65

```

```

1  Wman2IfSsDeviceCertEntry ::= SEQUENCE {
2      wman2IfSsDeviceCert          OCTET STRING,
3      wman2IfSsDeviceManufCert     OCTET STRING}
4
5
6  wman2IfSsDeviceCert OBJECT-TYPE
7      SYNTAX          OCTET STRING (SIZE(0..65535))
8      MAX-ACCESS      read-only
9      STATUS          current
10     DESCRIPTION
11         "The X509 DER-encoded subscriber station certificate."
12     ::= { wman2IfSsDeviceCertEntry 1 }
13
14
15  wman2IfSsDeviceManufCert OBJECT-TYPE
16      SYNTAX          OCTET STRING (SIZE(0..65535))
17      MAX-ACCESS      read-only
18      STATUS          current
19      DESCRIPTION
20         "The X509 DER-encoded manufacturer certificate which is
21         signed by the CA root authority certificate."
22     ::= { wman2IfSsDeviceCertEntry 2 }
23
24
25
26  --
27  -- Subscriber station Notification Group
28  -- wman2IfSsNotificationObjects contains the SS SNMP Trap objects
29  --
30
31  wman2IfSsNotification OBJECT IDENTIFIER ::= { wman2IfSsObjects 3 }
32  wman2IfSsTrapControl OBJECT IDENTIFIER ::= { wman2IfSsNotification 1 }
33  wman2IfSsTrapDefinitions OBJECT IDENTIFIER ::= { wman2IfSsNotification 2 }
34  }
35
36
37  -- This object groups all NOTIFICATION-TYPE objects for SS.
38  -- It is defined following RFC2758 sections 8.5 and 8.6
39  -- for the compatibility with SNMPv1.
40
41  wman2IfSsTrapPrefix OBJECT IDENTIFIER ::= { wman2IfSsTrapDefinitions 0 }
42
43
44  wman2IfSsTrapControlRegister OBJECT-TYPE
45      SYNTAX          BITS {wman2IfSsTlvUnknown(0),
46                          wman2IfSsDynamicServiceFail(1),
47                          wman2IfSsDhcpSuccess(2),
48                          wman2IfSsRssiStatusChange(3)}
49      MAX-ACCESS      read-write
50      STATUS          current
51      DESCRIPTION
52         "The object is used to enable Subscriber Station traps.
53         From left to right, the set bit indicates the corresponding
54         Subscriber Station trap is enabled."
55     ::= { wman2IfSsTrapControl 1 }
56
57
58
59  wman2IfSsThresholdConfigTable OBJECT-TYPE
60      SYNTAX          SEQUENCE OF Wman2IfSsThresholdConfigEntry
61      MAX-ACCESS      not-accessible
62      STATUS          current
63      DESCRIPTION
64         "This table contains threshold objects that can be set to
65

```

```

1         detect the threshold crossing events."
2         ::= { wman2IfSsTrapControl 2 }
3
4
5 wman2IfSsThresholdConfigEntry OBJECT-TYPE
6     SYNTAX      Wman2IfSsThresholdConfigEntry
7     MAX-ACCESS  not-accessible
8     STATUS      current
9     DESCRIPTION
10        "This table provides one row for each Ss, and is indexed
11        by ifIndex."
12    INDEX        { ifIndex }
13    ::= { wman2IfSsThresholdConfigTable 1 }
14
15
16 Wman2IfSsThresholdConfigEntry ::= SEQUENCE {
17     wman2IfSsRssiLowThreshold      Integer32,
18     wman2IfSsRssiHighThreshold    Integer32}
19
20
21 wman2IfSsRssiLowThreshold OBJECT-TYPE
22     SYNTAX      Integer32
23     UNITS       "dBm"
24     MAX-ACCESS  read-write
25     STATUS      current
26     DESCRIPTION
27        "Low RSSI threshold for generating the RSSI alarm trap."
28    ::= { wman2IfSsThresholdConfigEntry 1 }
29
30
31
32 wman2IfSsRssiHighThreshold OBJECT-TYPE
33     SYNTAX      Integer32
34     UNITS       "dBm"
35     MAX-ACCESS  read-write
36     STATUS      current
37     DESCRIPTION
38        "High RSSI threshold for generating a trap to indicate
39        the RSSI is restored."
40    ::= { wman2IfSsThresholdConfigEntry 2 }
41
42
43
44 wman2IfSsTlvUnknownTrap NOTIFICATION-TYPE
45     OBJECTS      {ifIndex,
46                  wman2IfSsMacAddress,
47                  wman2IfSsUnknownTlv}
48     STATUS      current
49     DESCRIPTION
50        "Event that notifies detection of unknown TLV during
51        the TLV parsing process."
52    ::= { wman2IfSsTrapPrefix 1 }
53
54
55
56 wman2IfSsDynamicServiceFailTrap NOTIFICATION-TYPE
57     OBJECTS      {ifIndex,
58                  wman2IfSsMacAddress,
59                  wman2IfSsDynamicServiceType,
60                  wman2IfSsDynamicServiceFailReason}
61     STATUS      current
62     DESCRIPTION
63        "An event to report the failure of a dynamic service
64
65

```

```

1         operation happened during the dynamic services process
2         and detected in the BS side."
3     ::= { wman2IfSsTrapPrefix 2 }
4
5
6 wman2IfSsDhcpSuccessTrap    NOTIFICATION-TYPE
7     OBJECTS      {ifIndex,
8                   wman2IfSsMacAddress}
9
10    STATUS        current
11    DESCRIPTION
12        "An event to report a successful Handshake to establish IP
13        connectivity."
14    ::= { wman2IfSsTrapPrefix 3 }
15
16
17 wman2IfSsRssiStatusChangeTrap NOTIFICATION-TYPE
18     OBJECTS      {ifIndex,
19                   wman2IfSsMacAddress,
20                   wman2IfSsRssiStatus,
21                   wman2IfSsRssiStatusInfo}
22
23     STATUS        current
24     DESCRIPTION
25         "An event to report that the downlink RSSI is below
26         wman2IfSsRssiLowThreshold, or above
27         wman2IfSsRssiHighThreshold after restore."
28     ::= { wman2IfSsTrapPrefix 4 }
29
30
31 wman2IfSsNotificationObjectsTable OBJECT-TYPE
32     SYNTAX        SEQUENCE OF Wman2IfSsNotificationObjectsEntry
33     MAX-ACCESS    not-accessible
34     STATUS        current
35     DESCRIPTION
36         "This table contains SS notification objects that have been
37         reported by the trap."
38     ::= { wman2IfSsTrapDefinitions 1 }
39
40
41
42 wman2IfSsNotificationObjectsEntry OBJECT-TYPE
43     SYNTAX        Wman2IfSsNotificationObjectsEntry
44     MAX-ACCESS    not-accessible
45     STATUS        current
46     DESCRIPTION
47         "This table provides one row for each SS that has
48         generated traps, and is indexed by ifIndex."
49     INDEX          { ifIndex }
50     ::= { wman2IfSsNotificationObjectsTable 1 }
51
52
53
54 Wman2IfSsNotificationObjectsEntry ::= SEQUENCE {
55     wman2IfSsMacAddress      MacAddress,
56     wman2IfSsUnknownTlv     OCTET STRING,
57     wman2IfSsDynamicServiceType INTEGER,
58     wman2IfSsDynamicServiceFailReason OCTET STRING,
59     wman2IfSsRssiStatus      INTEGER,
60     wman2IfSsRssiStatusInfo  OCTET STRING}
61
62
63 wman2IfSsMacAddress OBJECT-TYPE
64     SYNTAX        MacAddress
65

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "The MAC address of the SS generating the trap."
5      ::= { wman2IfSsNotificationObjectsEntry 1 }
6
7
8      wman2IfSsUnknownTlv    OBJECT-TYPE
9          SYNTAX      OCTET STRING (SIZE(0..65535))
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "Indicating the value of the unknown TLV."
14         ::= { wman2IfSsNotificationObjectsEntry 2 }
15
16
17      wman2IfSsDynamicServiceType    OBJECT-TYPE
18          SYNTAX      INTEGER {ssSfCreationReq(1),
19                               ssSfCreationRsp(2),
20                               ssSfCreationAck(3)}
21         MAX-ACCESS  read-only
22         STATUS      current
23         DESCRIPTION
24             "This object indicates the dynamic service flow
25             creation command type."
26         ::= { wman2IfSsNotificationObjectsEntry 3 }
27
28
29      wman2IfSsDynamicServiceFailReason    OBJECT-TYPE
30          SYNTAX      OCTET STRING (SIZE(0..255))
31         MAX-ACCESS  read-only
32         STATUS      current
33         DESCRIPTION
34             "This object indicates the reason why the service flow
35             creation has failed."
36         ::= { wman2IfSsNotificationObjectsEntry 4 }
37
38
39      wman2IfSsRssiStatus    OBJECT-TYPE
40          SYNTAX      INTEGER {ssRssiAlarm(1),
41                               ssRssiNoAlarm(2)}
42         MAX-ACCESS  read-only
43         STATUS      current
44         DESCRIPTION
45             "A RSSI alarm is generated if the RSSI is lower than
46             wman2IfSsRssiLowThreshold, or above
47             wman2IfSsRssiHighThreshold after alarm is restored."
48         ::= { wman2IfSsNotificationObjectsEntry 5 }
49
50
51      wman2IfSsRssiStatusInfo    OBJECT-TYPE
52          SYNTAX      OCTET STRING (SIZE(0..255))
53         MAX-ACCESS  read-only
54         STATUS      current
55         DESCRIPTION
56             "This object provides additional information about RSSI
57             alarm. It is implementation specific"
58         ::= { wman2IfSsNotificationObjectsEntry 6 }
59
60
61
62
63
64
65

```

```

1  --
2  -- Subscriber station PHY Group
3  --
4  wman2IfSsPhy OBJECT IDENTIFIER ::= { wman2IfSsObjects 5 }
5
6  --
7  --
8  -- SS OFDM PHY objects
9  --
10
11 wman2IfSsOfdmPhy OBJECT IDENTIFIER ::= { wman2IfSsPhy 1 }
12
13 wman2IfSsOfdmUplinkChannelTable OBJECT-TYPE
14     SYNTAX      SEQUENCE OF Wman2IfSsOfdmUplinkChannelEntry
15     MAX-ACCESS  not-accessible
16     STATUS      current
17     DESCRIPTION
18         "This table contains UCD channel attributes, defining the
19         transmission characteristics of uplink channels"
20     REFERENCE
21         "Table 349 and Table 352, in IEEE Std 802.16-2004"
22     ::= { wman2IfSsOfdmPhy 1 }
23
24
25
26 wman2IfSsOfdmUplinkChannelEntry OBJECT-TYPE
27     SYNTAX      Wman2IfSsOfdmUplinkChannelEntry
28     MAX-ACCESS  not-accessible
29     STATUS      current
30     DESCRIPTION
31         "This table provides one row for each uplink channel of
32         multi-sector BS, and is indexed by BS ifIndex. An entry
33         in this table exists for each ifEntry of BS with an
34         ifType of propBWAp2Mp."
35     INDEX { ifIndex }
36     ::= { wman2IfSsOfdmUplinkChannelTable 1 }
37
38
39
40
41 Wman2IfSsOfdmUplinkChannelEntry ::= SEQUENCE {
42     wman2IfSsOfdmCtBasedResvTimeout      INTEGER,
43     wman2IfSsOfdmBwReqOppSize             INTEGER,
44     wman2IfSsOfdmRangReqOppSize           INTEGER,
45     wman2IfSsOfdmUplinkCenterFreq         Unsigned32,
46     wman2IfSsOfdmNumSubChReqRegionFull    INTEGER,
47     wman2IfSsOfdmNumSymbolsReqRegionFull  INTEGER,
48     wman2IfSsOfdmSubChFocusCtCode         INTEGER,
49     wman2IfSsOfdmUpLinkChannelId         INTEGER}
50
51
52
53 wman2IfSsOfdmCtBasedResvTimeout OBJECT-TYPE
54     SYNTAX      INTEGER (1..255)
55     MAX-ACCESS  read-only
56     STATUS      current
57     DESCRIPTION
58         "The number of UL-MAPs to receive before contention-based
59         reservation is attempted again for the same connection."
60     REFERENCE
61         "Table 349, in IEEE Std 802.16-2004"
62     ::= { wman2IfSsOfdmUplinkChannelEntry 1 }
63
64
65

```

```

1  wman2IfSsOfdmBwReqOppSize OBJECT-TYPE
2      SYNTAX      INTEGER (1..65535)
3      UNITS       "PS"
4      MAX-ACCESS  read-only
5      STATUS      current
6      DESCRIPTION
7          "Size (in units of PS) of PHY payload that SS may use to
8          format and transmit a bandwidth request message in a
9          contention request opportunity. The value includes all
10         PHY overhead as well as allowance for the MAC data the
11         message may hold."
12
13     REFERENCE
14         "Table 349, in IEEE Std 802.16-2004"
15     ::= { wman2IfSsOfdmUplinkChannelEntry 2 }
16
17  wman2IfSsOfdmRangReqOppSize OBJECT-TYPE
18      SYNTAX      INTEGER (1..65535)
19      UNITS       "PS"
20      MAX-ACCESS  read-only
21      STATUS      current
22      DESCRIPTION
23          "Size (in units of PS) of PHY payload that SS may use to
24          format and transmit a RNG-REQ message in a contention
25          request opportunity. The value includes all PHY overhead
26          as well as allowance for the MAC data the message may
27          hold and the maximum SS/BS roundtrip propagation delay."
28
29     REFERENCE
30         "Table 349, in IEEE Std 802.16-2004"
31     ::= { wman2IfSsOfdmUplinkChannelEntry 3 }
32
33  wman2IfSsOfdmUplinkCenterFreq OBJECT-TYPE
34      SYNTAX      Unsigned32
35      UNITS       "kHz"
36      MAX-ACCESS  read-only
37      STATUS      current
38      DESCRIPTION
39          " Uplink center frequency (kHz)"
40
41     REFERENCE
42         "Table 349, in IEEE Std 802.16-2004"
43     ::= { wman2IfSsOfdmUplinkChannelEntry 4 }
44
45  wman2IfSsOfdmNumSubChReqRegionFull OBJECT-TYPE
46      SYNTAX      INTEGER {oneSubchannel(0),
47                          twoSubchannels(1),
48                          fourSubchannels(2),
49                          eightSubchannels(3),
50                          sixteenSubchannels(4)}
51      MAX-ACCESS  read-only
52      STATUS      current
53      DESCRIPTION
54          "Number of subchannels used by each transmit
55          opportunity when REQ Region-Full is allocated in
56          subchannelization region."
57
58     REFERENCE
59
60
61
62
63
64
65

```

```

1         "Table 352, in IEEE Std 802.16-2004"
2         ::= { wman2IfSsOfdmUplinkChannelEntry 5 }
3
4
5 wman2IfSsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
6     SYNTAX      INTEGER (0..31)
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "Number of OFDM symbols used by each transmit
11        opportunity when REQ Region-Full is allocated in
12        subchannelization region."
13
14     REFERENCE
15        "Table 352, in IEEE Std 802.16-2004"
16        ::= { wman2IfSsOfdmUplinkChannelEntry 6 }
17
18
19 wman2IfSsOfdmSubChFocusCtCode OBJECT-TYPE
20     SYNTAX      INTEGER (0..8)
21     MAX-ACCESS  read-only
22     STATUS      current
23     DESCRIPTION
24        "Number of contention codes (CSE) that shall only be used to
25        request a subchannelized allocation. Default value 0.
26        Allowed values 0-8."
27
28     REFERENCE
29        "Table 352, in IEEE Std 802.16-2004"
30
31     DEFVAL      { 0 }
32     ::= { wman2IfSsOfdmUplinkChannelEntry 7 }
33
34
35 wman2IfSsOfdmUpLinkChannelId OBJECT-TYPE
36     SYNTAX      INTEGER (0..255)
37     MAX-ACCESS  read-only
38     STATUS      current
39     DESCRIPTION
40        "The identifier of the uplink channel to which this
41        message refers."
42
43     REFERENCE
44        "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
45        ::= { wman2IfSsOfdmUplinkChannelEntry 8 }
46
47
48 wman2IfSsOfdmDownlinkChannelTable OBJECT-TYPE
49     SYNTAX      SEQUENCE OF Wman2IfSsOfdmDownlinkChannelEntry
50     MAX-ACCESS  not-accessible
51     STATUS      current
52     DESCRIPTION
53        "This table contains DCD channel attributes, defining the
54        transmission characteristics of downlink channels"
55
56     REFERENCE
57        "Table 358, in IEEE Std 802.16-2004"
58        ::= { wman2IfSsOfdmPhy 2 }
59
60
61 wman2IfSsOfdmDownlinkChannelEntry OBJECT-TYPE
62     SYNTAX      Wman2IfSsOfdmDownlinkChannelEntry
63     MAX-ACCESS  not-accessible
64     STATUS      current
65

```



```

1      DESCRIPTION
2          "This table provides one row for each downlink channel of
3          multi-sector BS, and is indexed by BS ifIndex. An entry
4          in this table exists for each ifEntry of BS with an
5          ifType of propBWA2Mp."
6      INDEX { ifIndex }
7      ::= { wman2IfSsOfdmDownlinkChannelTable 1 }
8
9
10
11  Wman2IfSsOfdmDownlinkChannelEntry ::= SEQUENCE {
12      wman2IfSsOfdmBsEIRP                INTEGER,
13      wman2IfSsOfdmChannelNumber          Wman2IfChannelNumber,
14      wman2IfSsOfdmTTG                    INTEGER,
15      wman2IfSsOfdmRTG                    INTEGER,
16      wman2IfSsOfdmInitRngMaxRSS          INTEGER,
17      wman2IfSsOfdmDownlinkCenterFreq     Unsigned32,
18      wman2IfSsOfdmBsId                   Wman2IfBsIdType,
19      wman2IfSsOfdmMacVersion              Wman2IfMacVersion,
20      wman2IfSsOfdmFrameDurationCode      INTEGER,
21      wman2IfSsOfdmDownLinkChannelId      INTEGER}
22
23
24
25  wman2IfSsOfdmBsEIRP OBJECT-TYPE
26      SYNTAX      INTEGER (-32768..32767)
27      UNITS       "dBm"
28      MAX-ACCESS  read-only
29      STATUS      current
30      DESCRIPTION
31          "The EIRP is the equivalent isotropic radiated power of
32          the base station, which is computed for a simple
33          single-antenna transmitter."
34      REFERENCE
35          "Table 358, in IEEE Std 802.16-2004"
36      ::= { wman2IfSsOfdmDownlinkChannelEntry 1 }
37
38
39
40
41  wman2IfSsOfdmChannelNumber OBJECT-TYPE
42      SYNTAX      Wman2IfChannelNumber
43      MAX-ACCESS  read-only
44      STATUS      current
45      DESCRIPTION
46          "Downlink channel number as defined in 8.5.
47          Used for license-exempt operation only."
48      REFERENCE
49          "Table 358, in IEEE Std 802.16-2004"
50      ::= { wman2IfSsOfdmDownlinkChannelEntry 2 }
51
52
53
54  wman2IfSsOfdmTTG OBJECT-TYPE
55      SYNTAX      INTEGER (0..255)
56      MAX-ACCESS  read-only
57      STATUS      current
58      DESCRIPTION
59          "Transmit / Receive Transition Gap."
60      REFERENCE
61          "Table 358, in IEEE Std 802.16-2004"
62      ::= { wman2IfSsOfdmDownlinkChannelEntry 3 }
63
64
65

```

```

1  wman2IfSsOfdmRTG OBJECT-TYPE
2      SYNTAX          INTEGER (0..255)
3      MAX-ACCESS      read-only
4      STATUS          current
5      DESCRIPTION
6          "Receive / Transmit Transition Gap."
7      REFERENCE
8          "Table 358, in IEEE Std 802.16-2004"
9      ::= { wman2IfSsOfdmDownlinkChannelEntry 4 }
10
11
12
13  wman2IfSsOfdmInitRngMaxRSS OBJECT-TYPE
14      SYNTAX          INTEGER (-32768..32767)
15      UNITS            "dBm"
16      MAX-ACCESS      read-only
17      STATUS          current
18      DESCRIPTION
19          "Initial Ranging Max. equivalent isotropic received power
20           at BS Signed in units of 1 dBm."
21      REFERENCE
22          "Table 358, in IEEE Std 802.16-2004"
23      ::= { wman2IfSsOfdmDownlinkChannelEntry 5 }
24
25
26
27  wman2IfSsOfdmDownlinkCenterFreq OBJECT-TYPE
28      SYNTAX          Unsigned32
29      UNITS            "kHz"
30      MAX-ACCESS      read-only
31      STATUS          current
32      DESCRIPTION
33          "Downlink center frequency (kHz)."

```

```

1  wman2IfSsOfdmFrameDurationCode OBJECT-TYPE
2      SYNTAX      INTEGER {duration2dot5ms(0),
3                      duration4ms(1),
4                      duration5ms(2),
5                      duration8ms(3),
6                      duration10ms(4),
7                      duration12dot5ms(5),
8                      duration20ms(6)}
9
10     MAX-ACCESS   read-only
11     STATUS       current
12     DESCRIPTION
13         "The duration of the frame. The frame duration code
14         values are specified in Table 230."
15     REFERENCE
16         "Subclause 11.4.1, Table 230, in IEEE Std 802.16-2004"
17     ::= { wman2IfSsOfdmDownlinkChannelEntry 9 }
18
19  wman2IfSsOfdmDownLinkChannelId OBJECT-TYPE
20
21     SYNTAX      INTEGER (0..255)
22     MAX-ACCESS   read-only
23     STATUS       current
24     DESCRIPTION
25         "The identifier of the downlink channel to which this
26         message refers."
27     REFERENCE
28         "Subclause 6.3.2.3.1, Table 15, in IEEE Std 802.16-2004"
29     ::= { wman2IfSsOfdmDownlinkChannelEntry 10 }
30
31  wman2IfSsOfdmUcdBurstProfileTable OBJECT-TYPE
32
33     SYNTAX      SEQUENCE OF Wman2IfSsOfdmUcdBurstProfileEntry
34     MAX-ACCESS   not-accessible
35     STATUS       current
36     DESCRIPTION
37         "This table contains UCD burst profiles for each uplink
38         channel"
39     REFERENCE
40         "Table 356, in IEEE Std 802.16-2004"
41     ::= { wman2IfSsOfdmPhy 3 }
42
43  wman2IfSsOfdmUcdBurstProfileEntry OBJECT-TYPE
44
45     SYNTAX      Wman2IfSsOfdmUcdBurstProfileEntry
46     MAX-ACCESS   not-accessible
47     STATUS       current
48     DESCRIPTION
49         "This table provides one row for each UCD burst profile.
50         This table is double indexed. The primary index is an
51         ifIndex with an ifType of propBWAp2Mp. The secondary index
52         is wman2IfSsOfdmOfdmUcdBurstProfIndex."
53     INDEX { ifIndex, wman2IfSsOfdmUiucIndex }
54     ::= { wman2IfSsOfdmUcdBurstProfileTable 1 }
55
56  Wman2IfSsOfdmUcdBurstProfileEntry ::= SEQUENCE {
57      wman2IfSsOfdmUiucIndex          INTEGER,
58      wman2IfSsOfdmUcdFecCodeType     Wman2IfOfdmFecCodeType,
59

```

```

1          wman2IfSsOfdmFocusCtPowerBoost          INTEGER,
2          wman2IfSsOfdmUcdTcsEnable                INTEGER}
3
4
5  wman2IfSsOfdmUiucIndex OBJECT-TYPE
6      SYNTAX          INTEGER (5 .. 12)
7      MAX-ACCESS      not-accessible
8      STATUS          current
9      DESCRIPTION
10         "The Uplink Interval Usage Code indicates the uplink burst
11         profile in the UCD message, and is used along with ifIndex
12         to identify an entry in the
13         wman2IfSsOfdmUcdBurstProfileTable."
14     REFERENCE
15         "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
16     ::= { wman2IfSsOfdmUcdBurstProfileEntry 1 }
17
18
19
20  wman2IfSsOfdmUcdFecCodeType OBJECT-TYPE
21      SYNTAX          Wman2IfOfdmFecCodeType
22      MAX-ACCESS      read-only
23      STATUS          current
24      DESCRIPTION
25         "Uplink FEC code type and modulation type"
26     REFERENCE
27         "Table 356, in IEEE Std 802.16-2004"
28     ::= { wman2IfSsOfdmUcdBurstProfileEntry 2 }
29
30
31
32  wman2IfSsOfdmFocusCtPowerBoost OBJECT-TYPE
33      SYNTAX          INTEGER (0 .. 255)
34      MAX-ACCESS      read-only
35      STATUS          current
36      DESCRIPTION
37         "The power boost in dB of focused contention carriers, as
38         described in 8.3.6.3.3."
39     REFERENCE
40         "Table 356, in IEEE Std 802.16-2004"
41     ::= { wman2IfSsOfdmUcdBurstProfileEntry 3 }
42
43
44
45  wman2IfSsOfdmUcdTcsEnable OBJECT-TYPE
46      SYNTAX          INTEGER {tcsDisabled(0),
47                          tcsEnabled(1)}
48      MAX-ACCESS      read-only
49      STATUS          current
50      DESCRIPTION
51         "This parameter determines the transmission convergence
52         sublayer, as described in 8.1.4.3, can be enabled on a
53         per-burst basis for both uplink and downlink. through
54         DIUC/UIUC messages."
55     REFERENCE
56         "Table 356, in IEEE Std 802.16-2004"
57     ::= { wman2IfSsOfdmUcdBurstProfileEntry 4 }
58
59
60
61
62  wman2IfSsOfdmDcdBurstProfileTable OBJECT-TYPE
63      SYNTAX          SEQUENCE OF Wman2IfSsOfdmDcdBurstProfileEntry
64      MAX-ACCESS      not-accessible
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table provides one row for each DCD burst profile.
4          This table is double indexed. The primary index is an
5          ifIndex with an ifType of propBWA2Mp. The secondary
6          index is wman2IfSsOfdmDiucIndex."
7      REFERENCE
8          "Table 362, in IEEE Std 802.16-2004"
9      ::= { wman2IfSsOfdmPhy 4 }
10
11
12
13  wman2IfSsOfdmDcdBurstProfileEntry OBJECT-TYPE
14      SYNTAX      Wman2IfSsOfdmDcdBurstProfileEntry
15      MAX-ACCESS  not-accessible
16      STATUS      current
17      DESCRIPTION
18          "This table provides one row for each DCD burst profile.
19          This table is double indexed. The primary index is an
20          ifIndex with an ifType of propBWA2Mp. The secondary index
21          is wman2IfSsOfdmDcdBurstProfIndex."
22      INDEX { ifIndex, wman2IfSsOfdmDiucIndex }
23      ::= { wman2IfSsOfdmDcdBurstProfileTable 1 }
24
25
26
27  Wman2IfSsOfdmDcdBurstProfileEntry ::= SEQUENCE {
28      wman2IfSsOfdmDiucIndex          INTEGER,
29      wman2IfSsOfdmDownlinkFrequency  Unsigned32,
30      wman2IfSsOfdmDcdFecCodeType     Wman2IfOfdmFecCodeType,
31      wman2IfSsOfdmDiucMandatoryExitThresh  INTEGER,
32      wman2IfSsOfdmDiucMinEntryThresh  INTEGER,
33      wman2IfSsOfdmTcsEnable          INTEGER}
34
35
36
37  wman2IfSsOfdmDiucIndex OBJECT-TYPE
38      SYNTAX      INTEGER (1..11)
39      MAX-ACCESS  not-accessible
40      STATUS      current
41      DESCRIPTION
42          "The Downlink Interval Usage Code indicates the downlink
43          burst profile in the DCD message, and is used along with
44          ifIndex to identify an entry in the
45          wman2IfSsOfdmDcdBurstProfileTable."
46      REFERENCE
47          "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
48      ::= { wman2IfSsOfdmDcdBurstProfileEntry 1 }
49
50
51
52
53  wman2IfSsOfdmDownlinkFrequency OBJECT-TYPE
54      SYNTAX      Unsigned32
55      UNITS       "kHz"
56      MAX-ACCESS  read-only
57      STATUS      current
58      DESCRIPTION
59          "Downlink Frequency (kHz)."
60      REFERENCE
61          "Table 362, in IEEE Std 802.16-2004"
62      ::= { wman2IfSsOfdmDcdBurstProfileEntry 2 }
63
64
65

```

```

1  wman2IfSsOfdmDcdFecCodeType OBJECT-TYPE
2      SYNTAX      Wman2IfOfdmFecCodeType
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "Downlink FEC code type and modulation type"
7      REFERENCE
8          "Table 362, in IEEE Std 802.16-2004"
9      ::= { wman2IfSsOfdmDcdBurstProfileEntry 3 }
10
11
12
13  wman2IfSsOfdmDiucMandatoryExitThresh OBJECT-TYPE
14      SYNTAX      INTEGER (0..255)
15      MAX-ACCESS  read-only
16      STATUS      current
17      DESCRIPTION
18          "DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or
19          below where this DIUC can no longer be used and where this
20          change to a more robust DIUC is required in 0.25 dB units."
21      REFERENCE
22          "Table 362, in IEEE Std 802.16-2004"
23      ::= { wman2IfSsOfdmDcdBurstProfileEntry 4 }
24
25
26
27  wman2IfSsOfdmDiucMinEntryThresh OBJECT-TYPE
28      SYNTAX      INTEGER (0..255)
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32          "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
33          required to start using this DIUC when changing from a more
34          robust DIUC is required, in 0.25 dB units."
35      REFERENCE
36          "Table 362, in IEEE Std 802.16-2004"
37      ::= { wman2IfSsOfdmDcdBurstProfileEntry 5 }
38
39
40
41  wman2IfSsOfdmTcsEnable OBJECT-TYPE
42      SYNTAX      INTEGER {tcsDisabled (0),
43                          tcsEnabled (1)}
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47          "Indicates whether Transmission CONvergence Sublayer
48          is enabled or disabled."
49      REFERENCE
50          "Table 362, in IEEE Std 802.16-2004"
51      ::= { wman2IfSsOfdmDcdBurstProfileEntry 6 }
52
53
54
55
56  --
57  -- SS OFDMA PHY objects
58  --
59
60  wman2IfSsOfdmaPhy OBJECT IDENTIFIER ::= { wman2IfSsPhy 2 }
61
62  wman2IfSsOfdmaUplinkChannelTable OBJECT-TYPE
63      SYNTAX      SEQUENCE OF Wman2IfSsOfdmaUplinkChannelEntry
64      MAX-ACCESS  not-accessible
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table contains UCD channel attributes, defining the
4          transmission characteristics of uplink channels"
5      REFERENCE
6          "Subclause 11.3.1, Table 349 and Table 353, in IEEE Std
7          802.16-2004"
8      ::= { wman2IfSsOfdmaPhy 1 }
9
10
11
12 wman2IfSsOfdmaUplinkChannelEntry OBJECT-TYPE
13     SYNTAX      Wman2IfSsOfdmaUplinkChannelEntry
14     MAX-ACCESS  not-accessible
15     STATUS      current
16     DESCRIPTION
17         "This table provides one row for each uplink channel of
18         multi-sector BS, and is indexed by BS ifIndex. An entry
19         in this table exists for each ifEntry of BS with an
20         ifType of propBWAp2Mp."
21     INDEX       { ifIndex }
22     ::= { wman2IfSsOfdmaUplinkChannelTable 1 }
23
24
25
26 Wman2IfSsOfdmaUplinkChannelEntry ::= SEQUENCE {
27     wman2IfSsOfdmaCtBasedResvTimeout      INTEGER,
28     wman2IfSsOfdmaBwReqOppSize            INTEGER,
29     wman2IfSsOfdmaRangReqOppSize          INTEGER,
30     wman2IfSsOfdmaUplinkCenterFreq        Unsigned32,
31     wman2IfSsOfdmaInitRngCodes            INTEGER,
32     wman2IfSsOfdmaPeriodicRngCodes        INTEGER,
33     wman2IfSsOfdmaBWRngReqCodes           INTEGER,
34     wman2IfSsOfdmaPerRngBackoffStart       INTEGER,
35     wman2IfSsOfdmaPerRngBackoffEnd        INTEGER,
36     wman2IfSsOfdmaStartOfRngCodes         INTEGER,
37     wman2IfSsOfdmaPermutationBase         INTEGER,
38     wman2IfSsOfdmaULAllocSubchBitmap      OCTET STRING,
39     wman2IfSsOfdmaOptPermULAllocSubchBitmap OCTET STRING,
40     wman2IfSsOfdmaBandAMCAllocThreshold   INTEGER,
41     wman2IfSsOfdmaBandAMCReleaseThreshold INTEGER,
42     wman2IfSsOfdmaBandAMCAllocTimer       INTEGER,
43     wman2IfSsOfdmaBandAMCReleaseTimer     INTEGER,
44     wman2IfSsOfdmaBandStatRepMAXPeriod    INTEGER,
45     wman2IfSsOfdmaBandAMCRetryTimer       INTEGER,
46     wman2IfSsOfdmaSafetyChAllocThreshold  INTEGER,
47     wman2IfSsOfdmaSafetyChReleaseThreshold INTEGER,
48     wman2IfSsOfdmaSafetyChAllocTimer      INTEGER,
49     wman2IfSsOfdmaSafetyChReleaseTimer    INTEGER,
50     wman2IfSsOfdmaBinStatRepMAXPeriod     INTEGER,
51     wman2IfSsOfdmaSafetyChARetryTimer     INTEGER,
52     wman2IfSsOfdmaHARQAackDelayULBurst    INTEGER,
53     wman2IfSsOfdmaCQICHBandAMCTranaDelay  INTEGER}
54
55
56
57 wman2IfSsOfdmaCtBasedResvTimeout OBJECT-TYPE
58     SYNTAX      INTEGER (1..255)
59     MAX-ACCESS  read-only
60     STATUS      current
61
62
63
64
65

```

```

1      DESCRIPTION
2          "The number of UL-MAPs to receive before contention-based
3          reservation is attempted again for the same connection."
4      REFERENCE
5          "Table 349, in IEEE Std 802.16-2004"
6          ::= { wman2IfSsOfdmaUplinkChannelEntry 1 }
7
8
9
10     wman2IfSsOfdmaBwReqOppSize OBJECT-TYPE
11         SYNTAX      INTEGER (1..65535)
12         UNITS        "PS"
13         MAX-ACCESS   read-only
14         STATUS       current
15         DESCRIPTION
16             "Size (in units of PS) of PHY payload that SS may use to
17             format and transmit a bandwidth request message in a
18             contention request opportunity. The value includes all
19             PHY overhead as well as allowance for the MAC data the
20             message may hold."
21         REFERENCE
22             "Table 349, in IEEE Std 802.16-2004"
23             ::= { wman2IfSsOfdmaUplinkChannelEntry 2 }
24
25
26
27     wman2IfSsOfdmaRangReqOppSize OBJECT-TYPE
28         SYNTAX      INTEGER (1..65535)
29         UNITS        "PS"
30         MAX-ACCESS   read-only
31         STATUS       current
32         DESCRIPTION
33             "Size (in units of PS) of PHY payload that SS may use to
34             format and transmit a RNG-REQ message in a contention
35             request opportunity. The value includes all PHY overhead
36             as well as allowance for the MAC data the message may
37             hold and the maximum SS/BS roundtrip propagation delay."
38         REFERENCE
39             "Table 349, in IEEE Std 802.16-2004"
40             ::= { wman2IfSsOfdmaUplinkChannelEntry 3 }
41
42
43
44     wman2IfSsOfdmaUplinkCenterFreq OBJECT-TYPE
45         SYNTAX      Unsigned32
46         UNITS        "kHz"
47         MAX-ACCESS   read-only
48         STATUS       current
49         DESCRIPTION
50             " Uplink center frequency (kHz)"
51         REFERENCE
52             "Table 349, in IEEE Std 802.16-2004"
53             ::= { wman2IfSsOfdmaUplinkChannelEntry 4 }
54
55
56
57     wman2IfSsOfdmaInitRngCodes OBJECT-TYPE
58         SYNTAX      INTEGER (0..255)
59         MAX-ACCESS   read-only
60         STATUS       current
61         DESCRIPTION
62             "Number of initial ranging CDMA codes. Possible values are
63
64
65

```



```

1           0..255. The total number of wman2IfSsOfdmaInitRngCodes,
2           wman2IfSsOfdmaPeriodicRngCodes and wman2IfSsOfdmaBWReqCodes
3           shall be equal or less than 256."
4
5       REFERENCE
6           "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
7       DEFVAL      { 30 }
8       ::= { wman2IfSsOfdmaUplinkChannelEntry 5 }
9
10
11 wman2IfSsOfdmaPeriodicRngCodes OBJECT-TYPE
12     SYNTAX      INTEGER (0..255)
13     MAX-ACCESS  read-only
14     STATUS      current
15     DESCRIPTION
16         "Number of periodic ranging CDMA codes. Possible values are
17         0..255. The total number of wman2IfSsOfdmaInitRngCodes,
18         wman2IfSsOfdmaPeriodicRngCodes and wman2IfSsOfdmaBWReqCodes
19         shall be equal or less than 256."
20     REFERENCE
21         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
22     DEFVAL      { 30 }
23     ::= { wman2IfSsOfdmaUplinkChannelEntry 6 }
24
25
26
27 wman2IfSsOfdmaBWReqCodes OBJECT-TYPE
28     SYNTAX      INTEGER (0..255)
29     MAX-ACCESS  read-only
30     STATUS      current
31     DESCRIPTION
32         "Number of bandwidth request codes. Possible values are
33         0..255. The total number of wman2IfSsOfdmaInitRngCodes,
34         wman2IfSsOfdmaPeriodicRngCodes and wman2IfSsOfdmaBWReqCodes
35         shall be equal or less than 256."
36     REFERENCE
37         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
38     DEFVAL      { 30 }
39     ::= { wman2IfSsOfdmaUplinkChannelEntry 7 }
40
41
42
43
44 wman2IfSsOfdmaPerRngBackoffStart OBJECT-TYPE
45     SYNTAX      INTEGER (0..15)
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49         "Initial backoff window size for periodic ranging
50         contention, expressed as a power of 2."
51     REFERENCE
52         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
53     DEFVAL      { 0 }
54     ::= { wman2IfSsOfdmaUplinkChannelEntry 8 }
55
56
57
58
59 wman2IfSsOfdmaPerRngBackoffEnd OBJECT-TYPE
60     SYNTAX      INTEGER (0 .. 15)
61     MAX-ACCESS  read-only
62     STATUS      current
63     DESCRIPTION
64         "Final backoff window size for periodic ranging contention,
65

```

```

1         expressed as a power of 2."
2     REFERENCE
3         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
4     DEFVAL      { 15 }
5     ::= { wman2IfSsOfdmaUplinkChannelEntry 9 }
6
7
8     wman2IfSsOfdmaStartOfRngCodes OBJECT-TYPE
9         SYNTAX      INTEGER (0..255)
10        MAX-ACCESS   read-only
11        STATUS       current
12        DESCRIPTION
13            "Indicates the starting number, S, of the group of codes
14             used for this uplink. All the ranging codes used on this
15             uplink will be between S and ((S+N+M+L) mod 256). Where,
16             N is the number of initial-ranging codes M is the number
17             of periodic-ranging codes L is the number of
18             bandwidth-request codes The range of values is 0 S255"
19        REFERENCE
20            "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
21        DEFVAL      { 0 }
22        ::= { wman2IfSsOfdmaUplinkChannelEntry 10 }
23
24
25     wman2IfSsOfdmaPermutationBase OBJECT-TYPE
26         SYNTAX      INTEGER (0..255)
27         MAX-ACCESS   read-only
28         STATUS       current
29         DESCRIPTION
30             "Determines the UL_IDcell parameter for the subcarrier
31             permutation to be used on this uplink channel"
32         REFERENCE
33             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
34         DEFVAL      { 0 }
35         ::= { wman2IfSsOfdmaUplinkChannelEntry 11 }
36
37
38     wman2IfSsOfdmaULAllocSubchBitmap OBJECT-TYPE
39         SYNTAX      OCTET STRING (SIZE (9))
40         MAX-ACCESS   read-only
41         STATUS       current
42         DESCRIPTION
43             "This is a bitmap describing the sub-channels allocated
44             to the segment in the UL, when using the uplink PUSC
45             permutation. The LSB of the first byte shall correspond to
46             subchannel 0. For any bit that is not set,
47             the corresponding subchannel shall not be used by the SS
48             on that segment"
49         REFERENCE
50             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
51         DEFVAL      { 0 }
52         ::= { wman2IfSsOfdmaUplinkChannelEntry 12 }
53
54
55     wman2IfSsOfdmaOptPermULAllocSubchBitmap OBJECT-TYPE
56         SYNTAX      OCTET STRING (SIZE (13))
57         MAX-ACCESS   read-only
58         STATUS       current
59         DESCRIPTION
60
61
62
63
64
65

```

1 "This is a bitmap describing the sub-channels allocated to
2 the segment in the UL, when using the uplink optional PUSC
3 permutation (see 8.4.6.2.5 in IEEE Std 802.16-2004). The
4 LSB of the first byte shall correspond to subchannel 0.
5 For any bit that is not set, the corresponding subchannel
6 shall not be used by the SS on that segment"
7
8 REFERENCE
9 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
10 ::= { wman2IfSsOfdmaUplinkChannelEntry 13 }
11
12
13 wman2IfSsOfdmaBandAMCAAllocThreshold OBJECT-TYPE
14 SYNTAX INTEGER (0 .. 255)
15 UNITS "dB"
16 MAX-ACCESS read-only
17 STATUS current
18 DESCRIPTION
19 "This object defines the OFDMA band AMC allocation
20 threshold."
21
22 REFERENCE
23 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
24 ::= { wman2IfSsOfdmaUplinkChannelEntry 14 }
25
26
27 wman2IfSsOfdmaBandAMCReleaseThreshold OBJECT-TYPE
28 SYNTAX INTEGER (0 .. 255)
29 UNITS "dB"
30 MAX-ACCESS read-only
31 STATUS current
32 DESCRIPTION
33 "This object defines the OFDMA band AMC release
34 threshold."
35
36 REFERENCE
37 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
38 ::= { wman2IfSsOfdmaUplinkChannelEntry 15 }
39
40
41
42 wman2IfSsOfdmaBandAMCAAllocTimer OBJECT-TYPE
43 SYNTAX INTEGER (0 .. 255)
44 UNITS "Frame"
45 MAX-ACCESS read-only
46 STATUS current
47 DESCRIPTION
48 "This object defines the OFDMA band AMC allocation
49 timer."
50
51 REFERENCE
52 "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
53 ::= { wman2IfSsOfdmaUplinkChannelEntry 16 }
54
55
56 wman2IfSsOfdmaBandAMCReleaseTimer OBJECT-TYPE
57 SYNTAX INTEGER (0 .. 255)
58 UNITS "Frame"
59 MAX-ACCESS read-only
60 STATUS current
61 DESCRIPTION
62 "This object defines the OFDMA band AMC release
63 timer."
64
65

```

1      REFERENCE
2          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
3      ::= { wman2IfSsOfdmaUplinkChannelEntry 17 }
4
5
6  wman2IfSsOfdmaBandStatRepMAXPeriod OBJECT-TYPE
7      SYNTAX      INTEGER (0 .. 255)
8      UNITS       "Frame"
9      MAX-ACCESS  read-only
10     STATUS      current
11     DESCRIPTION
12         "This object defines the OFDMA band status reporting
13         maximum period."
14     REFERENCE
15         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
16     ::= { wman2IfSsOfdmaUplinkChannelEntry 18 }
17
18
19
20  wman2IfSsOfdmaBandAMCRetryTimer OBJECT-TYPE
21      SYNTAX      INTEGER (0 .. 255)
22      UNITS       "Frame"
23      MAX-ACCESS  read-only
24      STATUS      current
25      DESCRIPTION
26         "This object defines the OFDMA band AMC retry
27         timer."
28     REFERENCE
29         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
30     ::= { wman2IfSsOfdmaUplinkChannelEntry 19 }
31
32
33
34
35  wman2IfSsOfdmaSafetyChAllocThreshold 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 allocation
42         threshold."
43     REFERENCE
44         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
45     ::= { wman2IfSsOfdmaUplinkChannelEntry 20 }
46
47
48
49  wman2IfSsOfdmaSafetyChReleaseThreshold OBJECT-TYPE
50      SYNTAX      INTEGER (0 .. 255)
51      UNITS       "dB"
52      MAX-ACCESS  read-only
53      STATUS      current
54      DESCRIPTION
55         "This object defines the OFDMA safety channel release
56         threshold."
57     REFERENCE
58         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
59     ::= { wman2IfSsOfdmaUplinkChannelEntry 21 }
60
61
62
63  wman2IfSsOfdmaSafetyChAllocTimer OBJECT-TYPE
64      SYNTAX      INTEGER (0 .. 255)
65

```

```

1      UNITS          "Frame"
2      MAX-ACCESS read-only
3      STATUS         current
4      DESCRIPTION
5          "This object defines the OFDMA safety channel allocation
6          timer."
7      REFERENCE
8          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
9      ::= { wman2IfSsOfdmaUplinkChannelEntry 22 }
10
11
12
13 wman2IfSsOfdmaSafetyChReleaseTimer OBJECT-TYPE
14     SYNTAX      INTEGER (0 .. 255)
15     UNITS       "Frame"
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "This object defines the OFDMA safety channel release
20         timer."
21     REFERENCE
22         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
23     ::= { wman2IfSsOfdmaUplinkChannelEntry 23 }
24
25
26
27 wman2IfSsOfdmaBinStatRepMAXPeriod OBJECT-TYPE
28     SYNTAX      INTEGER (0 .. 255)
29     UNITS       "Frame"
30     MAX-ACCESS  read-only
31     STATUS      current
32     DESCRIPTION
33         "This object defines the OFDMA bin status reporting
34         maximum period."
35     REFERENCE
36         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
37     ::= { wman2IfSsOfdmaUplinkChannelEntry 24 }
38
39
40
41
42 wman2IfSsOfdmaSafetyChaRetryTimer OBJECT-TYPE
43     SYNTAX      INTEGER (0 .. 255)
44     UNITS       "Frame"
45     MAX-ACCESS  read-only
46     STATUS      current
47     DESCRIPTION
48         "This object defines the OFDMA safety channel retry
49         timer."
50     REFERENCE
51         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
52     ::= { wman2IfSsOfdmaUplinkChannelEntry 25 }
53
54
55
56 wman2IfSsOfdmaHARQAackDelayULBurst OBJECT-TYPE
57     SYNTAX      INTEGER {oneframeoffset(1),
58                          twoframesoffset(2),
59                          threeframesoffset(3)}
60     MAX-ACCESS  read-only
61     STATUS      current
62     DESCRIPTION
63         "This object defines the OFDMA H-ARQ ACK delay for UL burst."
64
65

```

```

1          1 = one frame offset
2          2 = two frames offset
3          3 = three frames offset"
4
5      REFERENCE
6          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
7          ::= { wman2IfSsOfdmaUplinkChannelEntry 26 }
8
9
10     wman2IfSsOfdmaCQICHBandAMCTranaDelay OBJECT-TYPE
11         SYNTAX      INTEGER (0 .. 255)
12         UNITS        "Frame"
13         MAX-ACCESS   read-only
14         STATUS        current
15         DESCRIPTION
16             "This object defines the OFDMA CQICH band AMC transition
17             delay."
18         REFERENCE
19             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
20             ::= { wman2IfSsOfdmaUplinkChannelEntry 27 }
21
22
23
24     wman2IfSsOfdmaDownlinkChannelTable OBJECT-TYPE
25         SYNTAX      SEQUENCE OF Wman2IfSsOfdmaDownlinkChannelEntry
26         MAX-ACCESS   not-accessible
27         STATUS        current
28         DESCRIPTION
29             "This table contains DCD channel attributes, defining the
30             transmission characteristics of downlink channels"
31         REFERENCE
32             "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
33             ::= { wman2IfSsOfdmaPhy 2 }
34
35
36
37     wman2IfSsOfdmaDownlinkChannelEntry OBJECT-TYPE
38         SYNTAX      Wman2IfSsOfdmaDownlinkChannelEntry
39         MAX-ACCESS   not-accessible
40         STATUS        current
41         DESCRIPTION
42             "This table provides one row for each downlink channel of
43             multi-sector BS, and is indexed by BS ifIndex. An entry in
44             this table exists for each ifEntry of BS with an ifType of
45             propBWA2Mp."
46         INDEX        { ifIndex }
47         ::= { wman2IfSsOfdmaDownlinkChannelTable 1 }
48
49
50
51     Wman2IfSsOfdmaDownlinkChannelEntry ::= SEQUENCE {
52         wman2IfSsOfdmaBsEIRP                INTEGER,
53         wman2IfSsOfdmaChannelNumber          Wman2IfChannelNumber,
54         wman2IfSsOfdmaATTG                   INTEGER,
55         wman2IfSsOfdmaARTG                   INTEGER,
56         wman2IfSsOfdmaInitRngMaxRSS           INTEGER,
57         wman2IfSsOfdmaDownlinkCenterFreq     Unsigned32,
58         wman2IfSsOfdmaBsId                   Wman2IfBsIdType,
59         wman2IfSsOfdmaMacVersion              Wman2IfMacVersion,
60         wman2IfSsOfdmaFrameDurationCode      INTEGER,
61         wman2IfSsOfdmaSizeCqichIdField       INTEGER,
62         wman2IfSsOfdmaHARQAackDelayBurst     INTEGER}
63
64
65

```

```

1
2 wman2IfSsOfdmaBsEIRP OBJECT-TYPE
3     SYNTAX      INTEGER (-32768..32767)
4     UNITS       "dBm"
5     MAX-ACCESS  read-only
6     STATUS      current
7     DESCRIPTION
8         "The EIRP is the equivalent isotropic radiated power of
9         the base station, which is computed for a simple
10        single-antenna transmitter."
11    REFERENCE
12        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
13    ::= { wman2IfSsOfdmaDownlinkChannelEntry 1 }
14
15 wman2IfSsOfdmaChannelNumber OBJECT-TYPE
16     SYNTAX      Wman2IfChannelNumber
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20         "Downlink channel number as defined in 8.5. Used for
21         license-exempt operation only."
22    REFERENCE
23        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
24    ::= { wman2IfSsOfdmaDownlinkChannelEntry 2 }
25
26 wman2IfSsOfdmaTTG OBJECT-TYPE
27     SYNTAX      INTEGER (0..255)
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31         "Transmit / Receive Transition Gap."
32    REFERENCE
33        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
34    ::= { wman2IfSsOfdmaDownlinkChannelEntry 3 }
35
36 wman2IfSsOfdmaRTG OBJECT-TYPE
37     SYNTAX      INTEGER (0..255)
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41         "Receive / Transmit Transition Gap."
42    REFERENCE
43        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
44    ::= { wman2IfSsOfdmaDownlinkChannelEntry 4 }
45
46 wman2IfSsOfdmaInitRngMaxRSS OBJECT-TYPE
47     SYNTAX      INTEGER (-32768..32767)
48     UNITS       "dBm"
49     MAX-ACCESS  read-only
50     STATUS      current
51     DESCRIPTION
52         "Initial Ranging Max. equivalent isotropic received power
53         at BS Signed in units of 1 dBm."
54    REFERENCE
55

```

```

1         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
2         ::= { wman2IfSsOfdmaDownlinkChannelEntry 5 }
3
4
5 wman2IfSsOfdmaDownlinkCenterFreq OBJECT-TYPE
6     SYNTAX      Unsigned32
7     UNITS       "kHz"
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "Downlink center frequency (kHz)."

```



```

1  wman2IfSsOfdmaSizeCqichIdField OBJECT-TYPE
2      SYNTAX          INTEGER {threebits(1),
3                          fourbits(2),
4                          fivebits(3),
5                          sixbits(4),
6                          sevenbits(5),
7                          eightbits(6),
8                          ninebits(7)}
9
10
11  MAX-ACCESS    read-only
12  STATUS        current
13  DESCRIPTION
14      "This object defines the size of CQICH ID field.
15      0 = Reserved
16      1 = 3 bits
17      2 = 4 bits
18      3 = 5 bits
19      4 = 6 bits
20      5 = 7 bits
21      6 = 8 bits
22      7 = 9 bits
23      8...255 = Reserved"
24
25  REFERENCE
26      "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
27      ::= { wman2IfSsOfdmaDownlinkChannelEntry 10 }
28
29
30
31  wman2IfSsOfdmaHARQAackDelayBurst OBJECT-TYPE
32      SYNTAX          INTEGER {oneframeoffset(1),
33                          twoframesoffset(2),
34                          threeframesoffset(3)}
35
36  MAX-ACCESS    read-only
37  STATUS        current
38  DESCRIPTION
39      "This object defines the OFDMA H-ARQ ACK delay for DL burst.
40      1 = one frame offset
41      2 = two frames offset
42      3 = three frames offset"
43
44  REFERENCE
45      "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
46      ::= { wman2IfSsOfdmaDownlinkChannelEntry 11 }
47
48
49  wman2IfSsOfdmaUcdBurstProfileTable OBJECT-TYPE
50      SYNTAX          SEQUENCE OF Wman2IfSsOfdmaUcdBurstProfileEntry
51  MAX-ACCESS    not-accessible
52  STATUS        current
53  DESCRIPTION
54      "This table contains UCD burst profiles for each uplink
55      channel"
56
57  REFERENCE
58      "Subclause 11.3.1.1, Table 288 and Table 357, in IEEE
59      Std 802.16-2004"
60      ::= { wman2IfSsOfdmaPhy 3 }
61
62
63  wman2IfSsOfdmaUcdBurstProfileEntry OBJECT-TYPE
64      SYNTAX          Wman2IfSsOfdmaUcdBurstProfileEntry
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table provides one row for each UCD burst profile.
5          This table is double indexed. The primary index is an
6          ifIndex with an ifType of propBWA2Mp. The secondary index
7          is wman2IfSsOfdmaUiucIndex."
8      INDEX          { ifIndex, wman2IfSsOfdmaUiucIndex }
9      ::= { wman2IfSsOfdmaUcdBurstProfileTable 1 }
10
11
12
13      Wman2IfSsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
14          wman2IfSsOfdmaUiucIndex          INTEGER,
15          wman2IfSsOfdmaUcdFecCodeType     Wman2IfOfdmaFecCodeType,
16          wman2IfSsOfdmaRangingDataRatio   INTEGER,
17          wman2IfSsOfdmaNorCOverNOverride  OCTET STRING}
18
19
20      wman2IfSsOfdmaUiucIndex OBJECT-TYPE
21          SYNTAX      INTEGER (1 .. 10)
22          MAX-ACCESS  read-only
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              wman2IfSsOfdmaUcdBurstProfileTable."
29          REFERENCE
30              "Subclause 8.4.5.4.1, in IEEE Std 802.16-2004"
31          ::= { wman2IfSsOfdmaUcdBurstProfileEntry 1 }
32
33
34
35      wman2IfSsOfdmaUcdFecCodeType OBJECT-TYPE
36          SYNTAX      Wman2IfOfdmaFecCodeType
37          MAX-ACCESS  read-only
38          STATUS      current
39          DESCRIPTION
40              "Uplink FEC code type and modulation type"
41          REFERENCE
42              "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
43          ::= { wman2IfSsOfdmaUcdBurstProfileEntry 2 }
44
45
46
47      wman2IfSsOfdmaRangingDataRatio OBJECT-TYPE
48          SYNTAX      INTEGER (0 .. 255)
49          MAX-ACCESS  read-only
50          STATUS      current
51          DESCRIPTION
52              "Reducing factor in units of 1 dB, between the power used
53              for this burst and power should be used for CDMA Ranging."
54          REFERENCE
55              "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
56          ::= { wman2IfSsOfdmaUcdBurstProfileEntry 3 }
57
58
59
60      wman2IfSsOfdmaNorCOverNOverride OBJECT-TYPE
61          SYNTAX      OCTET STRING (SIZE (5))
62          MAX-ACCESS  read-only
63          STATUS      current
64
65

```

```

1      DESCRIPTION
2          "This is a list of numbers, where each number is encoded by
3          one nibble, and interpreted as a signed integer. The nibbles
4          correspond in order to the list define by Table 334 in IEEE
5          Std 802.16-2004 starting from the second line, such that
6          the LS nibble of the first byte corresponds to the second
7          line in the table. The number encoded by each nibble
8          represents the difference in normalized C/N relative to the
9          previous line in the table"
10
11      REFERENCE
12          "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
13      ::= { wman2IfSsOfdmaUcdBurstProfileEntry 4 }
14
15
16  wman2IfSsOfdmaDcdBurstProfileTable OBJECT-TYPE
17      SYNTAX      SEQUENCE OF Wman2IfSsOfdmaDcdBurstProfileEntry
18      MAX-ACCESS  not-accessible
19      STATUS      current
20      DESCRIPTION
21          "This table provides one row for each DCD burst profile.
22          This table is double indexed. The primary index is an
23          ifIndex with an ifType of propBWA2Mp. The secondary index
24          is wman2IfSsOfdmaDiucIndex."
25      ::= { wman2IfSsOfdmaPhy 4 }
26
27
28  wman2IfSsOfdmaDcdBurstProfileEntry OBJECT-TYPE
29      SYNTAX      Wman2IfSsOfdmaDcdBurstProfileEntry
30      MAX-ACCESS  not-accessible
31      STATUS      current
32      DESCRIPTION
33          "This table provides one row for each DCD burst profile,
34          and is double indexed. The primary index is an ifIndex
35          with an ifType of propBWA2Mp. The secondary index is
36          wman2IfSsOfdmaDiucIndex."
37      INDEX      { ifIndex, wman2IfSsOfdmaDiucIndex }
38      ::= { wman2IfSsOfdmaDcdBurstProfileTable 1 }
39
40
41  Wman2IfSsOfdmaDcdBurstProfileEntry ::= SEQUENCE {
42      wman2IfSsOfdmaDiucIndex          INTEGER,
43      wman2IfSsOfdmaDownlinkFrequency Unsigned32,
44      wman2IfSsOfdmaDcdFecCodeType     Wman2IfOfdmaFecCodeType,
45      wman2IfSsOfdmaDiucMandatoryExitThresh INTEGER,
46      wman2IfSsOfdmaDiucMinEntryThresh INTEGER}
47
48
49  wman2IfSsOfdmaDiucIndex OBJECT-TYPE
50      SYNTAX      INTEGER (0 .. 12)
51      MAX-ACCESS  read-only
52      STATUS      current
53      DESCRIPTION
54          "The Downlink Interval Usage Code indicates the downlink
55          burst profile in the DCD message, and is used
56          along with ifIndex to identify an entry in the
57          wman2IfSsOfdmaDcdBurstProfileTable."
58      REFERENCE
59          "Subclause 8.4.5.3.1, in IEEE Std 802.16-2004"
60
61
62
63
64
65

```

```

1      ::= { wman2IfSsOfdmaDcdBurstProfileEntry 1 }
2
3
4  wman2IfSsOfdmaDownlinkFrequency OBJECT-TYPE
5      SYNTAX      Unsigned32
6      UNITS       "kHz"
7      MAX-ACCESS  read-only
8      STATUS      current
9      DESCRIPTION
10         "Downlink Frequency (kHz)."
```

REFERENCE

```

11         "Subclause 11.4.2, Table 359, in IEEE Std 802.16-2004"
12
13     ::= { wman2IfSsOfdmaDcdBurstProfileEntry 2 }
14
15
16  wman2IfSsOfdmaDcdFecCodeType OBJECT-TYPE
17      SYNTAX      Wman2IfOfdmaFecCodeType
18      MAX-ACCESS  read-only
19      STATUS      current
20      DESCRIPTION
21         "Downlink FEC code type and modulation type"
```

REFERENCE

```

22         "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
23
24     ::= { wman2IfSsOfdmaDcdBurstProfileEntry 3 }
25
26
27  wman2IfSsOfdmaDiucMandatoryExitThresh OBJECT-TYPE
28      SYNTAX      INTEGER (0..255)
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32         "DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or
33         below where this DIUC can no longer be used and where this
34         change to a more robust DIUC is required in 0.25 dB units."
```

REFERENCE

```

35         "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
36
37     ::= { wman2IfSsOfdmaDcdBurstProfileEntry 4 }
38
39
40  wman2IfSsOfdmaDiucMinEntryThresh OBJECT-TYPE
41      SYNTAX      INTEGER (0..255)
42      MAX-ACCESS  read-only
43      STATUS      current
44      DESCRIPTION
45         "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
46         required to start using this DIUC when changing from a more
47         robust DIUC is required, in 0.25 dB units."
```

REFERENCE

```

48         "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
49
50     ::= { wman2IfSsOfdmaDcdBurstProfileEntry 5 }
51
52
53  --
54  -- Common object group - containing common tables and objects to be
55  -- implemented in both Base Station and Subscriber Station
56  --
57  -- wman2IfCmnPacketCs contain the Packet Convergence Sublayer objects
58  -- that are common to both Base Station and Subscriber Station
59  --
60
61
62
63
64
65
```

```

1  wman2IfCmnPacketCs OBJECT IDENTIFIER ::= { wman2IfCommonObjects 1 }
2
3  wman2IfCmnClassifierRuleTable OBJECT-TYPE
4      SYNTAX      SEQUENCE OF Wman2IfCmnClassifierRuleEntry
5      MAX-ACCESS  not-accessible
6      STATUS      current
7      DESCRIPTION
8          "This table contains packet classifier rules associated
9          with service flows."
10     ::= { wman2IfCmnPacketCs 1 }
11
12
13
14  wman2IfCmnClassifierRuleEntry OBJECT-TYPE
15      SYNTAX      Wman2IfCmnClassifierRuleEntry
16      MAX-ACCESS  not-accessible
17      STATUS      current
18      DESCRIPTION
19          "This table provides one row for each packet classifier
20          rule, and is indexed by ifIndex, wman2IfCmnCpsSfId, and
21          wman2IfCmnClassifierRuleIndex. ifIndex is associated with
22          the BS sector. wman2IfCmnCpsSfId identifies the service
23          flow, and wman2IfCmnClassifierRuleIndex identifies the
24          packet classifier rule."
25      INDEX { ifIndex, wman2IfCmnCpsSfId,
26              wman2IfCmnClassifierRuleIndex }
27      ::= { wman2IfCmnClassifierRuleTable 1 }
28
29
30
31
32  Wman2IfCmnClassifierRuleEntry ::= SEQUENCE {
33      wman2IfCmnClassifierRuleIndex      Unsigned32,
34      wman2IfCmnClassifierRulePriority    INTEGER,
35      wman2IfCmnClassifierRuleIpTosLow    INTEGER,
36      wman2IfCmnClassifierRuleIpTosHigh   INTEGER,
37      wman2IfCmnClassifierRuleIpTosMask   INTEGER,
38      wman2IfCmnClassifierRuleIpProtocol  Integer32,
39      wman2IfCmnClassifierRuleIpSourceAddr InetAddress,
40      wman2IfCmnClassifierRuleIpSourceMask InetAddress,
41      wman2IfCmnClassifierRuleIpDestAddr  InetAddress,
42      wman2IfCmnClassifierRuleIpDestMask  InetAddress,
43      wman2IfCmnClassifierRuleSourcePortStart Integer32,
44      wman2IfCmnClassifierRuleSourcePortEnd Integer32,
45      wman2IfCmnClassifierRuleDestPortStart Integer32,
46      wman2IfCmnClassifierRuleDestPortEnd Integer32,
47      wman2IfCmnClassifierRuleDestMacAddr MacAddress,
48      wman2IfCmnClassifierRuleDestMacMask MacAddress,
49      wman2IfCmnClassifierRuleSourceMacAddr MacAddress,
50      wman2IfCmnClassifierRuleSourceMacMask MacAddress,
51      wman2IfCmnClassifierRuleEnetProtocolType INTEGER,
52      wman2IfCmnClassifierRuleEnetProtocol Integer32,
53      wman2IfCmnClassifierRuleUserPriLow  Integer32,
54      wman2IfCmnClassifierRuleUserPriHigh Integer32,
55      wman2IfCmnClassifierRuleVlanId      Integer32,
56      wman2IfCmnClassifierRuleState       INTEGER,
57      wman2IfCmnClassifierRulePkts        Counter64,
58      wman2IfCmnClassifierRuleIpv6FlowLabel Wman2IfIpv6FlowLabel,
59      wman2IfCmnClassifierRuleBitMap      Wman2IfClassifierBitMap
60
61
62
63
64
65

```

```

1                                     }
2
3
4 wman2IfCmnClassifierRuleIndex OBJECT-TYPE
5     SYNTAX      Unsigned32 (1..4294967295)
6     MAX-ACCESS  not-accessible
7     STATUS      current
8     DESCRIPTION
9         "An index is assigned to each classifier in the classifiers
10        table"
11
12     ::= { wman2IfCmnClassifierRuleEntry 1 }
13
14 wman2IfCmnClassifierRulePriority OBJECT-TYPE
15     SYNTAX      INTEGER (0..255)
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "The value specifies the order of evaluation of the
20         classifiers. The higher the value the higher the
21         priority. The value of 0 is used as default in
22         provisioned service flows classifiers. The default
23         value of 64 is used for dynamic service flow classifiers.
24         If the referenced parameter is not present in a classifier
25         , this object reports the default value as defined above"
26
27     REFERENCE
28         "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
29
30     DEFVAL      { 0 }
31
32     ::= { wman2IfCmnClassifierRuleEntry 2 }
33
34
35 wman2IfCmnClassifierRuleIpTosLow OBJECT-TYPE
36     SYNTAX      INTEGER (0 .. 255)
37     MAX-ACCESS  read-only
38     STATUS      current
39     DESCRIPTION
40         "The low value of a range of TOS byte values. If the
41         referenced parameter is not present in a classifier, this
42         object reports the value of 0."
43
44     REFERENCE
45         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
46
47     ::= { wman2IfCmnClassifierRuleEntry 3 }
48
49
50 wman2IfCmnClassifierRuleIpTosHigh OBJECT-TYPE
51     SYNTAX      INTEGER (0 .. 255)
52     MAX-ACCESS  read-only
53     STATUS      current
54     DESCRIPTION
55         "The 8-bit high value of a range of TOS byte values.
56         If the referenced parameter is not present in a classifier
57         , this object reports the value of 0."
58
59     REFERENCE
60         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
61
62     ::= { wman2IfCmnClassifierRuleEntry 4 }
63
64
65 wman2IfCmnClassifierRuleIpTosMask OBJECT-TYPE
66     SYNTAX      INTEGER (0 .. 255)

```

```

1      MAX-ACCESS  read-only
2      STATUS      current
3      DESCRIPTION
4          "The mask value is bitwise ANDed with TOS byte in an IP
5          packet and this value is used for the range checking of
6          TosLow and TosHigh. If the referenced parameter is not
7          present in a classifier, this object reports the value
8          of 0."
9
10     REFERENCE
11         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
12     ::= { wman2IfCmnClassifierRuleEntry 5 }
13
14
15 wman2IfCmnClassifierRuleIpProtocol OBJECT-TYPE
16     SYNTAX      Integer32 (0..255)
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20         "This object indicates the value of the IP Protocol field
21         required for IP packets to match this rule. If the
22         referenced parameter is not present in a classifier, this
23         object reports the value of 0."
24     REFERENCE
25         "Subclause 11.13.19.3.4.3 in IEEE Std 802.16-2004"
26     ::= { wman2IfCmnClassifierRuleEntry 6 }
27
28
29 wman2IfCmnClassifierRuleIpSourceAddr OBJECT-TYPE
30     SYNTAX      InetAddress
31     MAX-ACCESS  read-only
32     STATUS      current
33     DESCRIPTION
34         "This object specifies the value of the IP Source Address
35         required for packets to match this rule. An IP packet
36         matches the rule when the packet ip source address bitwise
37         ANDed with the wman2IfCmnClassifierRuleIpSourceMask value
38         equals the wman2IfCmnClassifierRuleIpSourceAddr value.
39         If the referenced parameter is not present in a classifier
40         , this object reports the value of 0.0.0.0."
41     REFERENCE
42         "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
43     ::= { wman2IfCmnClassifierRuleEntry 7 }
44
45
46 wman2IfCmnClassifierRuleIpSourceMask OBJECT-TYPE
47     SYNTAX      InetAddress
48     MAX-ACCESS  read-only
49     STATUS      current
50     DESCRIPTION
51         "This object specifies which bits of a packet's IP Source
52         Address that are compared to match this rule. An IP packet
53         matches the rule when the packet source address bitwise
54         ANDed with the
55         wman2IfCmnClassifierRuleIpSourceMask value equals the
56         wman2IfCmnClassifierRuleIpSourceAddr value.
57         If the referenced parameter is not present in a classifier
58         , this object reports the value of 0.0.0.0."
59
60
61
62
63
64
65

```

```

1      REFERENCE
2          "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
3      ::= { wman2IfCmnClassifierRuleEntry 8 }
4
5
6  wman2IfCmnClassifierRuleIpDestAddr OBJECT-TYPE
7      SYNTAX      InetAddress
8      MAX-ACCESS  read-only
9      STATUS      current
10     DESCRIPTION
11         "This object specifies the value of the IP Destination
12         Address required for packets to match this rule. An IP
13         packet matches the rule when the packet IP destination
14         address bitwise ANDed with the
15         wman2IfCmnClassifierRuleIpDestMask value equals the
16         wman2IfCmnClassifierRuleIpDestAddr value.
17         If the referenced parameter is not present in a
18         classifier, this object reports the value of 0.0.0.0."
19     REFERENCE
20         "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
21     ::= { wman2IfCmnClassifierRuleEntry 9 }
22
23
24  wman2IfCmnClassifierRuleIpDestMask OBJECT-TYPE
25      SYNTAX      InetAddress
26      MAX-ACCESS  read-only
27      STATUS      current
28      DESCRIPTION
29         "This object specifies which bits of a packet's IP
30         Destination Address that are compared to match this rule.
31         An IP packet matches the rule when the packet destination
32         address bitwise ANDed with the
33         wman2IfCmnClassifierRuleIpDestMask value equals the
34         wman2IfCmnClassifierRuleIpDestAddr value.
35         If the referenced parameter is not present in a classifier
36         , this object reports the value of 0.0.0.0."
37     REFERENCE
38         "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
39     ::= { wman2IfCmnClassifierRuleEntry 10 }
40
41
42  wman2IfCmnClassifierRuleSourcePortStart OBJECT-TYPE
43      SYNTAX      Integer32 (0..65535)
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47         "This object specifies the low end inclusive range of
48         TCP/UDP source port numbers to which a packet is compared
49         . This object is irrelevant for non-TCP/UDP IP packets.
50         If the referenced parameter is not present in a
51         classifier, this object reports the value of 0."
52     REFERENCE
53         "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
54     ::= { wman2IfCmnClassifierRuleEntry 11 }
55
56
57  wman2IfCmnClassifierRuleSourcePortEnd OBJECT-TYPE
58      SYNTAX      Integer32 (0..65535)

```



```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "This object specifies the high end inclusive range of
5          TCP/UDP source port numbers to which a packet is compared.
6          This object is irrelevant for non-TCP/UDP IP packets.
7          If the referenced parameter is not present in a classifier,
8          this object reports the value of 65535."
9
10     REFERENCE
11         "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
12     ::= { wman2IfCmnClassifierRuleEntry 12 }
13
14
15 wman2IfCmnClassifierRuleDestPortStart OBJECT-TYPE
16     SYNTAX      Integer32 (0..65535)
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20         "This object specifies the low end inclusive range of
21         TCP/UDP destination port numbers to which a packet is
22         compared. If the referenced parameter is not present
23         in a classifier, this object reports the value of 0."
24     REFERENCE
25         "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
26     ::= { wman2IfCmnClassifierRuleEntry 13 }
27
28
29 wman2IfCmnClassifierRuleDestPortEnd OBJECT-TYPE
30     SYNTAX      Integer32 (0..65535)
31     MAX-ACCESS  read-only
32     STATUS      current
33     DESCRIPTION
34         "This object specifies the high end inclusive range of
35         TCP/UDP destination port numbers to which a packet is
36         compared. If the referenced parameter is not present
37         in a classifier, this object reports the value of
38         65535."
39     REFERENCE
40         "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
41     ::= { wman2IfCmnClassifierRuleEntry 14 }
42
43
44 wman2IfCmnClassifierRuleDestMacAddr OBJECT-TYPE
45     SYNTAX      MacAddress
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49         "An Ethernet packet matches an entry when its destination
50         MAC address bitwise ANDed with
51         wman2IfCmnClassifierRuleDestMacMask equals the value of
52         wman2IfCmnClassifierRuleDestMacAddr. If the referenced
53         parameter is not present in a classifier, this object
54         reports the value of '000000000000'H."
55     REFERENCE
56         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
57     ::= { wman2IfCmnClassifierRuleEntry 15 }
58
59
60
61
62
63
64
65

```

```

1  wman2IfCmnClassifierRuleDestMacMask OBJECT-TYPE
2      SYNTAX      MacAddress
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "An Ethernet packet matches an entry when its destination
7          MAC address bitwise ANDed with
8          wman2IfCmnClassifierRuleDestMacMask equals the value of
9          wman2IfCmnClassifierRuleDestMacAddr. If the referenced
10         parameter is not present in a classifier, this object
11         reports the value of '000000000000'H."
12     REFERENCE
13         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
14     ::= { wman2IfCmnClassifierRuleEntry 16 }
15
16  wman2IfCmnClassifierRuleSourceMacAddr OBJECT-TYPE
17      SYNTAX      MacAddress
18      MAX-ACCESS  read-only
19      STATUS      current
20      DESCRIPTION
21          "An Ethernet packet matches this entry when its source
22          MAC address bitwise ANDed with
23          wman2IfCmnClassifierRuleSourceMacMask equals the value
24          of wman2IfCmnClassifierRuleSourceMacAddr. If the
25          referenced parameter is not present in a classifier,
26          this object reports the value of '000000000000'H."
27     REFERENCE
28         "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
29     ::= { wman2IfCmnClassifierRuleEntry 17 }
30
31  wman2IfCmnClassifierRuleSourceMacMask 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          wman2IfCmnClassifierRuleSourceMacMask equals the value of
39          wman2IfCmnClassifierRuleSourceMacAddr. If the referenced
40          parameter is not present in a classifier, this object
41          reports the value of '000000000000'H."
42     REFERENCE
43         "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
44     ::= { wman2IfCmnClassifierRuleEntry 18 }
45
46  wman2IfCmnClassifierRuleEnetProtocolType OBJECT-TYPE
47      SYNTAX      INTEGER { none(0),
48                        ethertype(1),
49                        dsap(2) }
50      MAX-ACCESS  read-only
51      STATUS      current
52      DESCRIPTION
53          "This object indicates the format of the layer 3 protocol
54          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"
`::= { wman2IfCmnClassifierRuleEntry 19 }`

wman2IfCmnClassifierRuleEnetProtocol OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If `wman2IfCmnClassifierRuleEnetProtocolType` is `none(0)`, this object is ignored when considering whether a packet matches the current rule.
 If `wman2IfCmnClassifierRuleEnetProtocolType` 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 `wman2IfCmnClassifierRuleEnetProtocolType` 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"
`::= { wman2IfCmnClassifierRuleEntry 20 }`

wman2IfCmnClassifierRuleUserPriLow OBJECT-TYPE

SYNTAX Integer32 (0..7)

MAX-ACCESS read-only

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 `wman2IfCmnClassifierRulePriLow` and `wman2IfCmnClassifierRulePriHigh` in order to match this

```

1         rule.
2         If the referenced parameter is not present in the
3         classifier, the value of this object is reported as 0."
4     REFERENCE
5         "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
6     ::= { wman2IfCmnClassifierRuleEntry 21 }
7
8
9
10    wman2IfCmnClassifierRuleUserPriHigh OBJECT-TYPE
11        SYNTAX      Integer32 (0..7)
12        MAX-ACCESS   read-only
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 wman2IfCmnClassifierRulePriLow
21            and wman2IfCmnClassifierRulePriHigh 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        ::= { wman2IfCmnClassifierRuleEntry 22 }
28
29
30
31
32    wman2IfCmnClassifierRuleVlanId OBJECT-TYPE
33        SYNTAX      Integer32 (0..4095)
34        MAX-ACCESS   read-only
35        STATUS       current
36        DESCRIPTION
37            "This object applies only to Ethernet frames using the
38            802.1P/Q tag header.
39            If this object's value is nonzero, tagged packets must
40            have a VLAN Identifier that matches the value in order
41            to match the rule.
42            Only the least significant 12 bits of this object's
43            value are valid.
44            If the referenced parameter is not present in the
45            classifier, the value of this object is reported as 0."
46        REFERENCE
47            "Subclause 11.13.19.3.4.12 in IEEE Std 802.16-2004"
48        ::= { wman2IfCmnClassifierRuleEntry 23 }
49
50
51
52
53
54    wman2IfCmnClassifierRuleState OBJECT-TYPE
55        SYNTAX      INTEGER {active(1),
56                           inactive(2)}
57        MAX-ACCESS   read-only
58        STATUS       deprecated
59        DESCRIPTION
60            "This object indicates whether or not the classifier is
61            enabled to classify packets to a Service Flow.
62            If the referenced parameter is not present in the
63            classifier, the value of this object is reported
64
65

```

```

1         as active(1).
2         ::= { wman2IfCmnClassifierRuleEntry 24 }
3
4
5 wman2IfCmnClassifierRulePkts OBJECT-TYPE
6     SYNTAX      Counter64
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "This object counts the number of packets that have
11        been classified using this entry."
12        ::= { wman2IfCmnClassifierRuleEntry 25 }
13
14 wman2IfCmnClassifierRuleIpv6FlowLabel OBJECT-TYPE
15     SYNTAX      Wman2IfIpv6FlowLabel
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19        "The value of this field specifies the matching values for
20        the IPv6 Flow label field."
21        ::= { wman2IfCmnClassifierRuleEntry 26 }
22
23
24
25 wman2IfCmnClassifierRuleBitMap OBJECT-TYPE
26     SYNTAX      Wman2IfClassifierBitMap
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30        "This object indicates which parameter encodings were
31        actually present in the entry. A bit set to '1' indicates
32        the corresponding classifier encoding is present, and '0'
33        means otherwise"
34        ::= { wman2IfCmnClassifierRuleEntry 27 }
35
36
37
38 wman2IfCmnPhsRuleTable OBJECT-TYPE
39     SYNTAX      SEQUENCE OF Wman2IfCmnPhsRuleEntry
40     MAX-ACCESS  not-accessible
41     STATUS      current
42     DESCRIPTION
43        "This table contains PHS rule dictionary entries. Each
44        entry contains the data of the header to be suppressed
45        along with its identification - PHSI. The classifier
46        uniquely maps packets to its associated PHS Rule. The
47        receiving entity uses the CID and the PHSI to restore the
48        PHSF. Once a PHSF has been assigned to a PHSI, it shall
49        not be changed. To change the value of a PHSF on a
50        service flow, a new PHS rule shall be defined, the old
51        rule is removed from the service flow, and the new rule
52        is added. When a classifier is deleted, any associated
53        PHS rule shall also be deleted."
54
55     REFERENCE
56        "Subclause 5.2.3 in IEEE Std 802.16-2004"
57        ::= { wman2IfCmnPacketCs 2 }
58
59
60
61
62 wman2IfCmnPhsRuleEntry OBJECT-TYPE
63     SYNTAX      Wman2IfCmnPhsRuleEntry
64     MAX-ACCESS  not-accessible
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table provides one row for each PHS rule created
4          dynamically by the BS and SS on a given service flow. The
5          PHS rule is defined by the pair (PHSS, PHSM) for each
6          distinct header data. It is indexed by IfIndex,
7          wman2IfCmnCpsSfId, and wman2IfCmnPhsIndex. The table is
8          read-only for NMS. "
9
10     INDEX      { ifIndex, wman2IfCmnCpsSfId,
11                  wman2IfCmnPhsRulePhsIndex }
12
13     ::= { wman2IfCmnPhsRuleTable 1 }
14
15     Wman2IfCmnPhsRuleEntry ::= SEQUENCE {
16         wman2IfCmnPhsRulePhsIndex      INTEGER,
17         wman2IfCmnPhsRulePhsField      OCTET STRING,
18         wman2IfCmnPhsRulePhsMask      OCTET STRING,
19         wman2IfCmnPhsRulePhsSize      Integer32,
20         wman2IfCmnPhsRulePhsVerify    Wman2IfPhsRuleVerify}
21
22
23
24     wman2IfCmnPhsRulePhsIndex OBJECT-TYPE
25         SYNTAX      INTEGER (1..255)
26         MAX-ACCESS  not-accessible
27         STATUS      current
28         DESCRIPTION
29             "The PHSI (PHS Index) has a value between 1 and 255, which
30             uniquely references the suppressed byte string. The index
31             is unique per service flow. The uplink and downlink PHSI
32             values are independent of each other."
33         REFERENCE
34             "Subclause 11.13.19.3.7.1 in IEEE Std 802.16-2004"
35         ::= { wman2IfCmnPhsRuleEntry 1 }
36
37
38
39     wman2IfCmnPhsRulePhsField OBJECT-TYPE
40         SYNTAX      OCTET STRING (SIZE(0..65535))
41         MAX-ACCESS  read-only
42         STATUS      current
43         DESCRIPTION
44             "The PHSF (PHS Field) is a string of bytes containing the
45             header information to be suppressed by the sending CS and
46             reconstructed by the receiving CS. The most significant
47             byte of the string corresponds to the first byte of the
48             CS-SDU."
49         REFERENCE
50             "Subclause 11.13.19.3.7.2 in IEEE Std 802.16-2004"
51         ::= { wman2IfCmnPhsRuleEntry 2 }
52
53
54
55
56     wman2IfCmnPhsRulePhsMask OBJECT-TYPE
57         SYNTAX      OCTET STRING (SIZE(0..65535))
58         MAX-ACCESS  read-only
59         STATUS      current
60         DESCRIPTION
61             "The PHSM An 8-bit mask that indicates which bytes in the
62             PHS Field (PHSF) to suppress and which bytes to not
63             suppress. The PHSM allows fields, such as sequence numbers
64
65

```

```

1      or checksums (which vary in value), to be excluded from
2      suppression with the constant bytes around them suppressed.
3      It is encoded as follows:
4      bit 0:
5          0 = don't suppress the 1st byte of the suppression field
6          1 = suppress first byte of the suppression field
7      bit 1:
8          0 = don't suppress the 2nd byte of the suppression field
9          1 = suppress second byte of the suppression field
10     bit x:
11         0 = don't suppress the (x+1) byte of the suppression
12         field
13         1 = suppress (x+1) byte of the suppression field
14     where the length of the octet string is ceiling
15     (wman2IfCmnPhsRulePhsSize/8)."
16
17 REFERENCE
18     "Subclause 11.13.19.3.7.3 in IEEE Std 802.16-2004"
19 ::= { wman2IfCmnPhsRuleEntry 3 }
20
21 wman2IfCmnPhsRulePhsSize OBJECT-TYPE
22     SYNTAX      Integer32 (0..255)
23     UNITS       "byte"
24     MAX-ACCESS  read-only
25     STATUS      current
26     DESCRIPTION
27         "The value of this field - PHSS is the total number of bytes
28         in the header to be suppressed and then restored in a
29         service flow that uses PHS."
30     REFERENCE
31         "Subclause 11.13.19.3.7.4 in IEEE Std 802.16-2004"
32     DEFVAL      {0}
33     ::= { wman2IfCmnPhsRuleEntry 4 }
34
35 wman2IfCmnPhsRulePhsVerify OBJECT-TYPE
36     SYNTAX      Wman2IfPhsRuleVerify
37     MAX-ACCESS  read-only
38     STATUS      current
39     DESCRIPTION
40         "The value of this field indicates to the sending entity
41         whether or not the packet header contents are to be
42         verified prior to performing suppression."
43     DEFVAL      { phsVerifyEnable }
44     ::= { wman2IfCmnPhsRuleEntry 5 }
45
46 --
47 -- wman2IfCmnCps contain the Common Part Sublayer objects that are
48 -- common to both Base Station and Subscriber Station
49 --
50 wman2IfCmnCps OBJECT IDENTIFIER ::= { wman2IfCommonObjects 2 }
51
52 wman2IfCmnCpsServiceFlowTable OBJECT-TYPE
53     SYNTAX      SEQUENCE OF Wman2IfCmnCpsServiceFlowEntry
54     MAX-ACCESS  not-accessible
55     STATUS      current

```

```

1      DESCRIPTION
2          "This table contains Service Flow managed objects that
3          are common in BS and SS."
4      ::= { wman2IfCmnCps 1 }
5
6
7      wman2IfCmnCpsServiceFlowEntry OBJECT-TYPE
8          SYNTAX      Wman2IfCmnCpsServiceFlowEntry
9          MAX-ACCESS   not-accessible
10         STATUS      current
11         DESCRIPTION
12             "This table provides one row for each created service
13             flow for a given MacAddress, and is indexed by ifIndex,
14             wman2IfCmnCpsCpsSfMacAddress, and wman2IfCmnCpsSfId.
15             IfIndex is associated with the BS sector."
16         INDEX       { ifIndex, wman2IfCmnCpsSfMacAddress,
17                     wman2IfCmnCpsSfId }
18         ::= { wman2IfCmnCpsServiceFlowTable 1 }
19
20
21
22
23      Wman2IfCmnCpsServiceFlowEntry ::= SEQUENCE {
24          wman2IfCmnCpsSfMacAddress      MacAddress,
25          wman2IfCmnCpsSfId              Unsigned32,
26          wman2IfCmnCpsSfCid             Wman2IfCidType,
27          wman2IfCmnCpsSfDirection       INTEGER,
28          wman2IfCmnCpsSfState           Wman2IfSfState,
29          wman2IfCmnCpsTrafficPriority    INTEGER,
30          wman2IfCmnCpsMaxSustainedRate   Unsigned32,
31          wman2IfCmnCpsMaxTrafficBurst    Unsigned32,
32          wman2IfCmnCpsMinReservedRate    Unsigned32,
33          wman2IfCmnCpsToleratedJitter    Unsigned32,
34          wman2IfCmnCpsMaxLatency         Unsigned32,
35          wman2IfCmnCpsFixedVsVariableSduInd INTEGER,
36          wman2IfCmnCpsSduSize            Unsigned32,
37          wman2IfCmnCpsSfSchedulingType   Wman2IfSfSchedulingType,
38          wman2IfCmnCpsArqEnable          TruthValue,
39          wman2IfCmnCpsArqWindowSize      INTEGER,
40          wman2IfCmnCpsArqBlockLifetime   INTEGER,
41          wman2IfCmnCpsArqSyncLossTimeout INTEGER,
42          wman2IfCmnCpsArqDeliverInOrder  TruthValue,
43          wman2IfCmnCpsArqRxPurgeTimeout  INTEGER,
44          wman2IfCmnCpsArqBlockSize       INTEGER,
45          wman2IfCmnCpsMinRsvdTolerableRate Unsigned32,
46          wman2IfCmnCpsReqTxPolicy        BITS,
47          wman2IfCmnSfCsSpecification     Wman2IfCsSpecification,
48          wman2IfCmnCpsTargetSaid         INTEGER}
49
50
51
52
53
54
55      wman2IfCmnCpsSfMacAddress OBJECT-TYPE
56          SYNTAX      MacAddress
57          MAX-ACCESS   not-accessible
58          STATUS      current
59          DESCRIPTION
60              "When this table is implemented on the basestation, this
61              object contains the SS Mac address, the reported service
62              flow was created for. On the SS, the value returned is
63              the SS's own Mac address."
64
65

```



```

1      ::= { wman2IfCmnCpsServiceFlowEntry 1 }
2
3
4  wman2IfCmnCpsSfId OBJECT-TYPE
5      SYNTAX      Unsigned32 ( 1 .. 4294967295)
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "A 32 bit quantity that uniquely identifies a service flow
10         to both the subscriber station and base station (BS)."
```

```

11      ::= { wman2IfCmnCpsServiceFlowEntry 2 }
12
13
14  wman2IfCmnCpsSfCid OBJECT-TYPE
15      SYNTAX      Wman2IfCidType
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "A 16 bit channel identifier to identify the connection
20         being created by DSA."
21      ::= { wman2IfCmnCpsServiceFlowEntry 3 }
22
23
24
25  wman2IfCmnCpsSfDirection OBJECT-TYPE
26      SYNTAX      INTEGER {downstream(1),
27                        upstream(2)}
28      MAX-ACCESS  read-only
29      STATUS      current
30      DESCRIPTION
31          "An attribute indicating the service flow is downstream or
32         upstream."
33      ::= { wman2IfCmnCpsServiceFlowEntry 4 }
34
35
36
37  wman2IfCmnCpsSfState OBJECT-TYPE
38      SYNTAX      Wman2IfSfState
39      MAX-ACCESS  read-only
40      STATUS      current
41      DESCRIPTION
42          "wman2IfCmnCpsSfState indicates the service flow state:
43         Authorized (1), Admitted (2), and Active (3) service
44         flow state."
45      REFERENCE
46          "Subclause 6.3.14.6, in IEEE Std 802.16-2004"
47      ::= { wman2IfCmnCpsServiceFlowEntry 5 }
48
49
50
51  wman2IfCmnCpsTrafficPriority OBJECT-TYPE
52      SYNTAX      INTEGER (0 .. 7)
53      MAX-ACCESS  read-only
54      STATUS      current
55      DESCRIPTION
56          "The value of this parameter specifies the priority
57         assigned to a service flow. For uplink service flows,
58         the BS should use this parameter when determining
59         precedence in request service and grant generation,
60         and the SS shall preferentially select contention
61         Request opportunities for Priority Request CIDs
62         based on this priority"
63
64
65

```

```

1      REFERENCE
2          "Subclause 11.13.5 in IEEE Std 802.16-2004"
3      ::= { wman2IfCmnCpsServiceFlowEntry 6 }
4
5
6      wman2IfCmnCpsMaxSustainedRate OBJECT-TYPE
7          SYNTAX      Unsigned32
8          UNITS        "b/s"
9          MAX-ACCESS   read-only
10         STATUS      current
11         DESCRIPTION
12             "This parameter defines the peak information rate
13              of the service. The rate is expressed in bits per
14              second and pertains to the SDUs at the input to
15              the system."
16         REFERENCE
17             "Subclause 11.13.6 in IEEE Std 802.16-2004"
18         ::= { wman2IfCmnCpsServiceFlowEntry 7 }
19
20
21      wman2IfCmnCpsMaxTrafficBurst OBJECT-TYPE
22          SYNTAX      Unsigned32
23          UNITS        "byte"
24          MAX-ACCESS   read-only
25          STATUS      current
26          DESCRIPTION
27              "This parameter defines the maximum burst size that
28               must be accommodated for the service."
29          REFERENCE
30              "Subclause 11.13.7 in IEEE Std 802.16-2004"
31          ::= { wman2IfCmnCpsServiceFlowEntry 8 }
32
33
34      wman2IfCmnCpsMinReservedRate OBJECT-TYPE
35          SYNTAX      Unsigned32
36          UNITS        "byte"
37          MAX-ACCESS   read-only
38          STATUS      current
39          DESCRIPTION
40              "This parameter specifies the minimum rate reserved
41               for this service flow."
42          REFERENCE
43              "Subclause 11.13.8 in IEEE Std 802.16-2004"
44          ::= { wman2IfCmnCpsServiceFlowEntry 9 }
45
46
47      wman2IfCmnCpsToleratedJitter OBJECT-TYPE
48          SYNTAX      Unsigned32
49          UNITS        "millisecond"
50          MAX-ACCESS   read-only
51          STATUS      current
52          DESCRIPTION
53              "This parameter defines the Maximum delay
54               variation (jitter) for the connection."
55          REFERENCE
56              "Subclause 11.13.13 in IEEE Std 802.16-2004"
57          ::= { wman2IfCmnCpsServiceFlowEntry 10 }
58
59
60
61
62
63
64
65

```

```

1  wman2IfCmnCpsMaxLatency OBJECT-TYPE
2      SYNTAX      Unsigned32
3      UNITS       "millisecond"
4      MAX-ACCESS  read-only
5      STATUS      current
6      DESCRIPTION
7          "The value of this parameter specifies the maximum
8          latency between the reception of a packet by the BS
9          or SS on its network interface and the forwarding
10         of the packet to its RF Interface."
11
12     REFERENCE
13         "Subclause 11.13.14 in IEEE Std 802.16-2004"
14     ::= { wman2IfCmnCpsServiceFlowEntry 11 }
15
16
17  wman2IfCmnCpsFixedVsVariableSduInd OBJECT-TYPE
18      SYNTAX      INTEGER {variableLength(0),
19                          fixedLength(1)}
20      MAX-ACCESS  read-only
21      STATUS      current
22      DESCRIPTION
23          "The value of this parameter specifies whether the SDUs
24          on the service flow are variable-length (0) or
25          fixed-length (1). The parameter is used only if
26          packing is on for the service flow. The default value
27          is 0, i.e., variable-length SDUs."
28
29     REFERENCE
30         "Subclause 11.13.15 in IEEE Std 802.16-2004"
31     DEFVAL      { variableLength }
32     ::= { wman2IfCmnCpsServiceFlowEntry 12 }
33
34
35  wman2IfCmnCpsSduSize OBJECT-TYPE
36      SYNTAX      Unsigned32
37      UNITS       "byte"
38      MAX-ACCESS  read-only
39      STATUS      current
40      DESCRIPTION
41          "The value of this parameter specifies the length of the
42          SDU for a fixed-length SDU service flow. This parameter
43          is used only if packing is on and the service flow is
44          indicated as carrying fixed-length SDUs. The default
45          value is 49 bytes, i.e., VC-switched ATM cells with PHS.
46          The parameter is relevant for both ATM and Packet
47          Convergence Sublayers."
48
49     REFERENCE
50         "Subclause 11.13.16 in IEEE Std 802.16-2004"
51     DEFVAL      { 49 }
52     ::= { wman2IfCmnCpsServiceFlowEntry 13 }
53
54
55  wman2IfCmnCpsSfSchedulingType OBJECT-TYPE
56      SYNTAX      Wman2IfSfSchedulingType
57      MAX-ACCESS  read-only
58      STATUS      current
59      DESCRIPTION
60          "Specifies the upstream scheduling service used for

```

```

1         upstream service flow. If the referenced parameter
2         is not present in the corresponding 802.16 QOS
3         Parameter Set of an upstream service flow, the
4         default value of this object is bestEffort(2).
5     REFERENCE
6         "Subclause 11.13.11 in IEEE Std 802.16-2004"
7     DEFVAL      { bestEffort }
8     ::= { wman2IfCmnCpsServiceFlowEntry 14 }
9
10
11
12 wman2IfCmnCpsArqEnable OBJECT-TYPE
13     SYNTAX      TruthValue
14     MAX-ACCESS  read-only
15     STATUS      current
16     DESCRIPTION
17         "True(1) ARQ enabling is requested for the connection."
18     ::= { wman2IfCmnCpsServiceFlowEntry 15 }
19
20
21
22 wman2IfCmnCpsArqWindowSize OBJECT-TYPE
23     SYNTAX      INTEGER (1..1024)
24     MAX-ACCESS  read-only
25     STATUS      current
26     DESCRIPTION
27         "Indicates the maximum number of unacknowledged
28         fragments at any time."
29     ::= { wman2IfCmnCpsServiceFlowEntry 16 }
30
31
32
33 wman2IfCmnCpsArqBlockLifetime OBJECT-TYPE
34     SYNTAX      INTEGER (0 .. 65535)
35     UNITS       "10 us"
36     MAX-ACCESS  read-only
37     STATUS      current
38     DESCRIPTION
39         "The maximum time interval an ARQ fragment will be
40         managed by the transmitter ARQ machine, once
41         initial transmission of the fragment has occurred.
42         If transmission or retransmission of the fragment
43         is not acknowledged by the receiver before the
44         time limit is reached, the fragment is discarded.
45         A value of 0 means Infinite."
46     ::= { wman2IfCmnCpsServiceFlowEntry 17 }
47
48
49
50 wman2IfCmnCpsArqSyncLossTimeout OBJECT-TYPE
51     SYNTAX      INTEGER (0 .. 65535 )
52     UNITS       "10 us"
53     MAX-ACCESS  read-only
54     STATUS      current
55     DESCRIPTION
56         "The maximum interval before declaring a loss
57         of synchronization of the sender and receiver
58         state machines. A value of 0 means Infinite."
59     ::= { wman2IfCmnCpsServiceFlowEntry 18 }
60
61
62
63 wman2IfCmnCpsArqDeliverInOrder OBJECT-TYPE
64     SYNTAX      TruthValue
65

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "Indicates whether or not data is to be delivered
5          by the receiving MAC to its client application
6          in the order in which data was handed off to the
7          originating MAC."
8      ::= { wman2IfCmnCpsServiceFlowEntry 19 }
9
10
11
12  wman2IfCmnCpsArqRxPurgeTimeout OBJECT-TYPE
13      SYNTAX          INTEGER (0 .. 65535)
14      UNITS            "10 us"
15      MAX-ACCESS      read-only
16      STATUS          current
17      DESCRIPTION
18          "Indicates the time interval the ARQ window is advanced
19          after a fragment is received. A value of 0 means
20          Infinite."
21      ::= { wman2IfCmnCpsServiceFlowEntry 20 }
22
23
24
25  wman2IfCmnCpsArqBlockSize OBJECT-TYPE
26      SYNTAX          INTEGER (1..2040)
27      UNITS            "byte"
28      MAX-ACCESS      read-only
29      STATUS          current
30      DESCRIPTION
31          "This value of this parameter specifies the size of an
32          ARQ block. This parameter shall be established by
33          negotiation during the connection creation dialog."
34      REFERENCE
35          "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
36      ::= { wman2IfCmnCpsServiceFlowEntry 21 }
37
38
39
40
41  wman2IfCmnCpsMinRsvdTolerableRate OBJECT-TYPE
42      SYNTAX          Unsigned32
43      UNITS            "b/s"
44      MAX-ACCESS      read-only
45      STATUS          current
46      DESCRIPTION
47          "Minimum Tolerable Traffic Rate = R (bits/sec) with
48          time base T(sec) means the following. Let S denote
49          additional demand accumulated at the MAC SAP of the
50          transmitter during an arbitrary time interval of the
51          length T. Then the amount of data forwarded at the
52          receiver to CS (in bits) during this interval should
53          be not less than min {S, R * T}."
54      REFERENCE
55          "Subclause 11.13.9 in IEEE Std 802.16-2004"
56      ::= { wman2IfCmnCpsServiceFlowEntry 22 }
57
58
59
60
61  wman2IfCmnCpsReqTxPolicy OBJECT-TYPE
62      SYNTAX          BITS {noBroadcastBwReq(0),
63                          reserved1(1),
64                          noPiggybackReq(2),
65

```

```

1          noFragmentData(3),
2          noPHS(4),
3          noSduPacking(5),
4          noCrc(6),
5          reserved2(7)}
6
7      MAX-ACCESS    read-only
8      STATUS        current
9      DESCRIPTION
10         "The value of this parameter provides the capability to
11         specify certain attributes for the associated service
12         flow. An attribute is enabled by setting the
13         corresponding bit position to 1."
14
15      REFERENCE
16         "Subclause 11.13.12 in IEEE Std 802.16-2004"
17      ::= { wman2IfCmnCpsServiceFlowEntry 23 }
18
19
20      wman2IfCmnSfCsSpecification OBJECT-TYPE
21          SYNTAX      Wman2IfCsSpecification
22          MAX-ACCESS  read-only
23          STATUS      current
24          DESCRIPTION
25             "This parameter specifies the convergence sublayer
26             encapsulation mode."
27
28          REFERENCE
29             "Subclause 11.13.19.1 in IEEE Std 802.16-2004"
30          ::= { wman2IfCmnCpsServiceFlowEntry 24 }
31
32
33      wman2IfCmnCpsTargetSaid OBJECT-TYPE
34          SYNTAX      INTEGER (0 .. 65535)
35          MAX-ACCESS  read-only
36          STATUS      current
37          DESCRIPTION
38             "The target SAID parameter indicates the SAID onto
39             which the service flow being set up shall be mapped."
40
41          REFERENCE
42             "Subclause 11.13.17 in IEEE Std 802.16-2004"
43          ::= { wman2IfCmnCpsServiceFlowEntry 25 }
44
45
46      --
47      -- wman2IfCmnBsSsConfigurationTable contains global parameters
48      -- common in BS and SS
49      --
50
51      wman2IfCmnBsSsConfigurationTable OBJECT-TYPE
52          SYNTAX      SEQUENCE OF Wman2IfCmnBsSsConfigurationEntry
53          MAX-ACCESS  not-accessible
54          STATUS      current
55          DESCRIPTION
56             "This table provides one row for each BS sector that
57             contains the system parameters common in both SS and
58             BS. All SSs shall have the same parameters as the BS
59             to which the SSs are associated."
60
61          REFERENCE
62             "Subclause 10.1 in IEEE Std 802.16-2004"
63          ::= { wman2IfCmnCps 2 }
64
65

```

```

1
2 wman2IfCmnBsSsConfigurationEntry OBJECT-TYPE
3     SYNTAX      Wman2IfCmnBsSsConfigurationEntry
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "This table is indexed by ifIndex, indicating BS
8         sector."
9     INDEX        { ifIndex }
10    ::= { wman2IfCmnBsSsConfigurationTable 1 }
11
12
13
14 Wman2IfCmnBsSsConfigurationEntry ::= SEQUENCE {
15     wman2IfCmnInvitedRangRetries      INTEGER,
16     wman2IfCmnDSxReqRetries           Unsigned32,
17     wman2IfCmnDSxRespRetries          Unsigned32,
18     wman2IfCmnT7Timeout               INTEGER,
19     wman2IfCmnT8Timeout               INTEGER,
20     wman2IfCmnT10Timeout              INTEGER,
21     wman2IfCmnT22Timeout              INTEGER}
22
23
24
25 wman2IfCmnInvitedRangRetries OBJECT-TYPE
26     SYNTAX      INTEGER (16..65535)
27     MAX-ACCESS  read-write
28     STATUS      current
29     DESCRIPTION
30         "Number of retries on inviting Ranging Requests."
31     ::= { wman2IfCmnBsSsConfigurationEntry 1 }
32
33
34
35 wman2IfCmnDSxReqRetries OBJECT-TYPE
36     SYNTAX      Unsigned32
37     MAX-ACCESS  read-write
38     STATUS      current
39     DESCRIPTION
40         "Number of Timeout Retries on DSA/DSC/DSD Requests."
41     DEFVAL      { 3 }
42     ::= { wman2IfCmnBsSsConfigurationEntry 2 }
43
44
45
46 wman2IfCmnDSxRespRetries OBJECT-TYPE
47     SYNTAX      Unsigned32
48     MAX-ACCESS  read-write
49     STATUS      current
50     DESCRIPTION
51         "Number of Timeout Retries on DSA/DSC/DSD Responses."
52     DEFVAL      { 3 }
53     ::= { wman2IfCmnBsSsConfigurationEntry 3 }
54
55
56
57 wman2IfCmnT7Timeout OBJECT-TYPE
58     SYNTAX      INTEGER (0 .. 1000)
59     UNITS        "milliseconds"
60     MAX-ACCESS  read-write
61     STATUS      current
62     DESCRIPTION
63         "Wait for DSA/DSC/DSD Response Timeout in ms."
64     ::= { wman2IfCmnBsSsConfigurationEntry 4 }
65

```

```

1
2 wman2IfCmnT8Timeout OBJECT-TYPE
3     SYNTAX      INTEGER (0 .. 300)
4     UNITS       "milliseconds"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "Wait for DSA/DSC/DSD Acknowledge Timeout in ms."
9     ::= { wman2IfCmnBsSsConfigurationEntry 5 }
10
11
12
13 wman2IfCmnT10Timeout OBJECT-TYPE
14     SYNTAX      INTEGER (0 .. 3000)
15     UNITS       "milliseconds"
16     MAX-ACCESS  read-write
17     STATUS      current
18     DESCRIPTION
19         "Wait for Transaction End timeout in ms."
20     ::= { wman2IfCmnBsSsConfigurationEntry 6 }
21
22
23
24 wman2IfCmnT22Timeout OBJECT-TYPE
25     SYNTAX      INTEGER (0 .. 500)
26     UNITS       "milliseconds"
27     MAX-ACCESS  read-write
28     STATUS      current
29     DESCRIPTION
30         "Wait for ARQ Reset in ms."
31     ::= { wman2IfCmnBsSsConfigurationEntry 7 }
32
33
34 -- Common PKM group
35 -- wman2IfCmnPkmObjects contain the Privacy Sublayer objects that are
36 -- common to both Base Station and Subscriber Station
37 --
38
39 wman2IfCmnPkmObjects OBJECT IDENTIFIER ::= { wman2IfCommonObjects 3 }
40
41
42 --
43 -- Table wman2IfCmnCryptoSuiteTable
44 --
45
46 wman2IfCmnCryptoSuiteTable OBJECT-TYPE
47     SYNTAX      SEQUENCE OF Wman2IfCmnCryptoSuiteEntry
48     MAX-ACCESS  not-accessible
49     STATUS      current
50     DESCRIPTION
51         "This table describes the PKM cryptographic suite
52         capabilites for each SS or BS wireless interface."
53     ::= { wman2IfCmnPkmObjects 1 }
54
55
56 wman2IfCmnCryptoSuiteEntry OBJECT-TYPE
57     SYNTAX      Wman2IfCmnCryptoSuiteEntry
58     MAX-ACCESS  not-accessible
59     STATUS      current
60     DESCRIPTION
61         "Each entry contains the cryptographic suite pair that SS
62         or BS supports."
63     INDEX      { ifIndex, wman2IfCmnCryptoSuiteIndex }
64
65

```



```

1      ::= { wman2IfCmnCryptoSuiteTable 1 }
2
3
4  Wman2IfCmnCryptoSuiteEntry ::= SEQUENCE {
5      wman2IfCmnCryptoSuiteIndex      Integer32,
6      wman2IfCmnCryptoSuiteDataEncryptAlg  Wman2IfDataEncryptAlgId,
7      wman2IfCmnCryptoSuiteDataAuthentAlg  Wman2IfDataAuthAlgId,
8      wman2IfCmnCryptoSuiteTekEncryptAlg    Wman2IfTekEncryptAlgId}
9
10
11  wman2IfCmnCryptoSuiteIndex OBJECT-TYPE
12      SYNTAX      Integer32 (1 .. 1000)
13      MAX-ACCESS  not-accessible
14      STATUS      current
15      DESCRIPTION
16          "The index for a cryptographic suite row."
17      ::= { wman2IfCmnCryptoSuiteEntry 1 }
18
19
20  wman2IfCmnCryptoSuiteDataEncryptAlg OBJECT-TYPE
21      SYNTAX      Wman2IfDataEncryptAlgId
22      MAX-ACCESS  read-only
23      STATUS      current
24      DESCRIPTION
25          "The value of this object is the data encryption algorithm
26          for this cryptographic suite capability."
27      REFERENCE
28          "Table 375, IEEE Std 802.16-2004"
29      ::= { wman2IfCmnCryptoSuiteEntry 2 }
30
31
32
33  wman2IfCmnCryptoSuiteDataAuthentAlg OBJECT-TYPE
34      SYNTAX      Wman2IfDataAuthAlgId
35      MAX-ACCESS  read-only
36      STATUS      current
37      DESCRIPTION
38          "The value of this object is the data authentication
39          algorithm for this cryptographic suite capability."
40      REFERENCE
41          "Table 376, IEEE Std 802.16-2004"
42      ::= { wman2IfCmnCryptoSuiteEntry 3 }
43
44
45
46  wman2IfCmnCryptoSuiteTekEncryptAlg OBJECT-TYPE
47      SYNTAX      Wman2IfTekEncryptAlgId
48      MAX-ACCESS  read-only
49      STATUS      current
50      DESCRIPTION
51          "The value of this object is the TEK key encryption
52          algorithm for this cryptographic suite capability."
53      REFERENCE
54          "Table 377, IEEE Std 802.16-2004"
55      ::= { wman2IfCmnCryptoSuiteEntry 4 }
56
57
58
59
60  --
61  -- Conformance Information
62  --
63  wman2IfMibConformance OBJECT IDENTIFIER ::= {wman2IfMib 2}
64  wman2IfMibGroups      OBJECT IDENTIFIER ::= {wman2IfMibConformance 1}
65

```

```

1  wman2IfMibCompliances OBJECT IDENTIFIER ::= {wman2IfMibConformance 2}
2
3  -- compliance statements
4  wman2IfMibCompliance MODULE-COMPLIANCE
5      STATUS          current
6      DESCRIPTION
7          "The compliance statement for devices that implement
8          Wireless MAN interfaces as defined in IEEE Std 802.16-2004."
9
10
11  MODULE -- wman2IfMib
12
13
14  MANDATORY-GROUPS          -- unconditionally mandatory groups
15      { wman2IfMibCommonGroup }
16
17
18  GROUP wman2IfMibQoSGroup   -- unconditionally mandatory group
19  DESCRIPTION
20      "This group is mandatory for Base Station and subscriber
21      station."
22
23
24  GROUP wman2IfMibBsGroup    -- conditionally mandatory group
25  DESCRIPTION
26      "This group is mandatory for Base Station."
27
28
29  GROUP wman2IfMibBsAasGroup -- optional group
30  DESCRIPTION
31      "This group is mandatory for Base Station."
32
33
34  GROUP wman2IfMibSsGroup    -- conditionally mandatory group
35  DESCRIPTION
36      "This group is mandatory for Subscriber Station."
37
38
39  GROUP wman2IfMibBsOfdmGroup -- conditionally mandatory group
40  DESCRIPTION
41      "This group is mandatory for Base Station
42      implementaing the OFDM PHY."
43
44
45  GROUP wman2IfMibSsOfdmGroup -- conditionally mandatory group
46  DESCRIPTION
47      "This group is mandatory for Subscriber Station
48      implementing the OFDM PHY."
49
50
51  GROUP wman2IfMibBsOfdmaGroup -- conditionally mandatory group
52  DESCRIPTION
53      "This group is mandatory for Base Station
54      implementaing the OFDMA PHY."
55
56
57  GROUP wman2IfMibSsOfdmaGroup -- conditionally mandatory group
58  DESCRIPTION
59      "This group is mandatory for Subscriber Station
60      implementing the OFDMA PHY."
61
62
63  GROUP wman2IfMibBsNotificationGroup -- unconditionally
64      -- mandatory groups
65  DESCRIPTION

```

```

1         "This group is mandatory for Base Station."
2
3     GROUP wman2IfMibSsNotificationGroup -- optional group
4     DESCRIPTION
5         "This group is optional for Subscriber Station."
6
7
8     GROUP wman2IfMibCmnPhsGroup -- optional group
9     DESCRIPTION
10        "This group is optional for Base Station and
11        Subscriber Station."
12
13
14    GROUP wman2IfMibBsPhsGroup -- optional group
15    DESCRIPTION
16        "This group is optional for Base Station."
17    ::= { wman2IfMibCompliances 1 }
18
19
20 wman2IfMibCommonGroup      OBJECT-GROUP
21     OBJECTS {-- Classification
22         wman2IfCmnClassifierRulePriority,
23         wman2IfCmnClassifierRuleIpTosLow,
24         wman2IfCmnClassifierRuleIpTosHigh,
25         wman2IfCmnClassifierRuleIpTosMask,
26         wman2IfCmnClassifierRuleIpProtocol,
27         wman2IfCmnClassifierRuleIpSourceAddr,
28         wman2IfCmnClassifierRuleIpSourceMask,
29         wman2IfCmnClassifierRuleIpDestAddr,
30         wman2IfCmnClassifierRuleIpDestMask,
31         wman2IfCmnClassifierRuleSourcePortStart,
32         wman2IfCmnClassifierRuleSourcePortEnd,
33         wman2IfCmnClassifierRuleDestPortStart,
34         wman2IfCmnClassifierRuleDestPortEnd,
35         wman2IfCmnClassifierRuleDestMacAddr,
36         wman2IfCmnClassifierRuleDestMacMask,
37         wman2IfCmnClassifierRuleSourceMacAddr,
38         wman2IfCmnClassifierRuleSourceMacMask,
39         wman2IfCmnClassifierRuleEnetProtocolType,
40         wman2IfCmnClassifierRuleEnetProtocol,
41         wman2IfCmnClassifierRuleUserPriLow,
42         wman2IfCmnClassifierRuleUserPriHigh,
43         wman2IfCmnClassifierRuleVlanId,
44         wman2IfCmnClassifierRuleState,
45         wman2IfCmnClassifierRulePkts,
46         wman2IfCmnClassifierRuleIpv6FlowLabel,
47         wman2IfCmnClassifierRuleBitMap,
48
49         -- Configuration parameters
50         wman2IfCmnCpsTargetSaid,
51         wman2IfCmnInvitedRangRetries,
52         wman2IfCmnDSxReqRetries,
53         wman2IfCmnDSxRespRetries,
54         wman2IfCmnT7Timeout,
55         wman2IfCmnT8Timeout,
56         wman2IfCmnT10Timeout,
57         wman2IfCmnT22Timeout,
58
59
60
61
62
63
64
65

```

```

1          wman2IfCmnCryptoSuiteDataEncryptAlg,
2          wman2IfCmnCryptoSuiteDataAuthentAlg,
3          wman2IfCmnCryptoSuiteTekEncryptAlg}
4
5      STATUS          current
6
7      DESCRIPTION
8          "This group contains objects for both BS and SS,
9          and are independent of PHY."
10     ::= { wman2IfMibGroups 1 }
11
12 wman2IfMibQoSGroup      OBJECT-GROUP
13     OBJECTS {wman2IfCmnCpsSfId,
14             wman2IfCmnCpsSfCid,
15             wman2IfCmnCpsSfDirection,
16             wman2IfCmnCpsSfState,
17             wman2IfCmnCpsTrafficPriority,
18             wman2IfCmnCpsMaxSustainedRate,
19             wman2IfCmnCpsMaxTrafficBurst,
20             wman2IfCmnCpsMinReservedRate,
21             wman2IfCmnCpsToleratedJitter,
22             wman2IfCmnCpsMaxLatency,
23             wman2IfCmnCpsFixedVsVariableSduInd,
24             wman2IfCmnCpsSduSize,
25             wman2IfCmnCpsSfSchedulingType,
26             wman2IfCmnCpsArqEnable,
27             wman2IfCmnCpsArqWindowSize,
28             wman2IfCmnCpsArqBlockLifetime,
29             wman2IfCmnCpsArqSyncLossTimeout,
30             wman2IfCmnCpsArqDeliverInOrder,
31             wman2IfCmnCpsArqRxPurgeTimeout,
32             wman2IfCmnCpsArqBlockSize,
33             wman2IfCmnCpsMinRsvdTolerableRate,
34             wman2IfCmnCpsReqTxPolicy,
35             wman2IfCmnSfCsSpecification}
36
37     STATUS          current
38
39     DESCRIPTION
40         "This group contains QoS objects for both BS and SS."
41     ::= { wman2IfMibGroups 2 }
42
43
44
45
46 wman2IfMibBsGroup      OBJECT-GROUP
47     OBJECTS {-- Service classes
48             wman2IfBsSfDirection,
49             wman2IfBsServiceClassIndex,
50             wman2IfBsSfState,
51             wman2IfBsSfProvisionedTime,
52             wman2IfBsProvisionedSfRowStatus,
53             wman2IfBsSsProvisionedForSfRowStatus,
54             wman2IfBsSfCsSpecification,
55             wman2IfBsQoSServiceClassName,
56             wman2IfBsQoSSTrafficPriority,
57             wman2IfBsQoSMaxSustainedRate,
58             wman2IfBsQoSMaxTrafficBurst,
59             wman2IfBsQoSMinReservedRate,
60             wman2IfBsQOSToleratedJitter,
61             wman2IfBsQoSMaxLatency,
62
63
64
65

```

```

1      wman2IfBsQoSFixedVsVariableSduInd,
2      wman2IfBsQoSsduSize,
3      wman2IfBsQoSScSchedulingType,
4      wman2IfBsQoSScArqEnable,
5      wman2IfBsQoSScArqWindowSize,
6      wman2IfBsQoSScArqBlockLifetime,
7      wman2IfBsQoSScArqSyncLossTimeout,
8      wman2IfBsQoSScArqDeliverInOrder,
9      wman2IfBsQoSScArqRxPurgeTimeout,
10     wman2IfBsQoSScArqBlockSize,
11     wman2IfBsQoSSCMinRsvdTolerableRate,
12     wman2IfBsQoSReqTxPolicy,
13     wman2IfBsQoSServiceClassRowStatus,
14
15     -- Classification
16     wman2IfBsClassifierRulePriority,
17     wman2IfBsClassifierRuleIpTosLow,
18     wman2IfBsClassifierRuleIpTosHigh,
19     wman2IfBsClassifierRuleIpTosMask,
20     wman2IfBsClassifierRuleIpProtocol,
21     wman2IfBsClassifierRuleIpSourceAddr,
22     wman2IfBsClassifierRuleIpSourceMask,
23     wman2IfBsClassifierRuleIpDestAddr,
24     wman2IfBsClassifierRuleIpDestMask,
25     wman2IfBsClassifierRuleSourcePortStart,
26     wman2IfBsClassifierRuleSourcePortEnd,
27     wman2IfBsClassifierRuleDestPortStart,
28     wman2IfBsClassifierRuleDestPortEnd,
29     wman2IfBsClassifierRuleDestMacAddr,
30     wman2IfBsClassifierRuleDestMacMask,
31     wman2IfBsClassifierRuleSourceMacAddr,
32     wman2IfBsClassifierRuleSourceMacMask,
33     wman2IfBsClassifierRuleEnetProtocolType,
34     wman2IfBsClassifierRuleEnetProtocol,
35     wman2IfBsClassifierRuleUserPriLow,
36     wman2IfBsClassifierRuleUserPriHigh,
37     wman2IfBsClassifierRuleVlanId,
38     wman2IfBsClassifierRuleState,
39     wman2IfBsClassifierRulePhsSize,
40     wman2IfBsClassifierRulePhsMask,
41     wman2IfBsClassifierRulePhsVerify,
42     wman2IfBsClassifierRuleIpv6FlowLabel,
43     wman2IfBsClassifierRuleBitMap,
44     wman2IfBsClassifierRuleRowStatus,
45
46     -- Packet counters
47     wman2IfBsSsMacSduCount,
48     wman2IfBsSsOctetCount,
49     wman2IfBsSsResetCounter,
50     wman2IfBsSsResetCounterTime,
51
52     -- Capability negotiation
53     wman2IfBsSsBasicCid,
54     wman2IfBsSsPrimaryCid,

```

```

1      wman2IfBsSsSecondaryCid,
2      wman2IfBsSsManagementSupport,
3      wman2IfBsSsIpManagementMode,
4      wman2IfBs2ndMgmtDlQoSProfileIndex,
5      wman2IfBs2ndMgmtUlQoSProfileIndex,
6      wman2IfBsAutoSfidEnabled,
7      wman2IfBsAutoSfidRangeMin,
8      wman2IfBsAutoSfidRangeMax,
9      wman2IfBsResetSector,
10     wman2IfBsSs2ndMgmtArqEnable,
11     wman2IfBsSs2ndMgmtArqWindowSize,
12     wman2IfBsSs2ndMgmtArqDnLinkTxDelay,
13     wman2IfBsSs2ndMgmtArqUpLinkTxDelay,
14     wman2IfBsSs2ndMgmtArqDnLinkRxDelay,
15     wman2IfBsSs2ndMgmtArqUpLinkRxDelay,
16     wman2IfBsSs2ndMgmtArqBlockLifetime,
17     wman2IfBsSs2ndMgmtArqSyncLossTimeout,
18     wman2IfBsSs2ndMgmtArqDeliverInOrder,
19     wman2IfBsSs2ndMgmtArqRxPurgeTimeout,
20     wman2IfBsSs2ndMgmtArqBlockSize,
21     wman2IfBsSsVendorIdEncoding,
22     wman2IfBsSsAasBroadcastPermission,
23     wman2IfBsSsMaxTxPowerBpsk,
24     wman2IfBsSsMaxTxPowerQpsk,
25     wman2IfBsSsMaxTxPower16Qam,
26     wman2IfBsSsMaxTxPower64Qam,
27
28     -- Configuration parameters
29     wman2IfBsSsMacVersion,
30     wman2IfBsDcdInterval,
31     wman2IfBsUcdInterval,
32     wman2IfBsUcdTransition,
33     wman2IfBsDcdTransition,
34     wman2IfBsInitialRangingInterval,
35     wman2IfBsSsULMapProcTime,
36     wman2IfBsSsRangRespProcTime,
37     wman2IfBsT5Timeout,
38     wman2IfBsT9Timeout,
39     wman2IfBsT13Timeout,
40     wman2IfBsT15Timeout,
41     wman2IfBsT17Timeout,
42     wman2IfBsT27IdleTimer,
43     wman2IfBsT27ActiveTimer,
44
45     -- Performance monitoring
46     wman2IfBsHistogramIndex,
47     wman2IfBsChannelNumber,
48     wman2IfBsStartFrame,
49     wman2IfBsDuration,
50     wman2IfBsBasicReport,
51     wman2IfBsMeanCinrReport,
52     wman2IfBsMeanRssiReport,
53     wman2IfBsStdDeviationCinrReport,
54     wman2IfBsStdDeviationRssiReport,
55
56
57
58
59
60
61
62
63
64
65

```

```

1
2      -- Capability negotiation
3      wman2IfBsSsReqCapUplinkCidSupport,
4      wman2IfBsSsReqCapArqSupport,
5      wman2IfBsSsReqCapDsxFowControl,
6      wman2IfBsSsReqCapMacCrcSupport,
7      wman2IfBsSsReqCapMcaFlowControl,
8      wman2IfBsSsReqCapMcpGroupCidSupport,
9      wman2IfBsSsReqCapPkmFlowControl,
10     wman2IfBsSsReqCapAuthPolicyControl,
11     wman2IfBsSsReqCapMaxNumOfSupportedSA,
12     wman2IfBsSsReqCapIpVersion,
13     wman2IfBsSsReqCapMacCsSupportBitMap,
14     wman2IfBsSsReqCapMaxNumOfClassifier,
15     wman2IfBsSsReqCapPhsSupport,
16     wman2IfBsSsReqCapBandwidthAllocSupport,
17     wman2IfBsSsReqCapPduConstruction,
18     wman2IfBsSsReqCapTtgTransitionGap,
19     wman2IfBsSsReqCapRtgTransitionGap,
20     wman2IfBsSsRspCapUplinkCidSupport,
21     wman2IfBsSsRspCapArqSupport,
22     wman2IfBsSsRspCapDsxFowControl,
23     wman2IfBsSsRspCapMacCrcSupport,
24     wman2IfBsSsRspCapMcaFlowControl,
25     wman2IfBsSsRspCapMcpGroupCidSupport,
26     wman2IfBsSsRspCapPkmFlowControl,
27     wman2IfBsSsRspCapAuthPolicyControl,
28     wman2IfBsSsRspCapMaxNumOfSupportedSA,
29     wman2IfBsSsRspCapIpVersion,
30     wman2IfBsSsRspCapMacCsSupportBitMap,
31     wman2IfBsSsRspCapMaxNumOfClassifier,
32     wman2IfBsSsRspCapPhsSupport,
33     wman2IfBsSsRspCapBandwidthAllocSupport,
34     wman2IfBsSsRspCapPduConstruction,
35     wman2IfBsSsRspCapTtgTransitionGap,
36     wman2IfBsSsRspCapRtgTransitionGap,
37     wman2IfBsCapUplinkCidSupport,
38     wman2IfBsCapArqSupport,
39     wman2IfBsCapDsxFowControl,
40     wman2IfBsCapMacCrcSupport,
41     wman2IfBsCapMcaFlowControl,
42     wman2IfBsCapMcpGroupCidSupport,
43     wman2IfBsCapPkmFlowControl,
44     wman2IfBsCapAuthPolicyControl,
45     wman2IfBsCapMaxNumOfSupportedSA,
46     wman2IfBsCapIpVersion,
47     wman2IfBsCapMacCsSupportBitMap,
48     wman2IfBsCapMaxNumOfClassifier,
49     wman2IfBsCapPhsSupport,
50     wman2IfBsCapBandwidthAllocSupport,
51     wman2IfBsCapPduConstruction,
52     wman2IfBsCapTtgTransitionGap,
53     wman2IfBsCapRtgTransitionGap,
54     wman2IfBsCapCfgUplinkCidSupport,
55
56
57
58
59
60
61
62
63
64
65

```

```

1      wman2IfBsCapCfgArqSupport,
2      wman2IfBsCapCfgDsxFLOWControl,
3      wman2IfBsCapCfgMacCrcSupport,
4      wman2IfBsCapCfgMcaFlowControl,
5      wman2IfBsCapCfgMcpGroupCidSupport,
6      wman2IfBsCapCfgPkmFlowControl,
7      wman2IfBsCapCfgAuthPolicyControl,
8      wman2IfBsCapCfgMaxNumOfSupportedSA,
9      wman2IfBsCapCfgIpVersion,
10     wman2IfBsCapCfgMacCsSupportBitMap,
11     wman2IfBsCapCfgMaxNumOfClassifier,
12     wman2IfBsCapCfgPhsSupport,
13     wman2IfBsCapCfgBandwidthAllocSupport,
14     wman2IfBsCapCfgPduConstruction,
15     wman2IfBsCapCfgTtgTransitionGap,
16     wman2IfBsCapCfgRtgTransitionGap,
17     wman2IfBsSsActionsResetSs,
18     wman2IfBsSsActionsAbortSs,
19     wman2IfBsSsActionsOverrideDnFreq,
20     wman2IfBsSsActionsOverrideChannelId,
21     wman2IfBsSsActionsDeReRegSs,
22     wman2IfBsSsActionsDeReRegSsCode,
23     wman2IfBsSsActionsRowStatus,
24
25     -- Privacy sublayer
26     wman2IfBsPkmDefaultAuthLifetime,
27     wman2IfBsPkmDefaultTekLifetime,
28     wman2IfBsPkmDefaultSelfSigManufCertTrust,
29     wman2IfBsPkmCheckCertValidityPeriods,
30     wman2IfBsPkmAuthentInfos,
31     wman2IfBsPkmAuthRequests,
32     wman2IfBsPkmAuthReplies,
33     wman2IfBsPkmAuthRejects,
34     wman2IfBsPkmAuthInvalids,
35     wman2IfBsSsPkmAuthKeySequenceNumber,
36     wman2IfBsSsPkmAuthExpiresOld,
37     wman2IfBsSsPkmAuthExpiresNew,
38     wman2IfBsSsPkmAuthLifetime,
39     wman2IfBsSsPkmAuthReset,
40     wman2IfBsSsPkmAuthInfos,
41     wman2IfBsSsPkmAuthRequests,
42     wman2IfBsSsPkmAuthReplies,
43     wman2IfBsSsPkmAuthRejects,
44     wman2IfBsSsPkmAuthInvalids,
45     wman2IfBsSsPkmAuthRejectErrorCode,
46     wman2IfBsSsPkmAuthRejectErrorString,
47     wman2IfBsSsPkmAuthInvalidErrorCode,
48     wman2IfBsSsPkmAuthInvalidErrorString,
49     wman2IfBsSsPkmAuthPrimarySAId,
50     wman2IfBsSsPkmAuthValidStatus,
51     wman2IfBsPkmTekSAType,
52     wman2IfBsPkmTekDataEncryptAlg,
53     wman2IfBsPkmTekDataAuthentAlg,
54     wman2IfBsPkmTekEncryptAlg,
55
56
57
58
59
60
61
62
63
64
65

```



```

1      wman2IfBsPkmTekLifetime,
2      wman2IfBsPkmTekKeySequenceNumber,
3      wman2IfBsPkmTekExpiresOld,
4      wman2IfBsPkmTekExpiresNew,
5      wman2IfBsPkmTekReset,
6      wman2IfBsPkmKeyRequests,
7      wman2IfBsPkmKeyReplies,
8      wman2IfBsPkmKeyRejects,
9      wman2IfBsPkmTekInvalids,
10     wman2IfBsPkmKeyRejectErrorCode,
11     wman2IfBsPkmKeyRejectErrorString,
12     wman2IfBsPkmTekInvalidErrorCode,
13     wman2IfBsPkmTekInvalidErrorString,
14
15     -- Notification
16     wman2IfBsTrapControlRegister,
17     wman2IfBsStatusTrapControlRegister,
18     wman2IfBsRssiLowThreshold,
19     wman2IfBsRssiHighThreshold,
20     wman2IfBsSsNotificationMacAddr,
21     wman2IfBsSsStatusValue,
22     wman2IfBsSsStatusInfo,
23     wman2IfBsDynamicServiceType,
24     wman2IfBsDynamicServiceFailReason,
25     wman2IfBsSsRssiStatus,
26     wman2IfBsSsRssiStatusInfo,
27     wman2IfBsSsRegisterStatus}
28
29     STATUS          current
30
31     DESCRIPTION
32         "This group contains objects for BS, and are
33         independent of PHY."
34     ::= { wman2IfMibGroups 3 }
35
36
37 wman2IfMibBsAasGroup      OBJECT-GROUP
38     OBJECTS {-- AAS Configuration parameters
39         wman2IfBsAasChanFbckReqFreq,
40         wman2IfBsAasBeamSelectFreq,
41         wman2IfBsAasChanFbckReqResolution,
42         wman2IfBsAasBeamReqResolution,
43         wman2IfBsAasNumOptDiversityZones}
44
45     STATUS          current
46
47     DESCRIPTION
48         "This group contains objects for AAS in BS."
49     ::= { wman2IfMibGroups 4 }
50
51
52 wman2IfMibSsGroup         OBJECT-GROUP
53     OBJECTS {-- Configuration parameters
54         wman2IfSsLostDLMapInterval,
55         wman2IfSsLostULMapInterval,
56         wman2IfSsContentionRangRetries,
57         wman2IfSsRequestRetries,
58         wman2IfSsRegRequestRetries,
59         wman2IfSsTftpBackoffStart,
60         wman2IfSsTftpBackoffEnd,
61

```

```

1      wman2IfSsTftpRequestRetries,
2      wman2IfSsTftpDownloadRetries,
3      wman2IfSsTftpWait,
4      wman2IfSsToDRetries,
5      wman2IfSsToDRetryPeriod,
6      wman2IfSsT1Timeout,
7      wman2IfSsT2Timeout,
8      wman2IfSsT3Timeout,
9      wman2IfSsT4Timeout,
10     wman2IfSsT6Timeout,
11     wman2IfSsT12Timeout,
12     wman2IfSsT14Timeout,
13     wman2IfSsT16Timeout,
14     wman2IfSsT18Timeout,
15     wman2IfSsT19Timeout,
16     wman2IfSsT20Timeout,
17     wman2IfSsT21Timeout,
18     wman2IfSsSBCRequestRetries,
19     wman2IfSsTftpCpltRetries,
20     wman2IfSsT26Timeout,
21     wman2IfSsDLManagProcTime,
22
23     -- Performance monitoring
24     wman2IfSsChannelNumber,
25     wman2IfSsStartFrame ,
26     wman2IfSsDuration,
27     wman2IfSsBasicReport,
28     wman2IfSsMeanCinrReport,
29     wman2IfSsStdDeviationCinrReport,
30     wman2IfSsMeanRssiReport,
31     wman2IfSsStdDeviationRssiReport,
32
33     -- Privacy sublayer
34     wman2IfSsPkmAuthState,
35     wman2IfSsPkmAuthKeySequenceNumber,
36     wman2IfSsPkmAuthExpiresOld,
37     wman2IfSsPkmAuthExpiresNew ,
38     wman2IfSsPkmAuthReset,
39     wman2IfSsPkmAuthentInfos,
40     wman2IfSsPkmAuthRequests,
41     wman2IfSsPkmAuthReplies,
42     wman2IfSsPkmAuthRejects,
43     wman2IfSsPkmAuthInvalids,
44     wman2IfSsPkmAuthRejectErrorCode,
45     wman2IfSsPkmAuthRejectErrorString,
46     wman2IfSsPkmAuthInvalidErrorCode,
47     wman2IfSsPkmAuthInvalidErrorString ,
48     wman2IfSsPkmAuthGraceTime,
49     wman2IfSsPkmTekGraceTime,
50     wman2IfSsPkmAuthWaitTimeout,
51     wman2IfSsPkmReauthWaitTimeout,
52     wman2IfSsPkmOpWaitTimeout,
53     wman2IfSsPkmRekeyWaitTimeout,
54     wman2IfSsPkmAuthRejectWaitTimeout,
55
56
57
58
59
60
61
62
63
64
65

```

```

1      wman2IfSsPkmTekSAType,
2      wman2IfSsPkmTekDataEncryptAlg,
3      wman2IfSsPkmTekDataAuthentAlg,
4      wman2IfSsPkmTekEncryptAlg,
5      wman2IfSsPkmTekState,
6      wman2IfSsPkmTekKeySequenceNumber,
7      wman2IfSsPkmTekExpiresOld,
8      wman2IfSsPkmTekExpiresNew,
9      wman2IfSsPkmTekKeyRequests,
10     wman2IfSsPkmTekKeyReplies,
11     wman2IfSsPkmTekKeyRejects,
12     wman2IfSsPkmTekInvalids,
13     wman2IfSsPkmTekAuthPends,
14     wman2IfSsPkmTekKeyRejectErrorCode,
15     wman2IfSsPkmTekKeyRejectErrorString,
16     wman2IfSsPkmTekInvalidErrorCode,
17     wman2IfSsPkmTekInvalidErrorString,
18     wman2IfSsDeviceCert,
19     wman2IfSsDeviceManufCert,
20
21     -- Notofocation
22     wman2IfSsTrapControlRegister,
23     wman2IfSsRssiLowThreshold,
24     wman2IfSsRssiHighThreshold,
25     wman2IfSsMacAddress,
26     wman2IfSsUnknownTlv,
27     wman2IfSsDynamicServiceType,
28     wman2IfSsDynamicServiceFailReason,
29     wman2IfSsRssiStatus,
30     wman2IfSsRssiStatusInfo}
31
32 STATUS          current
33
34 DESCRIPTION
35     "This group contains objects for SS, and are
36     independent of PHY."
37 ::= { wman2IfMibGroups 5 }
38
39
40
41
42
43
44 wman2IfMibBsOfdmGroup      OBJECT-GROUP
45     OBJECTS {wman2IfBsOfdmCtBasedResvTimeout,
46             wman2IfBsOfdmBwReqOppSize,
47             wman2IfBsOfdmRangReqOppSize,
48             wman2IfBsOfdmUplinkCenterFreq,
49             wman2IfBsOfdmNumSubChReqRegionFull,
50             wman2IfBsOfdmNumSymbolsReqRegionFull,
51             wman2IfBsOfdmSubChFocusCtCode,
52             wman2IfBsOfdmUpLinkChannelId,
53             wman2IfBsOfdmBsEIRP,
54             wman2IfBsOfdmChannelNumber,
55             wman2IfBsOfdmTTG,
56             wman2IfBsOfdmRTG,
57             wman2IfBsOfdmInitRngMaxRSS,
58             wman2IfBsOfdmDownlinkCenterFreq,
59             wman2IfBsOfdmBsId,
60             wman2IfBsOfdmMacVersion,
61             wman2IfBsOfdmFrameDurationCode,
62
63
64
65

```

```

1      wman2IfBsOfdmDownLinkChannelId,
2      wman2IfBsOfdmUcdFecCodeType,
3      wman2IfBsOfdmFocusCtPowerBoost,
4      wman2IfBsOfdmUcdTcsEnable,
5      wman2IfBsOfdmUcdBurstProfileRowStatus,
6      wman2IfBsOfdmDownlinkFrequency,
7      wman2IfBsOfdmDcdFecCodeType,
8      wman2IfBsOfdmDiucMandatoryExitThresh,
9      wman2IfBsOfdmDiucMinEntryThresh,
10     wman2IfBsOfdmTcsEnable,
11     wman2IfBsOfdmDcdBurstProfileRowStatus,
12     wman2IfBsOfdmMinReqRegionFullTxOpp,
13     wman2IfBsOfdmMinFocusedCtTxOpp,
14     wman2IfBsOfdmMaxRoundTripDelay,
15     wman2IfBsOfdmRangeAbortTimingThold,
16     wman2IfBsOfdmRangeAbortPowerThold,
17     wman2IfBsOfdmRangeAbortFreqThold,
18     wman2IfBsOfdmDnlkRateId,
19     wman2IfBsOfdmRatioG,
20     wman2IfBsSsOfdmReqCapFftSizes,
21     wman2IfBsSsOfdmReqCapSsDemodulator,
22     wman2IfBsSsOfdmReqCapSsModulator,
23     wman2IfBsSsOfdmReqCapFocusedCtSupport,
24     wman2IfBsSsOfdmReqCapTcSublayerSupport,
25     wman2IfBsSsOfdmRspCapFftSizes,
26     wman2IfBsSsOfdmRspCapSsDemodulator,
27     wman2IfBsSsOfdmRspCapSsModulator,
28     wman2IfBsSsOfdmRspCapFocusedCtSupport,
29     wman2IfBsSsOfdmRspCapTcSublayerSupport,
30     wman2IfBsOfdmCapFftSizes,
31     wman2IfBsOfdmCapSsDemodulator,
32     wman2IfBsOfdmCapSsModulator,
33     wman2IfBsOfdmCapFocusedCtSupport,
34     wman2IfBsOfdmCapTcSublayerSupport,
35     wman2IfBsOfdmCapCfgFftSizes,
36     wman2IfBsOfdmCapCfgSsDemodulator,
37     wman2IfBsOfdmCapCfgSsModulator,
38     wman2IfBsOfdmCapCfgFocusedCtSupport,
39     wman2IfBsOfdmCapCfgTcSublayerSupport}
40
41 STATUS      current
42 DESCRIPTION
43     "This group contains objects for BS and OFDM PHY."
44 ::= { wman2IfMibGroups 6 }
45
46
47
48
49
50
51
52
53
54 wman2IfMibSsOfdmGroup      OBJECT-GROUP
55     OBJECTS {wman2IfSsOfdmCtBasedResvTimeout,
56             wman2IfSsOfdmBwReqOppSize,
57             wman2IfSsOfdmRangReqOppSize,
58             wman2IfSsOfdmUplinkCenterFreq,
59             wman2IfSsOfdmNumSubChReqRegionFull,
60             wman2IfSsOfdmNumSymbolsReqRegionFull,
61             wman2IfSsOfdmSubChFocusCtCode,
62             wman2IfSsOfdmUpLinkChannelId,
63             wman2IfSsOfdmBsEIRP,

```

```

1      wman2IfSsOfdmChannelNumber,
2      wman2IfSsOfdmTTG,
3      wman2IfSsOfdmRTG,
4      wman2IfSsOfdmInitRngMaxRSS,
5      wman2IfSsOfdmDownlinkCenterFreq,
6      wman2IfSsOfdmBsId,
7      wman2IfSsOfdmMacVersion,
8      wman2IfSsOfdmFrameDurationCode,
9      wman2IfSsOfdmDownLinkChannelId,
10     wman2IfSsOfdmUcdFecCodeType,
11     wman2IfSsOfdmFocusCtPowerBoost,
12     wman2IfSsOfdmUcdTcsEnable,
13     wman2IfSsOfdmDownlinkFrequency,
14     wman2IfSsOfdmDcdFecCodeType,
15     wman2IfSsOfdmDiucMandatoryExitThresh,
16     wman2IfSsOfdmDiucMinEntryThresh,
17     wman2IfSsOfdmTcsEnable}
18
19 STATUS      current
20
21 DESCRIPTION
22     "This group contains objects for SS and OFDM PHY."
23 ::= { wman2IfMibGroups 7 }
24
25
26
27 wman2IfMibBsOfdmaGroup      OBJECT-GROUP
28     OBJECTS {wman2IfBsOfdmaCtBasedResvTimeout,
29             wman2IfBsOfdmaBwReqOppSize,
30             wman2IfBsOfdmaRangReqOppSize,
31             wman2IfBsOfdmaUplinkCenterFreq,
32             wman2IfBsOfdmaInitRngCodes,
33             wman2IfBsOfdmaPeriodicRngCodes,
34             wman2IfBsOfdmaBWReqCodes,
35             wman2IfBsOfdmaPerRngBackoffStart,
36             wman2IfBsOfdmaPerRngBackoffEnd,
37             wman2IfBsOfdmaStartOfRngCodes,
38             wman2IfBsOfdmaPermutationBase,
39             wman2IfBsOfdmaULAllocSubchBitmap,
40             wman2IfBsOfdmaOptPermULAllocSubchBitmap,
41             wman2IfBsOfdmaBandAMCAllocThreshold,
42             wman2IfBsOfdmaBandAMCReleaseThreshold,
43             wman2IfBsOfdmaBandAMCAllocTimer,
44             wman2IfBsOfdmaBandAMCReleaseTimer,
45             wman2IfBsOfdmaBandStatRepMAXPeriod,
46             wman2IfBsOfdmaBandAMCRetryTimer,
47             wman2IfBsOfdmaSafetyChAllocThreshold,
48             wman2IfBsOfdmaSafetyChReleaseThreshold,
49             wman2IfBsOfdmaSafetyChAllocTimer,
50             wman2IfBsOfdmaSafetyChReleaseTimer,
51             wman2IfBsOfdmaBinStatRepMAXPeriod,
52             wman2IfBsOfdmaSafetyChARetryTimer,
53             wman2IfBsOfdmaHARQAackDelayULBurst,
54             wman2IfBsOfdmaCQICHBandAMCTranaDelay,
55             wman2IfBsOfdmaBsEIRP,
56             wman2IfBsOfdmaChannelNumber,
57             wman2IfBsOfdmaTTG,
58             wman2IfBsOfdmaRTG,
59
60
61
62
63
64
65

```

```

1          wman2IfBsOfdmaInitRngMaxRSS,
2          wman2IfBsOfdmaDownlinkCenterFreq,
3          wman2IfBsOfdmaBsId,
4          wman2IfBsOfdmaMacVersion,
5          wman2IfBsOfdmaFrameDurationCode,
6          wman2IfBsOfdmaSizeCqichIdField,
7          wman2IfBsOfdmaHARQAackDelayBurst,
8          wman2IfBsOfdmaUcdFecCodeType,
9          wman2IfBsOfdmaRangingDataRatio,
10         wman2IfBsOfdmaNorCOverNOVERRIDE,
11         wman2IfBsOfdmaUcdBurstProfileRowStatus,
12         wman2IfBsOfdmaDownlinkFrequency,
13         wman2IfBsOfdmaDcdFecCodeType,
14         wman2IfBsOfdmaDiucMandatoryExitThresh,
15         wman2IfBsOfdmaDiucMinEntryThresh,
16         wman2IfBsOfdmaDcdBurstProfileRowStatus}
17
18 STATUS      current
19
20 DESCRIPTION
21     "This group contains objects for BS and OFDMA PHY."
22
23 ::= { wman2IfMibGroups 8 }
24
25
26 wman2IfMibSsOfdmaGroup      OBJECT-GROUP
27     OBJECTS {wman2IfSsOfdmaCtBasedResvTimeout,
28             wman2IfSsOfdmaBwReqOppSize,
29             wman2IfSsOfdmaRangReqOppSize,
30             wman2IfSsOfdmaUplinkCenterFreq,
31             wman2IfSsOfdmaInitRngCodes,
32             wman2IfSsOfdmaPeriodicRngCodes,
33             wman2IfSsOfdmaBWReqCodes,
34             wman2IfSsOfdmaPerRngBackoffStart,
35             wman2IfSsOfdmaPerRngBackoffEnd,
36             wman2IfSsOfdmaStartOfRngCodes,
37             wman2IfSsOfdmaPermutationBase,
38             wman2IfSsOfdmaULAllocSubchBitmap,
39             wman2IfSsOfdmaOptPermULAllocSubchBitmap,
40             wman2IfSsOfdmaBandAMCAllocThreshold,
41             wman2IfSsOfdmaBandAMCReleaseThreshold,
42             wman2IfSsOfdmaBandAMCAllocTimer,
43             wman2IfSsOfdmaBandAMCReleaseTimer,
44             wman2IfSsOfdmaBandStatRepMAXPeriod,
45             wman2IfSsOfdmaBandAMCRetryTimer,
46             wman2IfSsOfdmaSafetyChAllocThreshold,
47             wman2IfSsOfdmaSafetyChReleaseThreshold,
48             wman2IfSsOfdmaSafetyChAllocTimer,
49             wman2IfSsOfdmaSafetyChReleaseTimer,
50             wman2IfSsOfdmaBinStatRepMAXPeriod,
51             wman2IfSsOfdmaSafetyChARetryTimer,
52             wman2IfSsOfdmaHARQAackDelayULBurst,
53             wman2IfSsOfdmaCQICHBandAMCTranaDelay,
54             wman2IfSsOfdmaBsEIRP,
55             wman2IfSsOfdmaChannelNumber,
56             wman2IfSsOfdmaTTG,
57             wman2IfSsOfdmaRTG,
58             wman2IfSsOfdmaInitRngMaxRSS,
59
60
61
62
63
64
65

```

```

1          wman2IfSsOfdmaDownlinkCenterFreq,
2          wman2IfSsOfdmaBsId,
3          wman2IfSsOfdmaMacVersion,
4          wman2IfSsOfdmaFrameDurationCode,
5          wman2IfSsOfdmaSizeCqichIdField,
6          wman2IfSsOfdmaHARQAackDelayBurst,
7          wman2IfSsOfdmaUiucIndex,
8          wman2IfSsOfdmaUcdFecCodeType,
9          wman2IfSsOfdmaRangingDataRatio,
10         wman2IfSsOfdmaNorCOVerNOVERRIDE,
11         wman2IfSsOfdmaDiucIndex,
12         wman2IfSsOfdmaDownlinkFrequency,
13         wman2IfSsOfdmaDcdFecCodeType,
14         wman2IfSsOfdmaDiucMandatoryExitThresh,
15         wman2IfSsOfdmaDiucMinEntryThresh}
16
17     STATUS          current
18
19     DESCRIPTION
20         "This group contains objects for SS and OFDMA PHY."
21     ::= { wman2IfMibGroups 9 }
22
23
24
25 wman2IfMibBsNotificationGroup      NOTIFICATION-GROUP
26     NOTIFICATIONS {wman2IfBsSsStatusNotificationTrap,
27                    wman2IfBsSsDynamicServiceFailTrap,
28                    wman2IfBsSsRssiStatusChangeTrap,
29                    wman2IfBsSsPkmFailTrap,
30                    wman2IfBsSsRegistrerTrap}
31
32     STATUS          current
33
34     DESCRIPTION
35         "This group contains event notifications for BS."
36     ::= { wman2IfMibGroups 10 }
37
38
39 wman2IfMibSsNotificationGroup      NOTIFICATION-GROUP
40     NOTIFICATIONS {wman2IfSsTlvUnknownTrap,
41                    wman2IfSsDynamicServiceFailTrap,
42                    wman2IfSsDhcpSuccessTrap,
43                    wman2IfSsRssiStatusChangeTrap}
44
45     STATUS          current
46
47     DESCRIPTION
48         "This group contains event notifications for SS."
49     ::= { wman2IfMibGroups 11 }
50
51
52 wman2IfMibCmnPhsGroup              OBJECT-GROUP
53     OBJECTS {-- Payload header supression
54              wman2IfCmnPhsRulePhsField,
55              wman2IfCmnPhsRulePhsMask,
56              wman2IfCmnPhsRulePhsSize,
57              wman2IfCmnPhsRulePhsVerify}
58
59     STATUS          current
60
61     DESCRIPTION
62         "This group contains common objects for PHS."
63     ::= { wman2IfMibGroups 12 }
64
65
66 wman2IfMibBsPhsGroup              OBJECT-GROUP
67     OBJECTS {-- Payload header supression

```

```
1          wman2IfBsClassifierRulePhsSize,
2          wman2IfBsClassifierRulePhsMask,
3          wman2IfBsClassifierRulePhsVerify,
4          wman2IfBsClassifierRuleBitMap}
5
6      STATUS          current
7      DESCRIPTION
8          "This group contains BS objects for PHS."
9      ::= { wman2IfMibGroups 13 }
10
11  END
```


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 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

```

1  wmanIfBsMobility OBJECT IDENTIFIER ::= { wmanIfBsObjects x } -- e.g. x = 7
2
3
4
5
6  wmanIfBsHandoverConfiguration OBJECT IDENTIFIER ::= { wmanIfBsMobility 2 }
7
8
9  wmanIfBsOperatorId OBJECT-TYPE
10     SYNTAX Integer32
11     MAX-ACCESS read-write
12     STATUS current
13     DESCRIPTION
14         "An unique operator identifier."
15     ::= { wmanIfBsHandoverConfiguration 1 }
16
17
18  wmanIfBsId OBJECT-TYPE
19     SYNTAX WmanIfBsIdType
20     MAX-ACCESS read-write
21     STATUS current
22     DESCRIPTION
23         "An unique BS identifier."
24     ::= { wmanIfBsHandoverConfiguration 2 }
25
26
27  wmanIfBsHandoverSupport OBJECT-TYPE
28     SYNTAX BITS
29         {
30             {
31                 MDHO/FBSS HO not supported(0),
32                 FBSS/MDHO DLRF combining supported(1),
33                 MDHO DL soft combining supported monitoring single MAP from anchor
34                 BS(2),
35                 MDHO DL soft combining supported monitoring MAPS from active BSs(3),
36                 reserved1(5),
37                 reserved2(6),
38                 reserved3(7)
39             }
40             }
41     MAX-ACCESS read-write
42     STATUS current
43     DESCRIPTION
44         "The Handover supported field indicates what type(s) of HO the BS and the MS
45         supports."
46     ::= { wmanIfBsHandoverConfiguration 3 }
47
48
49
50  wmanIfBsHandoverSupport OBJECT-TYPE
51     SYNTAX BITS
52         {
53             {
54                 mdho/fbss HO not supported(0),
55                 fbss/mdho DLRF combining supported(1),
56                 mdho DL soft combining supported monitoring single MAP from anchor BS(2),
57                 mdho DL soft combining supported monitoring MAPS from active BSs(3)
58             }
59             }
60     MAX-ACCESS read-write
61     STATUS current
62     DESCRIPTION
63         "The Handover supported field indicates what type(s) of HO the BS and the MS
64         supports."
65

```

1 ::= { wmanIfBsHandoverConfiguration 3 }

2
3 wmanIfBsResourceRetainTime OBJECT-TYPE

4 SYNTAX Integer32

5 MAX-ACCESS read-write

6 STATUS current

7 DESCRIPTION

8
9 "The Resource_Retain_Time is the duration for MS s connection information
10 that will be retained in serving BS. BS shall start Resource_Retain_Time timer at MS notification
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

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

```

DESCRIPTION

"Waiting value for the DREG-REQ message re-transmission
(measured in frames)."

::= { wmanIfBsPagingGroupEntry 6 }

wmanIfBsMACHashSkipThreshold OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"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'."

::= { wmanIfBsPagingGroupEntry 7 }

wmanIfBsCDMATransmissionOpportunityAssignment OBJECT-TYPE

SYNTAX INTEGER

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"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."

::= { wmanIfBsPagingGroupEntry 8 }

wmanIfBsPagingResponseWindow OBJECT-TYPE

SYNTAX INTEGER

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The Page-Response Window indicates the Page-Response window for a MS
who is paged to transmit
the assigned code for CDMA ranging channel."

::= { wmanIfBsPagingGroupEntry 9 }

wmanIfBsIdleModeTimer OBJECT-TYPE

SYNTAX INTEGER (128..65536)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"MS timed interval to conduct
Location Update. Set timer to MS
Idle Mode Timeout capabilities
setting. Timer recycles on successful
Idle Mode Location Update."

::= { wmanIfBsPagingGroupEntry 10 }

wmanIfBsIdleModeSystemTimer OBJECT-TYPE

SYNTAX INTEGER (128..65536)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

1 "For BS acting as Paging Controller,
 2 timed interval to receive notification
 3 of MS Idle Mode Location Update. Set
 4 timer to MS Idle Mode Timeout. Timer
 5 recycles on successful Idle Mode
 6 Location Update."
 7
 8 ::= { wmanIfBsPagingGroupEntry 11 }
 9

10
 11 wmanIfBsPagingIntervalLength OBJECT-TYPE
 12 SYNTAX INTEGER (2..5)
 13 MAX-ACCESS read-write
 14 STATUS current
 15 DESCRIPTION
 16 "time duration of Paging Interval
 17 of the BS."
 18
 19 ::= { wmanIfBsPagingGroupEntry 12 }
 20

21
 22 wmanIfBsPagingCycle OBJECT-TYPE
 23 SYNTAX INTEGER
 24 MAX-ACCESS read-write
 25 STATUS current
 26 DESCRIPTION
 27 "Cycle in which the paging message is transmitted
 28 within the paging group."
 29
 30 ::= { wmanIfBsPagingGroupEntry 13 }
 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